



Panduan Pengguna

AWS IoT SiteWise



AWS IoT SiteWise: Panduan Pengguna

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara para pelanggan, atau dengan cara apa pun yang menghina atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan properti dari masing-masing pemilik, yang mungkin berafiliasi, terkait dengan, atau disponsori oleh Amazon, atau tidak.

Table of Contents

Apa itu AWS IoT SiteWise?	1
Cara kerjanya	2
Menelan data industri	2
Model aset untuk mengontekstualisasikan data yang dikumpulkan	3
Analisis menggunakan kueri, alarm, dan prediksi	5
Visualisasikan operasi	20
Menyimpan data	20
Mengintegrasikan dengan layanan lainnya	57
Konsep	57
Kasus penggunaan	62
Manufaktur	62
Makanan dan minuman	62
Energi dan utilitas	63
Memulai	64
Persyaratan	64
Menyiapkan sebuah Akun AWS	65
Mendaftar untuk Akun AWS	65
Membuat pengguna administratif	65
Menggunakan demo mulai cepat	66
Membuat AWS IoT SiteWise demo	67
Menghapus demo AWS IoT SiteWise	69
Tutorial	71
Menghitung OEE	71
Prasyarat	71
Cara menghitung OEE	72
Menelan data dari AWS IoT berbagai hal	74
Prasyarat	75
Langkah 1: Buat kebijakan	75
Langkah 2: Buat AWS IoT sesuatu	78
Langkah 3: Buat model aset perangkat	80
Langkah 4: Buat armada perangkat	82
Langkah 5: Mewakili perangkat	83
Langkah 6: Mewakili armada perangkat	84
Langkah 7: Kirim data ke perangkat	85

Langkah 8: Skrip klien perangkat	88
Langkah 9: Bersihkan sumber daya	95
Memvisualisasikan dan berbagi data di Monitor SiteWise	97
Prasyarat	98
Langkah 1: Buat portal	99
Langkah 2: Masuk ke portal	103
Langkah 3: Buat proyek	105
Langkah 4: Buat dasbor	109
Langkah 5: Jelajahi portal	116
Langkah 6: Membersihkan sumber daya	117
Menerbitkan pembaruan nilai properti ke Amazon DynamoDB	120
Prasyarat	120
Langkah 1: Konfigurasi AWS IoT SiteWise untuk mempublikasikan pembaruan nilai properti	121
Langkah 2: Buat Aturan	123
Langkah 3: Buat tabel DynamoDB	125
Langkah 4: Konfigurasi tindakan aturan	127
Langkah 5: Jelajahi data	128
Langkah 6: Bersihkan Sumber Daya	129
Menelan data ke AWS IoT SiteWise	133
Mengelola aliran data	133
Kelola aliran data	135
Menggunakan AWS IoT Core aturan	142
Memberikan akses yang diperlukan	143
Mengkonfigurasi tindakan aturan	144
Mengurangi biaya dengan konsumsi dasar	153
Menggunakan AWS IoT Events tindakan	154
Menggunakan pengelola AWS IoT Greengrass aliran	154
Menggunakan AWS IoT SiteWise API	155
Menggunakan CreateBulkImportJob API	158
Buat pekerjaan impor massal (AWS CLI)	159
Jelaskan pekerjaan impor massal (AWS CLI)	163
Daftar pekerjaan impor massal (AWS CLI)	164
Menggunakan SiteWise gateway Edge	165
Persyaratan	165
Persyaratan	166

Membuat gateway SiteWise Edge	169
Buat gateway SiteWise Edge	169
Menginstal perangkat lunak gateway SiteWise Edge di perangkat lokal Anda	171
Mengaktifkan pemrosesan data tepi	174
Menyiapkan kemampuan tepi	174
Memproses data di tepi	176
Mengkonfigurasi Penerbit	178
Mengkonfigurasi sumber data	179
Konfigurasi sumber OPC-UA	180
Mengkonfigurasi otentikasi sumber data	202
Memilih tujuan untuk data server sumber Anda	207
Menambahkan sumber data mitra	210
Keamanan	210
Menambahkan sumber data mitra	211
Siapkan docker di gateway Edge Anda SiteWise	212
Sumber data mitra	213
Menggunakan paket	213
Paket upgrade	214
Mengelola SiteWise gateway Edge	215
Mengelola gateway SiteWise Edge Anda dengan AWS IoT SiteWise konsol	216
Mengelola gateway SiteWise Edge menggunakan for AWS OpsHubAWS IoT SiteWise	217
Mengakses gateway SiteWise Edge Anda menggunakan kredensial sistem operasi lokal ...	218
Mengelola sertifikat gateway SiteWise Edge	221
Mengubah versi paket komponen gateway SiteWise Edge	222
Running SiteWise Edge di Tepi Industri Siemens	222
Prasyarat	223
Keamanan	223
Membuat file konfigurasi	224
Pemecahan Masalah	225
Hubungi kami	226
Menyaring aset	226
Menyiapkan penyaringan tepi	226
Menggunakan API	227
Semua API yang tersedia untuk digunakan dengan perangkat AWS IoT SiteWise edge	228
API khusus tepi	229
Tutorial: Mendapatkan daftar model aset	231

Cadangkan dan pulihkan SiteWise gateway Edge	241
Backup harian data metrik	241
Kembalikan gateway SiteWise Edge	242
Kembalikan AWS IoT SiteWise data	243
Validasi backup dan restorasi yang berhasil	244
Menyiapkan gateway SiteWise Edge (AWS IoT Greengrass Version 1	246
Memilih perangkat gateway AWS IoT Greengrass V1 SiteWise Edge	247
Mengkonfigurasi gateway AWS IoT Greengrass V1 SiteWise Edge	248
Mengkonfigurasi sumber data di gateway AWS IoT Greengrass V1 SiteWise Edge	267
Pemodelan aset industri	288
Status aset dan model	290
Memeriksa status aset	290
Memeriksa status model aset atau model komponen	292
Model komposit khusus (Komponen)	294
Model komposit kustom sebaris	295
C model komposit component-model-based kustom	297
Menggunakan jalur untuk mereferensikan properti model komposit khusus	298
Bekerja dengan ID objek	300
Bekerja dengan objek UUID	301
Menggunakan ID eksternal	301
Membuat model aset dan model komponen	303
Membuat model aset	304
Membuat model komponen	319
Mendefinisikan properti data	323
Membuat model komposit khusus (Komponen)	403
Menciptakan aset	408
Membuat aset (konsol)	408
Membuat aset (AWS CLI)	409
Mengkonfigurasi aset baru	410
Mencari aset	411
Prasyarat	411
Pencarian lanjutan di Konsol AWS IoT SiteWise	411
Memetakan aliran data industri ke properti aset	414
Menyetel alias properti (konsol)	416
Mengatur alias properti (AWS CLI)	417
Memperbarui nilai atribut	419

Mengasosiasikan dan memisahkan aset	422
Mengasosiasikan dan memisahkan aset (konsol)	422
Mengasosiasikan dan memisahkan aset (AWS CLI)	424
Memperbarui aset dan model	426
Memperbarui aset	426
Memperbarui model aset dan model komponen	428
Memperbarui model komposit khusus (Komponen)	432
Menghapus aset dan model	435
Menghapus aset	435
Menghapus model aset	438
Operasi massal dengan aset dan model	439
Konsep dan terminologi kunci	440
Fungsionalitas yang didukung	441
Prasyarat operasi massal	442
Menjalankan pekerjaan impor massal	445
Menjalankan pekerjaan ekspor massal	447
Pelacakan kemajuan pekerjaan dan penanganan kesalahan	450
Impor contoh metadata	455
Contoh metadata ekspor	469
AWS IoT SiteWise skema pekerjaan transfer metadata	472
Memantau data dengan alarm	491
Jenis alarm	491
Status alarm	492
Properti status alarm	493
Mendefinisikan alarm pada model aset	496
Mendefinisikan alarm AWS IoT Events	499
Mendefinisikan alarm eksternal	534
Mengkonfigurasi alarm pada aset	536
Mengkonfigurasi nilai ambang (konsol)	537
Mengkonfigurasi nilai ambang (AWS CLI)	537
Mengkonfigurasi pengaturan notifikasi (konsol)	539
Mengkonfigurasi pengaturan notifikasi (CLI)	540
Menanggapi alarm	542
Menanggapi alarm (konsol)	542
Menanggapi alarm (API)	546
Menelan status alarm eksternal	546

Memetakan aliran status alarm eksternal	547
Menelan data status alarm	549
Memantau data dengan portal web	551
SiteWise Memantau peran	552
Federasi SAFL	554
SiteWise Pantau konsep	555
Memulai	557
Membuat portal	558
Mengkonfigurasi portal Anda	559
Mengundang administrator	563
Menambahkan pengguna portal	566
Membuat dasbor (CLI)	570
Mengaktifkan alarm untuk portal Anda	576
Mengaktifkan portal Anda di tepi	579
Mengelola portal Anda	579
Mengubah atribut portal	581
Menambahkan atau menghapus administrator portal	581
Mengirim undangan email ke administrator portal	584
Menambahkan atau menghapus pengguna portal	585
Menghapus portal	588
Memantau data dengan aplikasi dasbor IoT	590
Data kueri dari AWS IoT SiteWise	591
Kueri nilai aset saat ini	592
Kueri nilai saat ini properti aset (konsol)	592
Kueri nilai saat ini properti aset (AWS CLI)	592
Kueri nilai properti aset historis	593
Kueri riwayat nilai untuk properti aset (AWS CLI)	594
Agregat properti aset kueri	595
Agregat untuk properti aset (API)	596
Agregat untuk properti aset (AWS CLI)	597
AWS IoT SiteWise bahasa query	598
Prasyarat	599
Referensi bahasa kueri	599
Berinteraksi dengan layanan lain	607
Memahami topik MQTT properti aset	608
Bekerja dengan pemberitahuan properti aset	608

Mengaktifkan pemberitahuan properti aset (konsol)	609
Mengaktifkan notifikasi properti aset (AWS CLI)	609
Menanyakan pesan pemberitahuan properti aset	611
Mengekspor data ke Amazon S3	614
Buat AWS CloudFormation tumpukan	615
Lihat data Anda di Amazon S3	617
Analisis data yang diekspor	619
Sumber daya template dibuat	626
Integrasi dengan Grafana	630
Berintegrasi dengan AWS IoT TwinMaker	631
Mengaktifkan integrasi	632
Mengintegrasikan dan AWS IoT SiteWiseAWS IoT TwinMaker	632
Mendeteksi anomali peralatan	633
Menambahkan definisi prediksi (konsol)	635
Melatih prediksi (konsol)	638
Memulai atau menghentikan inferensi pada prediksi (konsol)	639
Menambahkan definisi prediksi (CLI)	639
Melatih prediksi dan inferensi awal (CLI)	643
Pelatihan prediksi (CLI)	644
Memulai atau menghentikan inferensi pada prediksi (CLI)	646
Mengelola penyimpanan data	649
Konfigurasi pengaturan penyimpanan	650
Dampak retensi data	650
Konfigurasi pengaturan penyimpanan untuk tingkat hangat (konsol)	651
Konfigurasi pengaturan penyimpanan untuk tingkat hangat (AWS CLI)	652
Konfigurasi pengaturan penyimpanan untuk tingkat dingin (konsol)	656
Konfigurasi pengaturan penyimpanan untuk tingkat dingin (AWS CLI)	658
Memecahkan masalah pengaturan penyimpanan	663
Kesalahan: Bucket tidak ada	664
Kesalahan: Akses ditolak ke jalur Amazon S3	664
Kesalahan: Peran ARN tidak dapat diasumsikan	665
Kesalahan: Gagal mengakses bucket Amazon S3 lintas wilayah	665
Jalur file dan skema data yang disimpan di tingkat dingin	665
Data peralatan (pengukuran)	665
Metrik, transformasi, dan agregat	670
Metadata aset	674

Metadata hierarki aset	678
File indeks data penyimpanan	680
Keamanan	681
Perlindungan data	682
Privasi lalu lintas jaringan internet	683
Enkripsi data	683
Enkripsi diam	684
Enkripsi dalam bergerak	686
Manajemen kunci	688
Pengelolaan identitas dan akses	690
Audiens	690
Mengautentikasi dengan identitas	691
Bagaimana AWS IoT SiteWise bekerja dengan IAM	694
Kebijakan terkelola	714
Peran terkait layanan	718
Menyiapkan izin untuk alarm	732
Pencegahan confused deputy lintas layanan	738
Pemecahan Masalah	740
Validasi kepatuhan	742
Ketangguhan	743
Keamanan infrastruktur	743
Konfigurasi dan analisis kerentanan	744
Titik akhir VPC	745
Operasi API yang didukung	745
Membuat titik akhir VPC antarmuka	748
Mengakses AWS IoT SiteWise melalui antarmuka VPC endpoint	748
Membuat kebijakan titik akhir VPC	750
Praktik terbaik keamanan	751
Menggunakan kredensi otentikasi di server OPC-UA Anda	751
Gunakan mode komunikasi terenkripsi untuk server OPC-UA Anda	751
Perbarui komponen Anda	752
Enkripsi sistem file gateway SiteWise Edge Anda	752
Akses aman ke konfigurasi tepi Anda	752
Grant SiteWise Monitor pengguna izin minimum yang mungkin	752
Jangan mengekspos informasi sensitif	753
Ikuti praktik terbaik AWS IoT Greengrass keamanan	753

Lihat juga	753
Pencatatan log dan pemantauan	754
Memantau log layanan	754
Mengelola masuk AWS IoT SiteWise	756
Contoh: entri file AWS IoT SiteWise log	758
Memantau log gateway SiteWise Edge	758
Menggunakan CloudWatch Log Amazon	758
Menggunakan log layanan	760
Menggunakan log peristiwa	762
Pemantauan dengan CloudWatch metrik Amazon	765
AWS IoT Greengrass Version 2 metrik gateway	765
AWS IoT Greengrass Version 1 metrik gateway	770
Membuat log panggilan API dengan AWS CloudTrail	775
AWS IoT SiteWise informasi di CloudTrail	776
AWS IoT SiteWise peristiwa data di CloudTrail	776
AWS IoT SiteWise acara manajemen di CloudTrail	779
Contoh: entri file AWS IoT SiteWise log	779
Menandai Sumber Daya Anda	781
Menggunakan tag di AWS IoT SiteWise	781
Menandai dengan AWS Management Console	781
Menandai dengan API AWS IoT SiteWise	782
Menggunakan tanda dengan kebijakan IAM	783
Pemecahan Masalah	785
Memecahkan masalah impor dan ekspor massal	785
Memecahkan masalah portal	786
Pengguna dan administrator tidak dapat mengakses portal AWS IoT SiteWise	786
Memecahkan masalah gateway	787
Mengkonfigurasi dan mengakses log gateway SiteWise Edge	788
Memecahkan masalah gateway SiteWise Edge	788
Memecahkan masalah AWS IoT Greengrass	793
Memecahkan masalah tindakan aturan AWS IoT SiteWise	794
Mengkonfigurasi log AWS IoT Core	794
Mengonfigurasi tindakan kesalahan penerbitan ulang	795
Memecahkan masalah	797
Memecahkan masalah aturan	799
Memecahkan masalah aturan	801

Titik akhir dan kuota	805
Titik akhir	805
.....	805
.....	805
.....	806
.....	806
.....	806
.....	806
.....	806
Kuota	807
Kuota untuk deteksi anomali	820
Riwayat dokumen	821
Daftar istilah AWS	839
.....	dcccxl

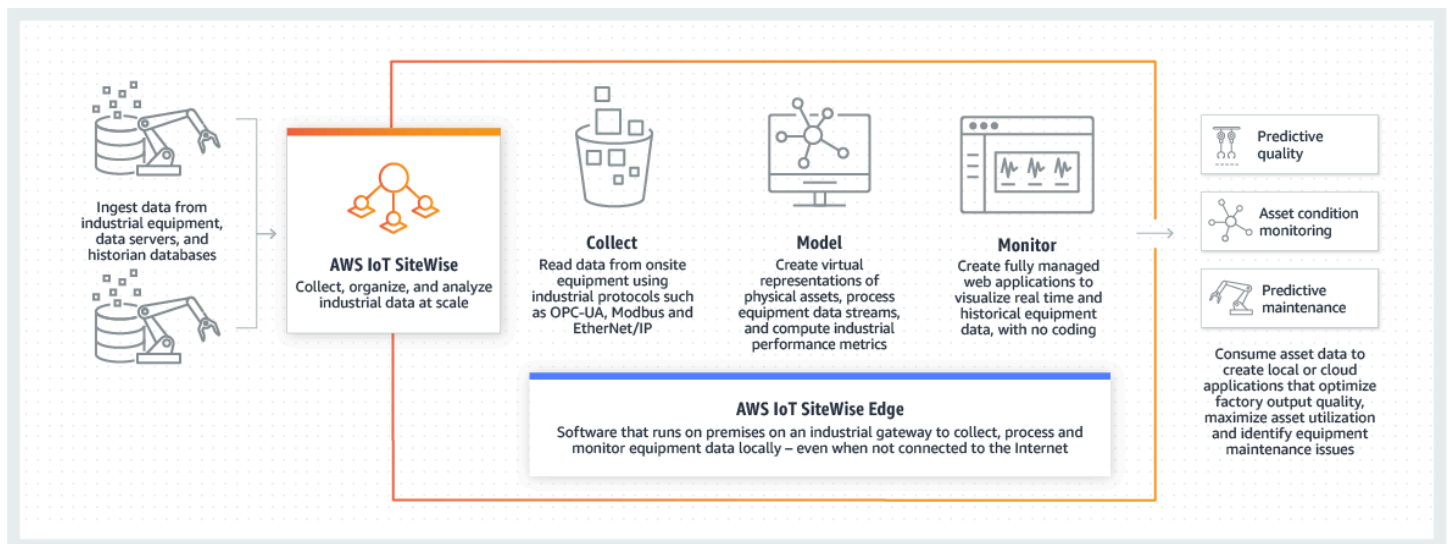
Apa itu AWS IoT SiteWise?

AWS IoT SiteWise adalah layanan terkelola yang memudahkan untuk mengumpulkan, menyimpan, mengatur, dan memantau data dari peralatan industri dalam skala besar untuk membantu Anda membuat keputusan yang lebih baik dan berbasis data. Anda dapat menggunakannya AWS IoT SiteWise untuk memantau operasi di seluruh fasilitas, dengan cepat menghitung metrik kinerja industri umum, dan membuat aplikasi yang menganalisis data peralatan industri untuk mencegah masalah peralatan yang mahal dan mengurangi kesenjangan dalam produksi.

AWS IoT SiteWise Monitor memungkinkan pengguna operasional Anda untuk dengan cepat membuat aplikasi web untuk melihat dan menganalisis data industri Anda secara real-time. Anda dapat memperoleh wawasan tentang operasi industri Anda dengan mengonfigurasi dan memantau metrik seperti waktu rata-rata antara kegagalan dan efektivitas peralatan secara keseluruhan (OEE).

AWS IoT SiteWise Edge adalah komponen AWS IoT SiteWise yang memungkinkan pengumpulan, penyimpanan, dan pemrosesan data pada perangkat lokal. Ini berguna jika Anda memiliki akses terbatas ke internet atau perlu menjaga data Anda tetap pribadi.

Diagram berikut menunjukkan arsitektur dasar AWS IoT SiteWise:



Topik

- [Bagaimana cara AWS IoT SiteWise kerja](#)
- [AWS IoT SiteWise konsep](#)
- [Gunakan kasus untuk AWS IoT SiteWise](#)

Bagaimana cara AWS IoT SiteWise kerja

AWS IoT SiteWise menawarkan kerangka pemodelan sumber daya yang dapat Anda gunakan untuk membuat representasi perangkat, proses, dan fasilitas industri Anda. Representasi peralatan dan proses Anda disebut model aset di AWS IoT SiteWise. Dengan model aset, Anda menentukan data mentah yang akan dikonsumsi dan cara memprosesnya menjadi metrik yang berguna. Bangun dan visualisasikan aset dan model untuk operasi industri Anda di [AWS IoT SiteWise konsol](#). Anda juga dapat mengonfigurasi model aset untuk mengumpulkan dan memproses data di edge atau di AWS Cloud.

Topik

- [Menelan data industri](#)
- [Model aset untuk mengontekstualisasikan data yang dikumpulkan](#)
- [Analisis menggunakan kueri, alarm, dan prediksi](#)
- [Visualisasikan operasi](#)
- [Menyimpan data](#)
- [Mengintegrasikan dengan layanan lainnya](#)

Menelan data industri

Mulailah menggunakan AWS IoT SiteWise dengan menelan data industri. Menelan data Anda dilakukan dengan salah satu dari beberapa cara:

- Konsumsi langsung dari server di tempat: Gunakan protokol seperti OPC-UA untuk membaca data langsung dari perangkat di tempat. Menyebarkan perangkat lunak gateway SiteWise Edge AWS IoT Greengrass V2, kompatibel dengan, pada berbagai platform seperti gateway industri umum atau server virtual. Anda dapat menghubungkan hingga 100 server OPC-UA ke satu AWS IoT SiteWise gateway. Untuk informasi selengkapnya, lihat [SiteWise Persyaratan gateway tepi](#).

Perhatikan bahwa protokol seperti Modbus TCP dan Ethernet/IP (EIP) didukung melalui kemitraan kami dengan dalam konteks. Domatica AWS IoT Greengrass V2

- Pemrosesan data tepi dengan paket: Tingkatkan gateway SiteWise Edge Anda dengan menambahkan paket untuk mengaktifkan kemampuan edge yang komprehensif. Dengan SiteWise Edge, tersedia di AWS IoT Greengrass V2, pemrosesan data dijalankan langsung di tempat sebelum dikirim dengan aman ke AWS Cloud menggunakan AWS IoT Greengrass aliran. Untuk informasi selengkapnya, lihat [Menggunakan paket](#).

- Penyerapan adaptif melalui Amazon S3 dengan operasi massal: Saat bekerja dengan sejumlah besar aset atau model aset, gunakan operasi massal untuk mengimpor dan mengekspor sumber daya massal dari bucket Amazon S3. Untuk informasi selengkapnya, lihat [Operasi massal dengan aset dan model](#).
- Pesan MQTT dengan Aturan AWS IoT Inti: Untuk perangkat yang terhubung ke AWS IoT Core yang mengirim pesan MQTT, gunakan mesin aturan AWS IoT Inti untuk mengarahkan pesan tersebut ke AWS IoT SiteWise. Jika Anda memiliki perangkat yang terhubung ke AWS IoT Core yang mengirim pesan [MQTT](#), gunakan mesin aturan Inti untuk mengarahkan pesan tersebut. AWS IoT SiteWise Untuk informasi selengkapnya, lihat [Menelan data menggunakan aturan AWS IoT Core](#).
- Penyerapan data yang dipicu peristiwa: Gunakan AWS IoT Events tindakan untuk mengonfigurasi SiteWise tindakan IoT AWS IoT Events untuk mengirim data saat peristiwa terjadi. AWS IoT SiteWise Untuk informasi selengkapnya, lihat [Menelan data dari AWS IoT Events](#).
- AWS IoT SiteWise API: Aplikasi Anda di Edge atau di cloud dapat langsung mengirim data ke AWS IoT SiteWise. Lihat informasi yang lebih lengkap di [Menyerap data menggunakan API AWS IoT SiteWise](#)

Model aset untuk mengontekstualisasikan data yang dikumpulkan

Setelah menelan data, Anda dapat menggunakan data untuk membuat representasi virtual dari aset, proses, dan fasilitas Anda dengan membangun model operasi fisik Anda. Aset, yang mewakili perangkat atau proses, mentransmisikan aliran data ke Cloud. AWS Aset juga dapat menandakan pengelompokan perangkat logis. Hirarki dibentuk dengan mengaitkan aset untuk mencerminkan operasi yang kompleks. Hirarki ini memungkinkan aset untuk mengakses data dari aset anak terkait. Aset dibuat dari model aset. Model aset adalah struktur deklaratif yang menstandarisasi format aset. Gunakan kembali komponen aset untuk organisasi dan pemeliharaan model Anda. Lihat informasi yang lebih lengkap di [Pemodelan aset industri](#)

Dengan AWS IoT SiteWise, Anda dapat mengonfigurasi aset Anda untuk mengubah data yang masuk menjadi metrik dan transformasi kontekstual.

- Mengubah pekerjaan saat menerima data peralatan.
- Metrik dihitung pada interval yang Anda tentukan.

Metrik dan transformasi berlaku untuk aset individu atau beberapa aset. AWS IoT SiteWise secara otomatis menghitung agregat statistik yang umum digunakan seperti rata-rata, jumlah, dan hitungan, di berbagai kerangka waktu yang relevan dengan data, metrik, dan transformasi peralatan Anda.

Aset dapat disinkronkan menggunakan AWS IoT TwinMaker. Untuk informasi selengkapnya, lihat

Untuk mengintegrasikan AWS IoT SiteWise dan AWS IoT TwinMaker, Anda harus memiliki yang berikut:

- AWS IoT SiteWise pengaturan peran terkait layanan di akun Anda
- AWS IoT TwinMaker pengaturan peran terkait layanan di akun Anda
- AWS IoT TwinMaker ruang kerja dengan ID `IoTSiteWiseDefaultWorkspace` di akun Anda di Wilayah.

Untuk mengintegrasikan dengan menggunakan AWS IoT SiteWise konsol

Saat Anda melihat AWS IoT TwinMaker spanduk Integrasi dengan di konsol, pilih Berikan izin. Prasyarat dibuat di akun Anda.

Untuk mengintegrasikan dengan menggunakan AWS CLI

Untuk mengintegrasikan AWS IoT SiteWise dan AWS IoT TwinMaker dengan menggunakan AWS CLI, masukkan perintah berikut:

1. Panggil `CreateServiceLinkedRole` dengan sebuah `AWSServiceName` dari `iotsitewise.amazonaws.com`.

```
aws iam create-service-linked-role --aws-service-name iotsitewise.amazonaws.com
```

2. Panggil `CreateServiceLinkedRole` dengan sebuah `AWSServiceName` dari `iottwinmaker.amazonaws.com`.

```
aws iam create-service-linked-role --aws-service-name iottwinmaker.amazonaws.com
```

3. Panggil `CreateWorkspace` dengan sebuah ID dari `IoTSiteWiseDefaultWorkspace`.

```
aws iottwinmaker create-workspace --workspace-id IoTSiteWiseDefaultWorkspace
```


Analisis menggunakan kueri, alarm, dan prediksi

Analisis tanggal yang dikumpulkan AWS IoT SiteWise dengan menjalankan kueri dan mengatur alarm. Anda juga dapat menggunakan Amazon Lookout untuk secara otomatis mendeteksi anomali dalam metrik dan mengidentifikasi akar penyebabnya.

- Atur alarm khusus untuk mengingatkan tim Anda saat peralatan atau proses menyimpang dari kinerja optimal, memastikan identifikasi dan penyelesaian masalah yang cepat. Untuk informasi selengkapnya, lihat [Memantau data dengan alarm](#).
- Gunakan operasi AWS IoT SiteWise API untuk menanyakan nilai saat ini, nilai historis, dan agregat properti aset Anda selama interval waktu tertentu. Untuk informasi selengkapnya, lihat [Data kueri dari AWS IoT SiteWise](#).
- Gunakan deteksi anomali dengan Amazon Lookout for Equipment untuk mengidentifikasi dan memvisualisasikan perubahan peralatan atau kondisi pengoperasian. Dengan deteksi anomali, Anda dapat menentukan langkah-langkah pemeliharaan preventif untuk operasi Anda. Integrasi ini memungkinkan pelanggan untuk menyinkronkan data antara AWS IoT SiteWise dan Amazon Lookout for Equipment. Untuk informasi selengkapnya, lihat

Note

Deteksi anomali hanya tersedia di Wilayah di mana Amazon Lookout for Equipment tersedia.

Anda dapat berintegrasi AWS IoT SiteWise dengan Amazon Lookout for Equipment untuk mendapatkan wawasan tentang peralatan industri Anda melalui deteksi anomali dan pemeliharaan prediktif peralatan industri. Lookout for Equipment adalah layanan machine learning (ML) untuk memantau peralatan industri yang mendeteksi perilaku peralatan abnormal dan mengidentifikasi potensi kegagalan. Dengan Lookout for Equipment, Anda dapat menerapkan program pemeliharaan prediktif dan mengidentifikasi proses peralatan yang kurang optimal. Untuk informasi selengkapnya tentang Lookout for Equipment, lihat [Apa itu Amazon Lookout for Equipment?](#) di Panduan Pengguna Amazon Lookout for Equipment.

Saat Anda membuat prediksi untuk melatih model ML guna mendeteksi perilaku peralatan anomali, AWS IoT SiteWise kirimkan nilai properti aset ke Lookout for Equipment untuk melatih model ML guna mendeteksi perilaku peralatan anomali. Untuk menentukan definisi prediksi pada model aset, Anda menentukan peran IAM yang diperlukan untuk Lookout for Equipment untuk

mengakses data dan properti yang akan dikirim ke Lookout for Equipment dan mengirim data yang diproses ke Amazon S3. Untuk informasi selengkapnya, lihat [Membuat model aset](#).

Untuk mengintegrasikan AWS IoT SiteWise dan Lookout for Equipment, Anda akan melakukan langkah-langkah tingkat tinggi berikut:

- Tambahkan definisi prediksi pada model aset yang menguraikan properti apa yang ingin Anda lacak. Definisi prediksi adalah kumpulan pengukuran, transformasi, dan metrik yang dapat

digunakan kembali yang digunakan untuk membuat prediksi pada aset yang didasarkan pada model aset tersebut.

- Latih prediksi berdasarkan data historis yang Anda berikan.
- Jadwal inferensi, yang menceritakan AWS IoT SiteWise seberapa sering menjalankan prediksi tertentu.

Setelah inferensi dijadwalkan, model Lookout for Equipment memantau data yang diterimanya dari peralatan Anda dan mencari anomali dalam perilaku peralatan. Anda dapat melihat dan menganalisis hasil di SiteWise Monitor, menggunakan operasi AWS IoT SiteWise GET API, atau konsol Lookout for Equipment. Anda juga dapat membuat alarm menggunakan detektor alarm dari model aset untuk mengingatkan Anda tentang perilaku peralatan yang tidak normal.

Topik

-
- [Menambahkan definisi prediksi \(konsol\)](#)
 - [Melatih prediksi \(konsol\)](#)
 - [Memulai atau menghentikan inferensi pada prediksi \(konsol\)](#)
 - [Menambahkan definisi prediksi \(CLI\)](#)
 - [Melatih prediksi dan inferensi awal \(CLI\)](#)
 - [Pelatihan prediksi \(CLI\)](#)
 - [Memulai atau menghentikan inferensi pada prediksi \(CLI\)](#)
-

Menambahkan definisi prediksi (konsol)

Untuk mulai mengirim data yang dikumpulkan oleh AWS IoT SiteWise Lookout for Equipment, Anda harus menambahkan definisi prediksi ke model AWS IoT SiteWise aset.

Untuk menambahkan definisi prediksi ke model AWS IoT SiteWise aset

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model dan pilih model aset yang ingin Anda tambahkan definisi prediksi.
3. Pilih Prediksi.
4. Pilih Tambahkan definisi prediksi.
5. Tentukan detail tentang definisi prediksi.
 - a. Masukkan Nama unik dan Deskripsi untuk definisi prediksi Anda. Pilih nama dengan serius karena setelah Anda membuat definisi prediksi, Anda tidak dapat mengubah namanya.
 - b. Buat atau pilih peran izin IAM yang memungkinkan Anda membagikan data aset Anda AWS IoT SiteWise dengan Amazon Lookout for Equipment. Peran tersebut harus memiliki IAM dan kebijakan kepercayaan berikut. Untuk bantuan membuat peran, lihat [Membuat peran menggunakan kebijakan kepercayaan khusus \(konsol\)](#).

Kebijakan IAM

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "L4EPermissions",
      "Effect": "Allow",
      "Action": [
        "lookoutequipment:CreateDataset",
        "lookoutequipment:CreateModel",
        "lookoutequipment:CreateInferenceScheduler",
        "lookoutequipment:DescribeDataset",
        "lookoutequipment:DescribeDataIngestionJob",
        "lookoutequipment:DescribeModel",
        "lookoutequipment:DescribeInferenceScheduler",
        "lookoutequipment:ListInferenceExecutions",
        "lookoutequipment:StartDataIngestionJob",
        "lookoutequipment:StartInferenceScheduler",
        "lookoutequipment:UpdateInferenceScheduler",
        "lookoutequipment:StopInferenceScheduler"
      ]
    }
  ],
```

```

    "Resource": "*"
  },
  {
    "Sid": "S3Permissions",
    "Effect": "Allow",
    "Action": [
      "s3:CreateBucket",
      "s3:ListBucket",
      "s3:PutObject",
      "s3:GetObject"
    ],
    "Resource": ["arn:aws:s3:::iotsitewise-*"]
  },
  {
    "Sid": "IAMPermissions",
    "Effect": "Allow",
    "Action": [
      "iam:GetRole",
      "iam:PassRole"
    ],
    "Resource": "arn:aws:iam:::role/*"
  }
]
}

```

Kebijakan kepercayaan

```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "Service": "iotsitewise.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "account_id"
      },
      "ArnEquals": {
        "aws:SourceArn":
"arn:aws:iotsitewise:region:account_id:asset/*"
      }
    }
  ]
}

```

```
    },  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "Service": "lookoutequipment.amazonaws.com"  
      },  
      "Action": "sts:AssumeRole",  
      "Condition": {  
        "StringEquals": {  
          "aws:SourceAccount": "account_id"  
        },  
        "ArnEquals": {  
          "aws:SourceArn":  
            "arn:aws:lookoutequipment:region:account_id:*"  
        }  
      }  
    }  
  ]  
}
```

c. Pilih Berikutnya.

6. Pilih atribut data (pengukuran, transformasi, dan metrik) yang ingin Anda kirim ke Lookout for Equipment.
 - a. (Opsional) Pilih pengukuran.
 - b. (Opsional) Pilih transformasi.
 - c. (Opsional) Pilih metrik.
 - d. Pilih Berikutnya.
7. Tinjau pilihan Anda. Untuk menambahkan definisi prediksi ke model aset, pada halaman ringkasan, pilih Tambahkan definisi prediksi.

Anda juga dapat Mengedit atau Menghapus definisi prediksi yang ada yang memiliki prediksi aktif yang dilampirkan.

Melatih prediksi (konsol)

Setelah Anda menambahkan definisi prediksi ke model aset, Anda dapat melatih prediksi yang ada di aset Anda.

Untuk melatih prediksi di AWS IoT SiteWise

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset, dan pilih aset yang ingin dipantau.
3. Pilih Prediksi.
4. Pilih prediksi yang ingin Anda latih.
5. Di bawah Tindakan, pilih Mulai pelatihan, dan lakukan hal berikut:
 - a. Di bawah Detail Prediksi, pilih peran izin IAM yang memungkinkan AWS IoT SiteWise untuk berbagi data aset Anda dengan Lookout for Equipment. Jika Anda perlu membuat peran baru, pilih Buat peran baru.
 - b. Untuk pengaturan data Pelatihan, masukkan rentang waktu data Pelatihan untuk memilih data mana yang akan digunakan untuk melatih prediksi.
 - c. (Opsional) Untuk label Data, berikan bucket dan awalan Amazon S3 yang menyimpan data pelabelan Anda. Untuk informasi selengkapnya tentang pelabelan data, lihat [Melabeli data Anda di Panduan Pengguna Amazon Lookout for Equipment](#).
 - d. Pilih Berikutnya.
6. (Opsional) Jika Anda ingin prediksi aktif segera setelah menyelesaikan pelatihan, di bawah Pengaturan lanjutan, pilih Aktifkan prediksi secara otomatis setelah pelatihan, lalu lakukan hal berikut:
 - a. Di bawah Input data, untuk frekuensi upload Data, tentukan seberapa sering data diunggah, dan untuk waktu tunda Offset, tentukan berapa banyak buffer yang akan digunakan.
 - b. Pilih Berikutnya.
7. Tinjau detail prediksi dan pilih Simpan dan mulai.

Memulai atau menghentikan inferensi pada prediksi (konsol)

Note

Biaya Lookout for Equipment berlaku untuk kesimpulan terjadwal dengan data yang ditransfer antara dan AWS IoT SiteWise Lookout for Equipment. Untuk informasi selengkapnya, lihat [harga Amazon Lookout for Equipment](#).

Jika Anda menambahkan prediksi tetapi tidak memilih untuk mengaktifkannya setelah pelatihan, Anda harus mengaktifkannya agar dapat mulai memantau aset Anda.

Untuk memulai inferensi untuk prediksi

1. Navigasikan ke konsol AWS IoT SiteWise tersebut.
 2. Di panel navigasi, pilih Aset, dan pilih aset tempat prediksi ditambahkan.
 3. Pilih Prediksi.
 4. Pilih prediksi yang ingin Anda aktifkan.
 5. Di bawah Tindakan, pilih Mulai inferensi, dan lakukan hal berikut:
 - a. Di bawah Input data, untuk frekuensi upload Data, tentukan seberapa sering data diunggah, dan untuk waktu tunda Offset, tentukan berapa banyak buffer yang akan digunakan.
 - b. Pilih Simpan dan mulai.
-

Untuk menghentikan inferensi untuk prediksi

1. Navigasikan ke konsol AWS IoT SiteWise tersebut.
 2. Di panel navigasi, pilih Aset, dan pilih aset tempat prediksi ditambahkan.
 3. Pilih Prediksi.
 4. Pilih prediksi yang ingin Anda hentikan.
 5. Di bawah Tindakan, pilih Hentikan inferensi.
-

Menambahkan definisi prediksi (CLI)

Untuk menentukan definisi prediksi pada model aset baru atau yang sudah ada, Anda dapat menggunakan AWS Command Line Interface (AWS CLI). Setelah Anda menentukan definisi prediksi pada model aset, Anda melatih, dan menjadwalkan inferensi untuk, prediksi pada aset AWS IoT SiteWise untuk melakukan deteksi anomali dengan Lookout for Equipment.

Prasyarat

Untuk menyelesaikan langkah-langkah ini, Anda harus memiliki model aset dan setidaknya satu aset yang dibuat. Lihat informasi yang lebih lengkap di [Membuat model aset \(AWS CLI\)](#) dan [Membuat aset \(AWS CLI\)](#).

Jika Anda baru AWS IoT SiteWise, Anda harus memanggil operasi `CreateBulkImportJob` API untuk mengimpor nilai properti aset ke dalam AWS IoT SiteWise, yang akan digunakan untuk melatih model. Untuk informasi selengkapnya, lihat [Buat pekerjaan impor massal \(AWS CLI\)](#).

Untuk menambahkan definisi prediksi

1. Buat file bernama `asset-model-payload.json`. Ikuti langkah-langkah di bagian lain ini untuk menambahkan detail model aset Anda ke file, tetapi jangan kirimkan permintaan untuk membuat atau memperbarui model aset.
 - Untuk informasi selengkapnya tentang cara membuat model aset, lihat [Membuat model aset \(AWS CLI\)](#)
 - Untuk informasi selengkapnya tentang cara memperbarui model aset yang ada, lihat [Memperbarui aset atau model komponen \(AWS CLI\)](#)
2. Tambahkan Lookout for Equipment composite model `assetModelCompositeModels()` ke model aset dengan menambahkan kode berikut.
 - Ganti *Property* dengan ID properti yang ingin Anda sertakan. Untuk mendapatkan ID tersebut, hubungi [DescribeAssetModel](#).
 - Ganti *RoleARN* dengan ARN dari peran IAM yang memungkinkan Lookout for Equipment mengakses data Anda. AWS IoT SiteWise

```
{
  ...
  "assetModelCompositeModels": [
    {
      "name": "L4Epredictiondefinition",
      "type": "AWS/L4E_ANOMALY",
      "properties": [
        {
          "name": "AWS/L4E_ANOMALY_RESULT",
          "dataType": "STRUCT",
          "dataTypeSpec": "AWS/L4E_ANOMALY_RESULT",
```



```

        "unit": "none",
        "type": {
          "measurement": {}
        }
      },
      {
        "name": "AWS/L4E_ANOMALY_INPUT",
        "dataType": "STRUCT",
        "dataTypeSpec": "AWS/L4E_ANOMALY_INPUT",
        "type": {
          "attribute": {
            "defaultValue": "{\"properties\": [\"Property1\",
\\\"Property2\\\"]}"
          }
        }
      },
      {
        "name": "AWS/L4E_ANOMALY_PERMISSIONS",
        "dataType": "STRUCT",
        "dataTypeSpec": "AWS/L4E_ANOMALY_PERMISSIONS",
        "type": {
          "attribute": {
            "defaultValue": "{\"roleArn\": \"RoleARN\"}"
          }
        }
      },
      {
        "name": "AWS/L4E_ANOMALY_DATASET",
        "dataType": "STRUCT",
        "dataTypeSpec": "AWS/L4E_ANOMALY_DATASET",
        "type": {
          "attribute": {}
        }
      },
      {
        "name": "AWS/L4E_ANOMALY_MODEL",
        "dataType": "STRUCT",
        "dataTypeSpec": "AWS/L4E_ANOMALY_MODEL",
        "type": {
          "attribute": {}
        }
      },
      {
        "name": "AWS/L4E_ANOMALY_INFERENCE",

```

```

        "dataType": "STRUCT",
        "dataTypeSpec": "AWS/L4E_ANOMALY_INFERENCE",
        "type": {
            "attribute": {}
        }
    },
    {
        "name": "AWS/L4E_ANOMALY_TRAINING_STATUS",
        "dataType": "STRUCT",
        "dataTypeSpec": "AWS/L4E_ANOMALY_TRAINING_STATUS",
        "type": {
            "attribute": {
                "defaultValue": "{}"
            }
        }
    },
    {
        "name": "AWS/L4E_ANOMALY_INFERENCE_STATUS",
        "dataType": "STRUCT",
        "dataTypeSpec": "AWS/L4E_ANOMALY_INFERENCE_STATUS",
        "type": {
            "attribute": {
                "defaultValue": "{}"
            }
        }
    }
}
]
}

```

3. Buat model aset atau perbarui model aset yang ada. Lakukan salah satu hal berikut:

- Untuk membuat model aset, jalankan perintah berikut:

```
aws iotsitewise create-asset-model --cli-input-json file://asset-model-payload.json
```

- Untuk memperbarui model aset yang ada, jalankan perintah berikut. Ganti *asset-model-id* dengan ID model aset yang ingin Anda perbarui.

```
aws iotsitewise update-asset-model \
  --asset-model-id asset-model-id \
  --cli-input-json file://asset-model-payload.json
```

Setelah Anda menjalankan perintah, perhatikan `assetModelId` dalam respon.

Melatih prediksi dan inferensi awal (CLI)

Sekarang setelah definisi prediksi ditentukan, Anda dapat melatih aset berdasarkan itu dan memulai inferensi. Jika Anda ingin melatih prediksi Anda tetapi tidak memulai inferensi, lompat ke [Pelatihan prediksi \(CLI\)](#) Untuk melatih prediksi dan memulai inferensi pada aset, Anda memerlukan sumber `assetId` daya target.

Untuk melatih dan memulai inferensi prediksi

1. Jalankan perintah berikut untuk menemukan bagian `assetModelCompositeModelId` bawah `assetModelCompositeModelSummaries`. Ganti `asset-model-id` dengan ID model aset yang Anda buat [Memperbarui aset atau model komponen \(AWS CLI\)](#).

```
aws iotsitewise describe-asset-model \  
  --asset-model-id asset-model-id \  
  --asset-model-composite-model-id asset-model-composite-model-id \  
  --output format \  
  --query query
```

2. Jalankan perintah berikut untuk menemukan `TrainingWithInference` tindakan. `actionDefinitionId` Ganti `asset-model-id` dengan ID yang digunakan pada langkah sebelumnya dan ganti `asset-model-composite-model-id` dengan ID yang dikembalikan pada langkah sebelumnya.

```
aws iotsitewise describe-asset-model-composite-model \  
  --asset-model-id asset-model-id \  
  --asset-model-composite-model-id asset-model-composite-model-id \  
  --output format \  
  --query query
```

3. Buat file bernama `train-start-inference-prediction.json` dan tambahkan kode berikut, ganti yang berikut ini:

- `asset-id` dengan ID aset target
- `action-definition-id` dengan ID `TrainingWithInference` tindakan
- `StartTime` dengan dimulainya data pelatihan, disediakan dalam hitungan detik
- `EndTime` dengan berakhirnya data pelatihan, disediakan dalam hitungan detik

```
{  
  "targetResource": {  
    "assetId": asset-id,  
    "actionDefinitionId": action-definition-id,  
    "startTime": start-time,  
    "endTime": end-time  }}
```

```
"assetId": "asset-id"
```

```
},
"actionDefinitionId": "action-definition-Id",
"actionPayload":{
  "stringValue": "{\"14ETrainingWithInference\":{\"trainingWithInferenceMode
\": \"START\", \"trainingPayload\": {\"exportDataStartTime\": StartTime,
\"exportDataEndTime\": EndTime}, \"inferencePayload\": {\"dataDelayOffsetInMinutes
\": 0, \"dataUploadFrequency\": \"PT5M\"}}}"
}
}
```

4. Jalankan perintah berikut untuk memulai pelatihan dan inferensi:

```
aws iotsitewise execute-action --cli-input-json file://train-start-inference-
prediction.json
```

Pelatihan prediksi (CLI)

Sekarang setelah definisi prediksi ditentukan, Anda dapat melatih aset berdasarkan itu. Untuk melatih prediksi aset, Anda memerlukan sumber daya target. `assetId`

Untuk melatih prediksi

1. Jalankan perintah berikut untuk menemukan bagian `assetModelCompositeModelId` bawah `assetModelCompositeModelSummaries`. Ganti *asset-model-id* dengan ID model aset yang Anda buat [Memperbarui aset atau model komponen \(AWS CLI\)](#).

```
aws iotsitewise describe-asset-model \
--asset-model-id asset-model-id \
```

2. Jalankan perintah berikut untuk menemukan Training tindakan. `actionDefinitionId` Ganti *asset-model-id* dengan ID yang digunakan pada langkah sebelumnya dan ganti *asset-model-composite-model-id* dengan ID yang dikembalikan pada langkah sebelumnya.

```
aws iotsitewise describe-asset-model-composite-model \
--asset-model-id asset-model-id \
--asset-model-composite-model-id asset-model-composite-model-id \
```

3. Buat file bernama `train-prediction.json` dan tambahkan kode berikut, ganti yang berikut ini:

- *asset-id* dengan ID aset target
- *action-definition-id* dengan ID tindakan pelatihan
- *StartTime* dengan dimulainya data pelatihan, disediakan dalam hitungan detik
- *EndTime* dengan berakhirnya data pelatihan, disediakan dalam hitungan detik
- (Opsional) *BucketName* dengan nama bucket Amazon S3 yang menyimpan data label Anda
- (Opsional) *Prefix* dengan awalan yang terkait dengan bucket Amazon S3.

 Note

Sertakan nama bucket dan awalan atau keduanya.

```
{
  "targetResource": {
    "assetId": "asset-id"
  },
  "actionDefinitionId": "action-definition-Id",
  "actionPayload": { "stringValue": "{\"14ETraining\": {\"trainingMode\":
  \"START\", \"exportDataStartTime\": StartTime, \"exportDataEndTime\": EndTime,
  \"labelInputConfiguration\": {\"bucketName\": \"BucketName\", \"prefix\":
  \"Prefix\"}}}"
  }
}
```

4. Jalankan perintah berikut untuk memulai pelatihan:

```
aws iotsitewise execute-action --cli-input-json file://train-prediction.json
```

Sebelum Anda dapat memulai inferensi, pelatihan harus diselesaikan. Untuk memeriksa status pelatihan, lakukan salah satu hal berikut:

- Dari konsol, arahkan ke aset tempat prediksi aktif.

- Dari AWS CLI, hubungi BatchGetAssetPropertyValue menggunakan `trainingStatus` properti. `propertyId`

Memulai atau menghentikan inferensi pada prediksi (CLI)

Setelah prediksi dilatih, Anda dapat memulai inferensi untuk memberi tahu Lookout for Equipment untuk mulai memantau aset Anda. Untuk memulai atau menghentikan inferensi, Anda memerlukan `assetId` sumber daya target.

Untuk memulai inferensi

1. Jalankan perintah berikut untuk menemukan bagian `assetModelCompositeModelId` bawah `assetModelCompositeModelSummaries`. Ganti `asset-model-id` dengan ID model aset yang Anda buat [Memperbarui aset atau model komponen \(AWS CLI\)](#).

```
aws iotsitewise describe-asset-model \  
  --asset-model-id asset-model-id \  
  --output json
```

2. Jalankan perintah berikut untuk menemukan Inference tindakan. `actionDefinitionId` Ganti `asset-model-id` dengan ID yang digunakan pada langkah sebelumnya dan ganti `asset-model-composite-model-id` dengan ID yang dikembalikan pada langkah sebelumnya.

```
aws iotsitewise describe-asset-model-composite-model \  
  --asset-model-id asset-model-id \  
  --asset-model-composite-model-id asset-model-composite-model-id \  
  --output json
```

3. Buat file bernama `start-inference.json` dan tambahkan kode berikut, ganti yang berikut ini:

- `asset-id` dengan ID aset target
- `action-definition-id` dengan ID dari tindakan inferensi awal
- `Offset` dengan jumlah buffer yang akan digunakan
- `Frequency` dengan seberapa sering data diunggah

```
{  
  "targetResource": {
```

```
"assetId": "asset-id"
```

```
},
"actionDefinitionId": "action-definition-Id",
"actionPayload":{ "stringValue": "{\\"l4EInference\\": {\\"inferenceMode\\":
\\"START\\",\\"dataDelayOffsetInMinutes\\": Offset, \\"dataUploadFrequency\\":
\\"Frequency\\"}}"
}}
```

4. Jalankan perintah berikut untuk memulai inferensi:

```
aws iotsitewise execute-action --cli-input-json file://start-inference.json
```

Untuk menghentikan inferensi

1. Jalankan perintah berikut untuk menemukan bagian `assetModelCompositeModelId` bawah `assetModelCompositeModelSummaries`. Ganti *asset-model-id* dengan ID model aset yang Anda buat [Memperbarui aset atau model komponen \(AWS CLI\)](#).

```
aws iotsitewise describe-asset-model \
--asset-model-id asset-model-id \
```

2. Jalankan perintah berikut untuk menemukan Inference tindakan. `actionDefinitionId` Ganti *asset-model-id* dengan ID yang digunakan pada langkah sebelumnya dan ganti *asset-model-composite-model-id* dengan ID yang dikembalikan pada langkah sebelumnya.

```
aws iotsitewise describe-asset-model-composite-model \
--asset-model-id asset-model-id \
--asset-model-composite-model-id asset-model-composite-model-id \
```

3. Buat file bernama `stop-inference.json` dan tambahkan kode berikut, ganti yang berikut ini:

- *asset-id* dengan ID aset target
- *action-definition-id* dengan ID dari tindakan inferensi awal

```
{
"targetResource": {
"assetId": "asset-id"
```

```
},
```

```
"actionDefinitionId": "action-definition-Id",  
"actionPayload":{ "stringValue": "{\\"14EInference\\":{\\"inferenceMode\\":\\"STOP\\\\"}}"  
}}  
}}
```

4. Jalankan perintah berikut untuk menghentikan inferensi:

```
aws iotsitewise execute-action --cli-input-json file://stop-inference.json
```

Visualisasikan operasi

Siapkan SiteWise Monitor untuk membuat aplikasi web untuk karyawan operasional Anda. Aplikasi web membantu karyawan untuk memvisualisasikan operasi Anda. Menangani berbagai tingkat akses untuk karyawan Anda menggunakan IAM Identity Center atau IAM. Konfigurasi login dan izin unik untuk setiap karyawan untuk melihat himpunan bagian tertentu dari seluruh operasi industri. AWS IoT SiteWise menyediakan [panduan aplikasi](#) bagi karyawan ini untuk mempelajari cara menggunakan SiteWise Monitor.

Untuk informasi selengkapnya tentang memvisualisasikan operasi Anda, lihat [Memantau data dengan AWS IoT SiteWise Monitor](#)

Menyimpan data

Anda dapat mengintegrasikan penyimpanan deret waktu dengan danau data industri Anda. AWS IoT SiteWise memiliki tiga tingkatan penyimpanan untuk data industri:

- Tingkat penyimpanan panas yang dioptimalkan untuk aplikasi waktu nyata.
- Tingkat penyimpanan hangat yang dioptimalkan untuk beban kerja analitis.
- Tingkat penyimpanan dingin yang dikelola pelanggan menggunakan Amazon S3 untuk aplikasi data operasional dengan toleransi latensi tinggi.

AWS IoT SiteWise membantu Anda mengelola biaya penyimpanan dengan menyimpan data terbaru di tingkat penyimpanan panas. Kemudian, Anda menentukan kebijakan penyimpanan data

untuk memindahkan data historis ke penyimpanan tingkat hangat atau dingin. Untuk informasi selengkapnya, lihat

Anda dapat mengonfigurasi AWS IoT SiteWise untuk menyimpan data Anda di tingkatan penyimpanan berikut:

Tingkat panas

Tingkat penyimpanan panas adalah penyimpanan deret waktu yang AWS IoT SiteWise dikelola. Hot tier paling efektif untuk data yang sering diakses, dengan write-to-read latensi rendah. Data yang disimpan di hot tier digunakan oleh aplikasi industri yang membutuhkan akses cepat ke nilai pengukuran terbaru di peralatan Anda. Ini termasuk aplikasi yang memvisualisasikan metrik real-time dengan dasbor interaktif, atau aplikasi yang memantau operasi dan meluncurkan alarm untuk mengidentifikasi masalah kinerja.

Secara default, data yang tertelan AWS IoT SiteWise disimpan di hot tier. Anda dapat menentukan periode retensi untuk tingkat panas, setelah itu AWS IoT SiteWise memindahkan data di tingkat panas ke penyimpanan tingkat hangat atau dingin, berdasarkan konfigurasi Anda. Untuk kinerja terbaik dan efisiensi biaya, atur periode retensi hot tier Anda menjadi lebih lama dari waktu yang dibutuhkan untuk sering mengambil data. Ini digunakan untuk metrik waktu nyata, alarm, dan skenario pemantauan. Jika periode retensi tidak ditetapkan, data Anda disimpan tanpa batas waktu di tingkat panas.

Tingkat hangat

Tingkat penyimpanan hangat adalah tingkat AWS IoT SiteWise terkelola yang efektif untuk penyimpanan data historis yang hemat biaya. Ini paling baik digunakan untuk mengambil volume besar data dengan karakteristik write-to-read latensi sedang. Gunakan tingkat hangat untuk menyimpan data historis yang diperlukan untuk beban kerja yang besar. Misalnya, ini digunakan untuk pengambilan data untuk analitik, aplikasi intelijen bisnis (BI), alat pelaporan, dan pelatihan model pembelajaran mesin (ML). Jika Anda mengaktifkan tingkat penyimpanan dingin, Anda dapat menentukan periode retensi tingkat hangat. Setelah periode retensi berakhir, AWS IoT SiteWise menghapus data dari tingkat hangat.

Tingkat dingin

Tingkat penyimpanan dingin menggunakan bucket Amazon S3 untuk menyimpan data yang jarang digunakan. Dengan tingkat dingin diaktifkan, AWS IoT SiteWise mereplikasi deret waktu, termasuk pengukuran, metrik, transformasi dan agregat, dan definisi model aset setiap 6 jam. Tingkat dingin digunakan untuk menyimpan data yang mentolerir latensi baca tinggi untuk laporan historis dan cadangan.

Topik

- [Konfigurasi pengaturan penyimpanan](#)
- [Memecahkan masalah pengaturan penyimpanan](#)
- [Jalur file dan skema data yang disimpan di tingkat dingin](#)

Konfigurasi pengaturan penyimpanan

Anda dapat mengonfigurasi pengaturan penyimpanan untuk ikut serta dalam layanan penyimpanan tingkat hangat yang dikelola, dan juga untuk mereplikasi data ke tingkat dingin. Untuk mempelajari lebih lanjut tentang periode retensi untuk tingkat hangat dan panas, lihat [Dampak retensi data](#). Saat mengonfigurasi pengaturan penyimpanan, lakukan hal berikut:

- Retensi tingkat panas — Tetapkan periode retensi untuk berapa lama data Anda disimpan di hot tier sebelum dihapus, dan dipindahkan ke layanan penyimpanan tingkat hangat atau penyimpanan tingkat dingin yang dikelola berdasarkan pengaturan penyimpanan Anda. AWS IoT SiteWise akan menghapus data apa pun di hot tier yang ada sebelum periode retensi berakhir. Jika Anda tidak menetapkan periode retensi, data Anda disimpan tanpa batas waktu di tingkat panas.
- Retensi tingkat hangat — Tetapkan periode retensi untuk berapa lama data Anda disimpan di tingkat hangat sebelum dihapus dari AWS IoT SiteWise penyimpanan dan dipindahkan ke penyimpanan tingkat dingin yang dikelola pelanggan. AWS IoT SiteWise menghapus data apa pun dari tingkat hangat yang ada sebelum periode retensi berakhir. Jika periode retensi tidak diatur, data Anda disimpan tanpa batas waktu di tingkat hangat.

Note

Untuk meningkatkan kinerja kueri, tetapkan periode retensi tingkat panas dengan penyimpanan tingkat hangat.

Dampak retensi data dalam penyimpanan tingkat panas dan hangat

- Saat Anda mengurangi periode retensi penyimpanan tingkat panas, data dipindahkan secara permanen dari tingkat panas ke tingkat hangat atau dingin. Saat Anda mengurangi periode retensi tingkat hangat, data dipindahkan ke tingkat dingin, dan dihapus secara permanen dari tingkat hangat.

- Saat Anda meningkatkan periode retensi penyimpanan tingkat panas atau hangat, perubahan memengaruhi data yang dikirim AWS IoT SiteWise sejak saat itu. AWS IoT SiteWise tidak

mengambil data dari penyimpanan hangat atau dingin untuk mengisi tingkat panas. Misalnya, jika periode retensi penyimpanan hot tier awalnya ditetapkan selama 30 hari dan kemudian ditingkatkan menjadi 60 hari, dibutuhkan 30 hari untuk penyimpanan hot tier berisi data senilai 60 hari.

Topik

- [Konfigurasi pengaturan penyimpanan untuk tingkat hangat \(konsol\)](#)
- [Konfigurasi pengaturan penyimpanan untuk tingkat hangat \(AWS CLI\)](#)
- [Konfigurasi pengaturan penyimpanan untuk tingkat dingin \(konsol\)](#)
- [Konfigurasi pengaturan penyimpanan untuk tingkat dingin \(AWS CLI\)](#)

Konfigurasi pengaturan penyimpanan untuk tingkat hangat (konsol)

Prosedur berikut menunjukkan cara mengonfigurasi pengaturan penyimpanan untuk mereplikasi data ke tingkat hangat di AWS IoT SiteWise konsol.

Untuk mengonfigurasi pengaturan penyimpanan di konsol

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, di bawah Pengaturan, pilih Penyimpanan.
3. Di sudut kanan atas, pilih Edit.
4. Pada halaman Edit penyimpanan, lakukan hal berikut:
5. Untuk pengaturan Hot tier, lakukan hal berikut:
 - Jika Anda ingin menetapkan periode retensi untuk berapa lama data disimpan di hot tier sebelum dihapus, dan dipindahkan ke penyimpanan tingkat hangat yang dikelola layanan, pilih Aktifkan periode retensi.
 - Untuk mengonfigurasi periode retensi, masukkan bilangan bulat dan pilih unit. Periode retensi harus lebih besar dari atau sama dengan 30 hari.

AWS IoT SiteWise menghapus data apa pun di tingkat panas yang lebih lama dari periode retensi. Jika Anda tidak menetapkan periode retensi, data Anda disimpan tanpa batas waktu.

6. (Disarankan) Untuk pengaturan tingkat hangat, lakukan hal berikut:

- Untuk memilih penyimpanan tingkat hangat, pilih Saya mengonfirmasi keikutsertaan penyimpanan tingkat hangat untuk memilih penyimpanan tingkat hangat.
- (Opsional) Untuk mengonfigurasi periode retensi, masukkan bilangan bulat dan pilih unit. Periode retensi harus lebih besar dari atau sama dengan 365 hari.

AWS IoT SiteWise menghapus data di tingkat hangat yang ada lebih awal dari periode retensi. Jika Anda tidak menetapkan periode retensi, data Anda disimpan tanpa batas waktu.

Note

- Saat Anda memilih tingkat hangat, konfigurasi hanya akan ditampilkan sekali.
- Untuk mengatur retensi tingkat panas, Anda harus memiliki penyimpanan tingkat hangat atau dingin. Untuk efisiensi biaya dan pengambilan data historis, AWS IoT SiteWise merekomendasikan agar Anda menyimpan data jangka panjang di tingkat hangat.
- Untuk mengatur retensi tingkat hangat, Anda harus memiliki penyimpanan tingkat dingin.

7. Pilih Simpan untuk menyimpan pengaturan penyimpanan Anda.

Di bagian AWS IoT SiteWise penyimpanan, penyimpanan tingkat hangat berada di salah satu status berikut:

- Diaktifkan — Jika data Anda ada sebelum periode retensi tingkat AWS IoT SiteWise panas, pindahkan data ke tingkat hangat.
- Dinonaktifkan - Penyimpanan tingkat hangat dinonaktifkan.

Konfigurasi pengaturan penyimpanan untuk tingkat hangat (AWS CLI)

Anda dapat mengonfigurasi pengaturan penyimpanan untuk memindahkan data ke tingkat hangat dengan menggunakan perintah AWS CLI dan perintah berikut.

Untuk mencegah penggantian konfigurasi yang ada, ambil informasi konfigurasi penyimpanan saat ini dengan menjalankan perintah berikut:

```
aws iotsitewise describe-storage-configuration
```

Example respons tanpa konfigurasi tingkat dingin yang ada

```
{
  "storageType": "SITEWISE_DEFAULT_STORAGE",
  "disassociatedDataStorage": "ENABLED",
  "configurationStatus": {
    "state": "ACTIVE"
  },
  "lastUpdateDate": "2021-10-14T15:53:35-07:00",
  "warmTier": "DISABLED"
}
```

Example respons dengan konfigurasi tingkat dingin yang ada

```
{
  "storageType": "MULTI_LAYER_STORAGE",
  "multiLayerStorage": {
    "customerManagedS3Storage": {
      "s3ResourceArn": "arn:aws:s3:::bucket-name/prefix/",
      "roleArn": "arn:aws:iam::aws-account-id:role/role-name"
    }
  },
  "disassociatedDataStorage": "ENABLED",
```

Example AWS IoT SiteWise konfigurasi dengan tingkat panas dan hangat

```
{
  "storageType": "SITEWISE_DEFAULT_STORAGE",
  "disassociatedDataStorage": "ENABLED",
  "warmTier": "ENABLED",
  "retentionPeriod": {
    "numberOfDays": hot-tier-retention-in-days
  }
}
```

hot-tier-retention-in-day harus bilangan bulat yang lebih besar dari atau sama dengan 30 hari.

Example response

```
{
  "storageType": "SITEWISE_DEFAULT_STORAGE",
  "configurationStatus": {
    "state": "UPDATE_IN_PROGRESS"
  }
}
```

Jika Anda mengaktifkan penyimpanan tingkat dingin, lihat [Konfigurasi pengaturan penyimpanan dengan AWS CLI dan tingkat dingin yang ada](#).

Konfigurasi pengaturan penyimpanan dengan AWS CLI dan tingkat dingin yang ada

Konfigurasi pengaturan penyimpanan untuk tingkat hangat (AWS CLI)

Konfigurasi pengaturan penyimpanan menggunakan AWS CLI penyimpanan tingkat dingin yang ada

- Ganti *awalan* dengan *awalan* Amazon S3 Anda.
- Ganti *aws-account-id* dengan ID AWS akun Anda.
- Ganti *role-name* dengan nama peran akses Amazon S3 yang AWS IoT SiteWise memungkinkan untuk mengirim data ke Amazon S3.
- Ganti *hot-tier-retention-in-hari* dengan bilangan bulat yang lebih besar dari atau sama dengan 30 hari.
- Ganti *warm-tier-retention-in-hari* dengan bilangan bulat yang lebih besar dari atau sama dengan 365 hari.

Note

AWS IoT SiteWise akan menghapus data apa pun di tingkat hangat yang lebih tua dari periode retensi tingkat dingin. Jika Anda tidak menetapkan periode retensi, data Anda disimpan tanpa batas waktu.

```
{
  "storageType": "MULTI_LAYER_STORAGE",
  "multiLayerStorage": {
    "customerManagedS3Storage": {
      "s3ResourceArn": "arn:aws:s3:::bucket-name/prefix/",
      "roleArn": "arn:aws:iam::aws-account-id:role/role-name"
    }
  },
  "disassociatedDataStorage": "ENABLED",
  "retentionPeriod": {
    "numberOfDays": hot-tier-retention-in-days
  },
  "warmTier": "ENABLED",
  "warmTierRetentionPeriod": {
    "numberOfDays": warm-tier-retention-in-days
  }
}
```

Example response

```
{
  "storageType": "MULTI_LAYER_STORAGE",
```

```
"configurationStatus": {
```

```
  "state": "UPDATE_IN_PROGRESS"
```

```
  }
```

```
}
```

Konfigurasi pengaturan penyimpanan untuk tingkat dingin (konsol)

Prosedur berikut menunjukkan cara mengonfigurasi pengaturan penyimpanan untuk mereplikasi data ke tingkat dingin di AWS IoT SiteWise konsol.

Untuk mengkonfigurasi pengaturan penyimpanan di konsol

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, di bawah Pengaturan, pilih Penyimpanan.
3. Di sudut kanan atas, pilih Edit.
4. Pada halaman Edit penyimpanan, lakukan hal berikut:
 - a. Untuk pengaturan Penyimpanan, pilih Aktifkan penyimpanan tingkat dingin. Penyimpanan tingkat dingin dinonaktifkan secara default.
 - b. Untuk lokasi bucket S3, masukkan nama bucket Amazon S3 yang ada dan awalan.

Note

- Amazon S3 menggunakan awalan sebagai nama folder di bucket Amazon S3. Awalan harus memiliki 1-255 karakter dan diakhiri dengan garis miring (/). AWS IoT SiteWise Data Anda disimpan di folder ini.
- Jika Anda tidak memiliki bucket Amazon S3, pilih View, lalu buat bucket di konsol Amazon S3. Untuk informasi selengkapnya, lihat [Membuat bucket S3 pertama Anda](#) di Panduan Pengguna Amazon S3.

- c. Untuk peran akses S3, lakukan salah satu hal berikut:
 - Pilih Buat peran dari templat AWS terkelola, AWS secara otomatis membuat peran IAM yang memungkinkan AWS IoT SiteWise untuk mengirim data ke Amazon S3.
 - Pilih Gunakan peran yang ada, lalu pilih peran yang Anda buat dari daftar.

Note

- Anda harus menggunakan nama bucket Amazon S3 yang sama untuk lokasi bucket S3 yang Anda gunakan pada langkah sebelumnya dan dalam kebijakan IAM Anda.
- Pastikan bahwa peran Anda memiliki izin yang ditunjukkan dalam contoh berikut.

Example kebijakan izin:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject",
        "s3:GetBucketLocation",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::bucket-name",
        "arn:aws:s3:::bucket-name/*"
      ]
    }
  ]
}
```

Ganti nama *ember* dengan nama *bucket* Amazon S3 Anda.

- d. Untuk mengatur hot tier, lihat Langkah 5 di [Konfigurasi pengaturan penyimpanan untuk tingkat hangat \(konsol\)](#).
- e. (Opsional) Untuk AWS IoT Analytics integrasi, lakukan hal berikut.
 - i. Jika Anda AWS IoT Analytics ingin menggunakan kueri data Anda, pilih Penyimpanan AWS IoT Analytics data yang diaktifkan.

- ii. AWS IoT SiteWise menghasilkan nama untuk penyimpanan data Anda atau Anda dapat memasukkan nama yang berbeda.

AWS IoT SiteWise secara otomatis membuat penyimpanan data AWS IoT Analytics untuk menyimpan data Anda. Untuk menanyakan data, Anda dapat menggunakan AWS IoT Analytics untuk membuat kumpulan data. Untuk informasi selengkapnya, lihat [Bekerja dengan AWS IoT SiteWise data](#) di Panduan AWS IoT Analytics Pengguna.

f. Pilih Simpan.

Di bagian AWS IoT SiteWise penyimpanan, penyimpanan tingkat dingin dapat menjadi salah satu dari nilai berikut:

- Diaktifkan - AWS IoT SiteWise mereplikasi data Anda ke bucket Amazon S3 yang ditentukan.
- Mengaktifkan — AWS IoT SiteWise memproses permintaan Anda untuk mengaktifkan penyimpanan tingkat dingin. Proses ini bisa memakan waktu beberapa menit untuk menyelesaikannya.
- Enable_Failed — AWS IoT SiteWise tidak dapat memproses permintaan Anda untuk mengaktifkan penyimpanan tingkat dingin. Jika Anda mengaktifkan AWS IoT SiteWise untuk mengirim log ke Amazon CloudWatch Logs, Anda dapat menggunakan log ini untuk memecahkan masalah. Untuk informasi selengkapnya, lihat [Pemantauan dengan Amazon CloudWatch Logs](#).
- Dinonaktifkan - Penyimpanan tingkat dingin dinonaktifkan.

Konfigurasi pengaturan penyimpanan untuk tingkat dingin (AWS CLI)

Prosedur berikut menunjukkan cara mengkonfigurasi pengaturan penyimpanan untuk mereplikasi data ke tingkat dingin menggunakan AWS CLI.

Untuk mengkonfigurasi pengaturan penyimpanan menggunakan AWS CLI

1. Untuk mengekspor data ke bucket Amazon S3 di akun Anda, jalankan perintah berikut untuk mengonfigurasi pengaturan penyimpanan. Ganti *nama file* dengan nama file yang berisi konfigurasi AWS IoT SiteWise penyimpanan.

```
aws iotsitewise put-storage-configuration --cli-input-json file://file-name.json
```

Example AWS IoT SiteWise konfigurasi penyimpanan

- Ganti nama *ember* dengan nama *bucket* Amazon S3 Anda.
- Ganti *awalan* dengan *awalan* Amazon S3 Anda.
- Ganti *aws-account-id* dengan ID AWS akun Anda.
- Ganti *role-name* dengan nama peran akses Amazon S3 yang AWS IoT SiteWise memungkinkan untuk mengirim data ke Amazon S3.
- Ganti *retention-in-days* dengan bilangan bulat lebih besar dari atau sama dengan 30 hari.

```
{
  "storageType": "MULTI_LAYER_STORAGE",
  "multiLayerStorage": {
    "customerManagedS3Storage": {
      "s3ResourceArn": "arn:aws:s3::bucket-name/prefix/",
      "roleArn": "arn:aws:iam::aws-account-id:role/role-name"
    }
  },
  "retentionPeriod": {
    "numberOfDays": retention-in-days,
    "unlimited": false
  }
}
```

Note

- Anda harus menggunakan nama bucket Amazon S3 yang sama dalam konfigurasi AWS IoT SiteWise penyimpanan dan kebijakan IAM.
- Pastikan bahwa peran Anda memiliki izin yang ditunjukkan dalam contoh berikut.
Example kebijakan izin:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
```

```

        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject",
        "s3:GetBucketLocation",
        "s3:ListBucket"
    ],
    "Resource": [
        "arn:aws:s3:::bucket-name",
        "arn:aws:s3:::bucket-name/*"
    ]
}
]
}

```

Ganti nama *ember* dengan nama *bucket* Amazon S3 Anda.

Example response

```

{
  "storageType": "MULTI_LAYER_STORAGE",
  "retentionPeriod": {
    "numberOfDays": 100,
    "unlimited": false
  },
  "configurationStatus": {
    "state": "UPDATE_IN_PROGRESS"
  }
}

```

Note

Diperlukan beberapa menit AWS IoT SiteWise untuk memperbarui konfigurasi penyimpanan.

2. Untuk mengambil informasi konfigurasi penyimpanan, jalankan perintah berikut.

```
aws iotsitewise describe-storage-configuration
```

Example response

```
{
  "storageType": "MULTI_LAYER_STORAGE",
  "multiLayerStorage": {
    "customerManagedS3Storage": {
      "s3ResourceArn": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/torque/",
      "roleArn": "arn:aws:iam::123456789012:role/SWAccessS3Role"
    }
  },
  "retentionPeriod": {
    "numberOfDays": 100,
    "unlimited": false
  },
  "configurationStatus": {
    "state": "ACTIVE"
  },
  "lastUpdateDate": "2021-03-30T15:54:14-07:00"
}
```

3. Untuk berhenti mengeksport data ke bucket Amazon S3, jalankan perintah berikut untuk mengonfigurasi pengaturan penyimpanan.

```
aws iotsitewise put-storage-configuration --storage-type SITEWISE_DEFAULT_STORAGE
```

Note

Secara default, data Anda hanya disimpan di tingkat panas AWS IoT SiteWise.

Example response

```
{
  "storageType": "SITEWISE_DEFAULT_STORAGE",
  "configurationStatus": {
    "state": "UPDATE_IN_PROGRESS"
  }
}
```

4. Untuk mengambil informasi konfigurasi penyimpanan, jalankan perintah berikut.

```
aws iotsitewise describe-storage-configuration
```

Example response

```
{
  "storageType": "SITEWISE_DEFAULT_STORAGE",
  "configurationStatus": {
    "state": "ACTIVE"
  },
  "lastUpdateDate": "2021-03-30T15:57:14-07:00"
}
```

(Opsional) Buat penyimpanan AWS IoT Analytics data (AWS CLI)

Penyimpanan AWS IoT Analytics data adalah repositori yang dapat diskalakan dan dapat dikueri yang menerima dan menyimpan data. Anda dapat menggunakan AWS IoT SiteWise konsol atau AWS IoT Analytics API untuk membuat penyimpanan AWS IoT Analytics data untuk menyimpan AWS IoT SiteWise data Anda. Untuk menanyakan data, Anda membuat kumpulan data dengan menggunakan [AWS IoT Analytics](#). Untuk informasi selengkapnya, lihat [Bekerja dengan AWS IoT SiteWise data](#) di Panduan AWS IoT Analytics Pengguna.

Langkah-langkah berikut digunakan AWS CLI untuk membuat penyimpanan data di AWS IoT Analytics.

Untuk membuat penyimpanan data, jalankan perintah berikut. Ganti *nama file* dengan nama file yang berisi konfigurasi penyimpanan data.

```
aws iotanalytics create-datastore --cli-input-json file://file-name.json
```

Note

- Anda harus menentukan nama bucket Amazon S3 yang ada. Jika Anda tidak memiliki ember Amazon S3, buat ember terlebih dahulu. Untuk informasi selengkapnya, lihat [Membuat bucket S3 pertama Anda](#) di Panduan Pengguna Amazon S3.

- Anda harus menggunakan nama bucket Amazon S3 yang sama dalam konfigurasi AWS IoT SiteWise penyimpanan, kebijakan IAM, dan konfigurasi penyimpanan AWS IoT Analytics data.

Example AWS IoT Analytics konfigurasi penyimpanan data

Ganti *data-store-name* dan *s3-bucket-name* dengan nama penyimpanan data dan AWS IoT Analytics nama bucket Amazon S3 Anda.

```
{
  "datastoreName": "data-store-name",
  "datastoreStorage": {
    "iotSiteWiseMultiLayerStorage": {
      "customerManagedS3Storage": {
        "bucket": "s3-bucket-name"
      }
    }
  },
  "retentionPeriod": {
    "numberOfDays": 90
  }
}
```

Example response

```
{
  "datastoreName": "datastore_IoTSiteWise_demo",
  "datastoreArn": "arn:aws:iotanalytics:us-west-2:123456789012:datastore/
  datastore_IoTSiteWise_demo",
```

Masalah

- [Kesalahan: Bucket tidak ada](#)
- [Kesalahan: Akses ditolak ke jalur Amazon S3](#)
- [Kesalahan: Peran ARN tidak dapat diasumsikan](#)
- [Kesalahan: Gagal mengakses bucket Amazon S3 lintas wilayah](#)

Kesalahan: Bucket tidak ada

Solusi: AWS IoT SiteWise tidak dapat menemukan ember Amazon S3 Anda. Pastikan Anda memasukkan nama bucket Amazon S3 yang ada di Wilayah saat ini.

Kesalahan: Akses ditolak ke jalur Amazon S3

Solusi: AWS IoT SiteWise tidak dapat mengakses bucket Amazon S3 Anda. Lakukan hal-hal berikut:

- Pastikan Anda menggunakan bucket Amazon S3 yang sama dengan yang Anda tentukan dalam kebijakan IAM.
- Pastikan bahwa peran Anda memiliki izin yang ditunjukkan dalam contoh berikut.

Example kebijakan izin

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject",
        "s3:GetBucketLocation",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::bucket-name",
        "arn:aws:s3:::bucket-name/*"
      ]
    }
  ]
}
```



```
}
```

Ganti nama *ember* dengan nama *bucket* Amazon S3 Anda.

Kesalahan: Peran ARN tidak dapat diasumsikan

Solusi: AWS IoT SiteWise tidak dapat mengambil peran IAM atas nama Anda. Pastikan peran Anda mempercayai layanan berikut: `iotsitewise.amazonaws.com`. Untuk informasi selengkapnya, lihat [Saya tidak dapat mengambil peran](#), lihat Panduan Pengguna IAM.

Kesalahan: Gagal mengakses bucket Amazon S3 lintas wilayah

Solusi: Bucket Amazon S3 yang Anda tentukan berada di Wilayah yang berbeda AWS . Pastikan bucket dan AWS IoT SiteWise aset Amazon S3 Anda berada di Wilayah yang sama.

Jalur file dan skema data yang disimpan di tingkat dingin

AWS IoT SiteWise menyimpan data Anda di tingkat dingin dengan mereplikasi deret waktu, termasuk pengukuran, metrik, transformasi dan agregat, serta definisi model aset dan aset. Berikut ini menjelaskan jalur file dan skema data yang dikirim ke tingkat dingin.

Topik

- [Data peralatan \(pengukuran\)](#)
- [Metrik, transformasi, dan agregat](#)
- [Metadata aset](#)
- [Metadata hierarki aset](#)
- [File indeks data penyimpanan](#)

Data peralatan (pengukuran)

AWS IoT SiteWise mengekspor data peralatan (pengukuran) ke tingkat dingin setiap enam jam sekali. Data mentah disimpan dalam tingkat dingin dalam format [Apache AVRO](#) (`.avro`).

Jalur berkas

AWS IoT SiteWise menyimpan data peralatan (pengukuran) di tingkat dingin menggunakan templat berikut.

```
{keyPrefix}/raw/startYear={startYear}/startMonth={startMonth}/startDay={startDay}/
```

```
seriesBucket={seriesBucket}/raw_{timeseriesId}_{startTimestamp}_{quality}.avro
```

Setiap path file ke data mentah di Amazon S3 berisi komponen-komponen berikut.

<code>keyPrefix</code>	Awalan Amazon S3 yang Anda tentukan dalam konfigurasi penyimpanan. AWS IoT SiteWise Amazon S3 menggunakan awalan sebagai nama folder di bucket.
<code>raw</code>	Folder yang menyimpan data deret waktu dari peralatan (pengukuran). <code>rawFolder</code> disimpan di folder awalan.
<code>seriesBucket</code>	Angka heksadesimal antara 00 dan ff. Angka ini berasal dari <code>timeSeriesId</code> . Partisi ini digunakan untuk meningkatkan throughput saat AWS IoT SiteWise menulis ke tingkat dingin. Saat Anda menggunakan Amazon Athena untuk menjalankan kueri, Anda dapat menggunakan partisi untuk partisi butir halus guna meningkatkan kinerja kueri. <code>seriesBucket</code> dan <code>timeSeriesBucket</code> dalam metadata aset adalah angka yang sama.
<code>startYear</code>	Tahun waktu mulai eksklusif yang terkait dengan data deret waktu.
<code>startMonth</code>	Bulan waktu mulai eksklusif yang terkait dengan data deret waktu.
<code>startDay</code>	Hari bulan dari waktu mulai eksklusif yang terkait dengan data deret waktu.

<code>fileName</code>	Nama file menggunakan karakter garis bawah (<code>_</code>) sebagai pembatas untuk memisahkan yang berikut:
	<ul style="list-style-type: none"> <code>rawAwalan</code>. <code>timeSeriesId</code> Nilainya. Stempel waktu zaman dari waktu mulai eksklusif yang terkait dengan data deret waktu. Kualitas data. Nilai-nilai yang valid: GOOD, BAD, dan UNCERTAIN . Untuk informasi selengkapnya, lihat AssetPropertyValuedi Referensi AWS IoT SiteWise API. <p>File disimpan dalam <code>.avro</code> format dengan menggunakan kompresi Snappy.</p>

Komponen jalur	Deskripsi
----------------	-----------

Example jalur file ke data mentah di tingkat dingin

```
keyPrefix/raw/startYear=2021/startMonth=1/startDay=2/seriesBucket=a2/
raw_7020c8e2-e6db-40fa-9845-ed0ddddd4c77d_95e63da7-d34e-43e1-
bc6f-1b490154b07a_1609577700_GOOD.avro
```

Bidang

Skema data mentah yang diekspor ke tingkat dingin berisi bidang-bidang berikut.

<code>seriesId</code>	<code>string</code>	N/A	ID yang mengidentifikasi data deret waktu dari peralatan (pengukuran). Anda dapat menggunakan

			an bidang ini untuk menggabungkan data mentah dan metadata aset dalam kueri.
<code>timeInSeconds</code>	<code>long</code>	N/A	Tanggal stempel waktu, dalam hitungan detik, dalam format epoch Unix. Data nanodetik pecahan disediakan oleh. <code>offsetInNanos</code>
<code>offsetInNanos</code>	<code>long</code>	N/A	Nanodetik offset dari. <code>timeInSeconds</code>
<code>quality</code>	<code>string</code>	N/A	Kualitas nilai deret waktu.
<code>doubleValue</code>	<code>double</code> atau <code>null</code>	<code>null</code>	Data deret waktu tipe ganda (nomor floating point).
<code>stringValue</code>	<code>string</code> atau <code>null</code>	<code>null</code>	Data deret waktu dari tipe string (urutan karakter).
<code>integerValue</code>	<code>int</code> atau <code>null</code>	<code>null</code>	Data deret waktu tipe integer (bilangan bulat).
<code>booleanValue</code>	<code>boolean</code> atau <code>null</code>	<code>null</code>	Data deret waktu tipe Boolean (benar atau salah).

<code>jsonValue</code>	string atau null	null	Data deret waktu tipe JSON (tipe
			data kompleks yang disimpan sebagai string).
<code>recordVersion</code>	long atau null	null	Nomor versi untuk catatan. Anda dapat menggunakan nomor versi untuk memilih catatan terbaru. Catatan yang lebih baru memiliki nomor versi yang lebih besar.
Nama bidang	Jenis yang didukung	Jenis default	Deskripsi

Example data mentah di tingkat dingin

<pre>{"seriesId":"e9687d2a-0dbe-4f65-9ed6-6f443cba41f7_95e63da7-d34e-43e1-bc6f-1b490154b07a","timeInSeconds":1625675887,"offsetInNanos":0,"quality":"GOOD","doubleValue":0.75},"stringValue":null,"integerValue":null,"booleanValue":null,"jsonValue":null,"recordVersion":1}</pre>
<pre>{"seriesId":"e9687d2a-0dbe-4f65-9ed6-6f443cba41f7_95e63da7-d34e-43e1-bc6f-1b490154b07a","timeInSeconds":1625675889,"offsetInNanos":0,"quality":"GOOD","doubleValue":0.69},"stringValue":null,"integerValue":null,"booleanValue":null,"jsonValue":null,"recordVersion":2}</pre>
<pre>{"seriesId":"e9687d2a-0dbe-4f65-9ed6-6f443cba41f7_95e63da7-d34e-43e1-bc6f-1b490154b07a","timeInSeconds":1625675890,"offsetInNanos":0,"quality":"GOOD","doubleValue":0.66},"stringValue":null,"integerValue":null,"booleanValue":null,"jsonValue":null,"recordVersion":3}</pre>
<pre>{"seriesId":"e9687d2a-0dbe-4f65-9ed6-6f443cba41f7_95e63da7-d34e-43e1-bc6f-1b490154b07a","timeInSeconds":1625675891,"offsetInNanos":0,"quality":"GOOD","doubleValue":0.92},"stringValue":null,"integerValue":null,"booleanValue":null,"jsonValue":null,"recordVersion":4}</pre>

Metrik, transformasi, dan agregat

AWS IoT SiteWise mengekspor metrik, transformasi, dan agregat ke tingkat dingin setiap enam jam sekali. Metrik, transformasi, dan agregat disimpan dalam tingkat dingin dalam format [Apache AVRO](#) (). `.avro`

Jalur berkas

AWS IoT SiteWise menyimpan metrik, transformasi, dan agregat di tingkat dingin menggunakan templat berikut.

```
{keyPrefix}/agg/startYear={startYear}/startMonth={startMonth}/startDay={startDay}/
```

```
seriesBucket={seriesBucket}/agg_{timeseriesId}_{startTimestamp}_{quality}.avro
```

Setiap jalur file ke metrik, transformasi, dan agregat di Amazon S3 berisi komponen berikut.

<code>keyPrefix</code>	Awalan Amazon S3 yang Anda tentukan dalam konfigurasi penyimpanan. AWS IoT SiteWise Amazon S3 menggunakan awalan sebagai nama folder di bucket.
<code>agg</code>	Folder yang menyimpan data deret waktu dari metrik. <code>aggFolder</code> disimpan di folder awalan.
<code>seriesBucket</code>	Angka heksadesimal antara 00 dan ff. Angka ini berasal dari <code>timeSeriesId</code> . Partisi ini digunakan untuk meningkatkan throughput saat AWS IoT SiteWise menulis ke tingkat dingin. Saat Anda menggunakan Amazon Athena untuk menjalankan kueri, Anda dapat menggunakan partisi untuk partisi butir halus guna meningkatkan kinerja kueri. <code>seriesBucket</code> dan <code>timeSeriesBucket</code> dalam metadata aset adalah angka yang sama.

<code>startYear</code>	Tahun waktu mulai eksklusif yang terkait dengan data deret waktu.
<code>startMonth</code>	Bulan waktu mulai eksklusif yang terkait dengan data deret waktu.
<code>startDay</code>	Hari bulan dari waktu mulai eksklusif yang terkait dengan data deret waktu.
<code>fileName</code>	<p>Nama file menggunakan karakter garis bawah (<code>_</code>) sebagai pembatas untuk memisahkan yang berikut:</p> <ul style="list-style-type: none"> <code>rawAwalan</code>. <code>timeSeriesId</code> Nilainya. Stempel waktu zaman dari waktu mulai eksklusif yang terkait dengan data deret waktu. Kualitas data. Nilai-nilai yang valid: <code>GOOD</code>, <code>BAD</code>, dan <code>UNCERTAIN</code> . Untuk informasi selengkapnya, lihat AssetPropertyValuedi Referensi AWS IoT SiteWise API. <p>File disimpan dalam <code>.avro</code> format dengan menggunakan kompresi Snappy.</p>

Komponen jalur	Deskripsi
----------------	-----------

Example jalur file ke metrik di tingkat dingin

```
keyPrefix/agg/startYear=2021/startMonth=1/startDay=2/seriesBucket=a2/agg_7020c8e2-e6db-40fa-9845-ed0dddd4c77d_95e63da7-d34e-43e1-bc6f-1b490154b07a_1609577700_GOOD.avro
```

Bidang

Skema metrik, transformasi, dan agregat yang diekspor ke tingkat dingin berisi bidang berikut.

<code>seriesId</code>	<code>string</code>	N/A	ID yang mengidentifikasi data deret waktu dari peralatan, metrik, atau transformasi. Anda dapat menggunakan bidang ini untuk menggabungkan data mentah dan metadata aset dalam kueri.
<code>timeInSeconds</code>	<code>long</code>	N/A	Tanggal stempel waktu, dalam hitungan detik, dalam format epoch Unix. Data nanodetik pecahan disediakan oleh <code>offsetInNanos</code>
<code>offsetInNanos</code>	<code>long</code>	N/A	Nanodetik offset dari <code>timeInSeconds</code>
<code>quality</code>	<code>string</code>	N/A	Kualitas yang digunakan untuk memfilter data aset.
<code>resolution</code>	<code>string</code>	N/A	Interval waktu untuk mengumpulkan data.
<code>count</code>	<code>double</code> atau <code>null</code>	<code>null</code>	Jumlah total titik data untuk variabel yang diberikan selama interval waktu saat ini.

average	double atau null	null	Rata-rata nilai variabel yang diberikan selama interval waktu saat ini.
min	double atau null	null	Minimum nilai variabel yang diberikan selama interval waktu saat ini.
max	boolean atau null	null	Maksimum nilai variabel yang diberikan selama interval waktu saat ini.
sum	string atau null	null	Jumlah nilai variabel yang diberikan selama interval waktu saat ini.
recordVersion	long atau null	null	Nomor versi untuk catatan. Anda dapat menggunakan nomor versi untuk memilih catatan terbaru. Catatan yang lebih baru memiliki nomor versi yang lebih besar.
Nama bidang	Jenis yang didukung	Jenis default	Deskripsi

Example Data metrik di tingkat dingin

```
{"seriesId":"f74c2828-5317-4df3-
```

```
ba16-6d41b5bcb531","timeInSeconds":1637334060,"offsetInNanos":0,"quality":"GOOD","resolution":
```

```
{"double":16.0},"min":{"double":1.0},"max":{"double":31.0},"sum":
```

```
{"double":496.0},"recordVersion":null}
```

```
{"seriesId":"f74c2828-5317-4df3-
```

```
ba16-6d41b5bcb531","timeInSeconds":1637334120,"offsetInNanos":0,"quality":"GOOD","resolution":"
```

```
{"double":46.0},"min":{"double":32.0},"max":{"double":60.0},"sum":
```

```
{"double":1334.0},"recordVersion":null}
```

```
{"seriesId":"f74c2828-5317-4df3-
```

```
ba16-6d41b5bcb531","timeInSeconds":1637334540,"offsetInNanos":0,"quality":"GOOD","resolution":"
```

```
{"double":16.0},"min":{"double":1.0},"max":{"double":31.0},"sum":
```

```
{"double":496.0},"recordVersion":null}
```

```
{"seriesId":"f74c2828-5317-4df3-
```

```
ba16-6d41b5bcb531","timeInSeconds":1637334600,"offsetInNanos":0,"quality":"GOOD","resolution":"
```

```
{"double":46.0},"min":{"double":32.0},"max":{"double":60.0},"sum":
```

```
{"double":1334.0},"recordVersion":null}
```

```
{"seriesId":"f74c2828-5317-4df3-
```

```
ba16-6d41b5bcb531","timeInSeconds":1637335020,"offsetInNanos":0,"quality":"GOOD","resolution":"
```

```
{"double":16.0},"min":{"double":1.0},"max":{"double":31.0},"sum":
```

```
{"double":496.0},"recordVersion":null}
```

Metadata aset

Saat Anda mengaktifkan AWS IoT SiteWise untuk mengekspor data ke tingkat dingin untuk pertama kalinya, metadata aset diekspor ke tingkat dingin. Setelah konfigurasi awal, AWS IoT SiteWise ekspor metadata aset ke tingkat hanya jika Anda mengubah definisi model aset atau definisi aset. Metadata aset disimpan dalam tingkat dingin dalam format JSON () yang dibatasi baris baru. .ndjson

Jalur berkas

AWS IoT SiteWise menyimpan metadata aset di tingkat dingin menggunakan templat berikut.

Metadata aset

```
{keyPrefix}/asset_metadata/asset_{assetId}.ndjson
```

46

Setiap jalur file ke metadata aset di tingkat dingin berisi komponen berikut.

<code>asset_metadata</code>	Folder yang menyimpan metadata aset. <code>asset_metadata</code> Folder disimpan di folder awalan.
<code>fileName</code>	Nama file menggunakan karakter garis bawah (<code>_</code>) sebagai pembatas untuk memisahkan yang berikut: <ul style="list-style-type: none"> <code>assetAwalan</code>. <code>assetIdNilainya</code>. File disimpan dalam <code>.ndjson</code> format.
Komponen jalur	Deskripsi

Example jalur file ke metadata aset di tingkat yang lebih dingin

`keyPrefix/asset_metadata/asset_35901915-d476-4dca-8637-d9ed4df939ed.ndjson`

Bidang

Skema metadata aset yang diekspor ke tingkat dingin berisi bidang-bidang berikut.

<code>assetId</code>	ID aset.
<code>assetName</code>	Nama aset.
<code>assetExternalId</code>	ID eksternal aset.
<code>assetModelId</code>	ID model aset yang digunakan untuk membuat aset ini.
<code>assetModelName</code>	Nama model aset.
<code>assetModelExternalId</code>	ID eksternal dari model aset.
<code>assetPropertyId</code>	ID dari properti aset.

<code>assetPropertyName</code>	Nama properti aset.
<code>assetPropertyExternalId</code>	ID eksternal dari properti aset.
<code>assetPropertyDataType</code>	Tipe data properti aset.
<code>assetPropertyUnit</code>	Unit properti aset (misalnya, Newtons danRPM).
<code>assetPropertyAlias</code>	Alias yang mengidentifikasi properti aset, seperti jalur aliran data server OPC-UA (misalnya,). <code>/company/windfarm/3/turbine/7/temperature</code>
<code>timeSeriesId</code>	ID yang mengidentifikasi data deret waktu dari peralatan, metrik, atau transformasi. Anda dapat menggunakan bidang ini untuk menggabungkan data mentah dan metadata aset dalam kueri.
<code>timeSeriesBucket</code>	Angka heksadesimal antara 00 dan ff. Angka ini berasal dari <code>timeSeriesId</code> . Partisi ini digunakan untuk meningkatkan throughput saat AWS IoT SiteWise menulis ke tingkat dingin. Saat Anda menggunakan Amazon Athena untuk menjalankan kueri, Anda dapat menggunakan partisi untuk partisi butir halus guna meningkatkan kinerja kueri. <code>timeSeriesBucket</code> dan <code>seriesBucket</code> di jalur file ke data mentah adalah nomor yang sama.
<code>assetCompositeModelId</code>	ID dari model komposit.
<code>assetCompositeModelExternalId</code>	ID eksternal dari model komposit.

<code>assetCompositeModelDescription</code>	Deskripsi dari model komposit.
<code>assetCompositeModelName</code>	Nama dari model komposit.
<code>assetCompositeModelType</code>	Jenis dari model komposit. Untuk model komposit alarm, jenis ini berupa <code>AWS/ALARM</code> .
<code>assetCreationDate</code>	Tanggal aset dibuat, dalam waktu zaman Unix.
<code>assetLastUpdateDate</code>	Tanggal aset terakhir diperbarui, dalam waktu zaman Unix.
<code>assetStatusErrorCode</code>	Kode kesalahan.
<code>assetStatusErrorMessage</code>	Pesan kesalahan.
<code>assetStatusState</code>	Status aset saat ini.
Nama bidang	Deskripsi

Example metadata aset di tingkat dingin

```

{"assetId":"7020c8e2-e6db-40fa-9845-ed0dddd4c77d",
"assetExternalId":null,
"assetName":"Wind Turbine Asset 2",
"assetModelId":"ec1d924f-f07d-444f-b072-e2994c165d35",
"assetModelExternalId":null,
"assetModelName":"Wind Turbine Asset Model",
"assetPropertyId":"95e63da7-d34e-43e1-bc6f-1b490154b07a",
"assetPropertyExternalId":null,
"assetPropertyName":"Temperature",
"assetPropertyDescription":"Washington/Seattle/WT2/temp",
"timeSeriesId":"7020c8e2-e6db-40fa-9845-ed0dddd4c77d_95e63da7-d34e-43e1-bc6f-1b490154b07a",
"timeSeriesBucket":"f6",
"assetArn":null,
"assetCompositeModelDescription":null,
"assetCompositeModelName":null,
"assetCompositeModelType":null,
"assetCreationDate":1546300800,
"assetLastUpdateDate":1546300800,
"assetStatusErrorCode":null,
"assetStatusErrorMessage":null,
"assetStatusState":"OK"}

```

```
2", "assetModelId": "ec1d924f-f07d-444f-b072-
```

```
e2994c165d35", "assetModelExternalId": null, "assetModelName": "Wind
```

```
Turbine Asset Model", "assetPropertyId": "8cf1162f-dead-4fbe-b468-
```

```
c8e24cde9f50", "assetPropertyExternalId": null, "assetPropertyName": "Max
```

```
Temperature", "assetPropertyDataType": "DOUBLE", "assetPropertyUnit": null, "assetPropertyAlias": null,
```

```
e6db-40fa-9845-ed0ddd4c77d_8cf1162f-dead-4fbe-b468-
```

```
c8e24cde9f50", "timeSeriesBucket": "d7", "assetArn": null, "assetCompositeModelDescription": null, "assetPr
```

```
{"assetId": "3a5f2a22-3b37-4332-9c1c-404ea1d73fab", "assetExternalId": null, "assetName": "BatchAss
```

```
ebc75e75e827", "assetModelExternalId": null, "assetModelName": "FlashTestAssetModelDouble", "assetPr
```

```
b410-
```

```
ab401a9176ed", "assetPropertyExternalId": null, "assetPropertyName": "measurementProperty", "assetPr
```

```
ae89-
```

```
ff316f5ff8aa", "timeSeriesBucket": "af", "assetArn": null, "assetCompositeModelDescription": null, "as
```

Metadata hierarki aset

Saat Anda mengaktifkan AWS IoT SiteWise untuk menyimpan data dalam tingkat dingin untuk pertama kalinya, metadata hierarki aset diekspor ke tingkat dingin. Setelah konfigurasi awal, AWS IoT SiteWise mengeksport metadata hierarki aset ke tingkat dingin hanya jika Anda membuat perubahan pada model aset atau definisi aset. Metadata hierarki aset disimpan dalam tingkat dingin dalam format JSON () yang dibatasi baris baru. .ndjson

Pengidentifikasi eksternal untuk hierarki, aset target, atau aset sumber diambil dengan memanggil API. [DescribeAsset](#)

	Amazon S3 menggunakan awalan sebagai nama folder di bucket.
<code>asset_hierarchy_metadata</code>	Folder yang menyimpan metadata hierarki aset. <code>asset_hierarchy_metadata</code> Folder disimpan di folder awalan.
<code>fileName</code>	Nama file menggunakan karakter garis bawah (<code>_</code>) sebagai pembatas untuk memisahkan yang berikut: <ul style="list-style-type: none"> <code>parentAssetId</code> Nilainya. <code>hierarchyId</code> Nilainya. File disimpan dalam <code>.ndjson</code> format.
Komponen jalur	Deskripsi

Example jalur file ke metadata hierarki aset di tingkat dingin

```
keyPrefix/asset_hierarchy_metadata/35901915-d476-4dca-8637-
d9ed4df939ed_c5b3ced8-589a-48c7-9998-cdcccfc9747a0.ndjson
```

Bidang

Skema metadata hierarki aset yang diekspor ke tingkat dingin berisi bidang berikut.

<code>sourceAssetId</code>	ID aset sumber dalam hubungan aset ini.
<code>targetAssetId</code>	ID aset target dalam hubungan aset ini.
<code>hierarchyId</code>	ID hierarki.
<code>associationType</code>	Jenis asosiasi dari hubungan aset ini. Nilainya harus <code>CHILD</code> . Aset target adalah aset anak dari aset sumber.
Nama bidang	Deskripsi

Example metadata hierarki aset di tingkat dingin

```

{"sourceAssetId":"80388e72-2284-44fb-9c89-
bfba0dfedd2","targetAssetId":"2b866c25-0c74-4750-bdf5-
b73683c8a2a2","hierarchyId":"bbed9f59-0412-4585-
a61d-6044db526aee","associationType":"CHILD"}

{"sourceAssetId":"80388e72-2284-44fb-9c89-
bfba0dfedd2","targetAssetId":"6b51246e-984d-460d-
bc0b-470ea47d1e31","hierarchyId":"bbed9f59-0412-4585-
a61d-6044db526aee","associationType":"CHILD"}

```

Untuk melihat data Anda di tingkat dingin

1. Arahkan ke [konsol Amazon S3](#).
2. Di panel navigasi, pilih Bucket, lalu pilih bucket Amazon S3 Anda.
3. Arahkan ke folder yang berisi data mentah, metadata aset, atau metadata hierarki aset.
4. Pilih file, dan kemudian dari Tindakan, pilih Unduh.

File indeks data penyimpanan

AWS IoT SiteWise menggunakan file-file ini untuk mengoptimalkan kinerja kueri data. Mereka muncul di ember Amazon S3 Anda, tetapi Anda tidak perlu menggunakannya.

Jalur berkas

AWS IoT SiteWise menyimpan file indeks data di tingkat dingin menggunakan template berikut.

```

keyPrefix/index/series=timeseriesId/startYear=startYear/startMonth=startMonth/
startDay=startDay/index_timeseriesId_startTimestamp_quality

```

Example jalur file ke file indeks penyimpanan data

File indeks data penyimpanan

```

keyPrefix/index/series=7020c8e2-e6db-40fa-9845-ed0ddd4c77d_95e63da7-
d34e-43e1-bc6f-1b490154b07a/startYear=2022/startMonth=02/startDay=03/
index_7020c8e2-e6db-40fa-9845-ed0ddd4c77d_95e63da7-d34e-43e1-

```


Anda juga dapat mengimpor dan mengekspor metadata aset. Untuk mengetahui informasi selengkapnya, lihat

Saat Anda mengaktifkan AWS IoT SiteWise untuk mengekspor data ke tingkat dingin untuk pertama kalinya, metadata aset diekspor ke tingkat dingin. Setelah konfigurasi awal, AWS IoT SiteWise ekspor metadata aset ke tingkat hanya jika Anda mengubah definisi model aset atau definisi aset. Metadata aset disimpan dalam tingkat dingin dalam format JSON () yang dibatasi baris baru. .ndjson

Jalur berkas

AWS IoT SiteWise menyimpan metadata aset di tingkat dingin menggunakan templat berikut.

```
{keyPrefix}/asset_metadata/asset_{assetId}.ndjson
```

Setiap jalur file ke metadata aset di tingkat dingin berisi komponen berikut.

keyPrefix	Awalan Amazon S3 yang Anda tentukan dalam konfigurasi penyimpanan AWS IoT SiteWise s. Amazon S3 menggunakan awalan sebagai nama folder di bucket.
asset_metadata	Folder yang menyimpan metadata aset. asset_metadata Folder disimpan di folder awalan.
fileName	Nama file menggunakan karakter garis bawah (_) sebagai pembatas untuk memisahkan yang berikut: <ul style="list-style-type: none"> assetAwalan. assetIdNilainya. File disimpan dalam .ndjson format.
Komponen jalur	Deskripsi

Example jalur file ke metadata aset di tingkat yang lebih dingin

`keyPrefix/asset_metadata/asset_35901915-d476-4dca-8637-d9ed4df939ed.ndjson`

Bidang

Skema metadata aset yang diekspor ke tingkat dingin berisi bidang-bidang berikut.

<code>assetId</code>	ID aset.
<code>assetName</code>	Nama aset.
<code>assetExternalId</code>	ID eksternal aset.
<code>assetModelId</code>	ID model aset yang digunakan untuk membuat aset ini.
<code>assetModelName</code>	Nama model aset.
<code>assetModelExternalId</code>	ID eksternal dari model aset.
<code>assetPropertyId</code>	ID dari properti aset.
<code>assetPropertyName</code>	Nama properti aset.
<code>assetPropertyExternalId</code>	ID eksternal dari properti aset.
<code>assetPropertyDataType</code>	Tipe data properti aset.
<code>assetPropertyUnit</code>	Unit properti aset (misalnya, Newtons danRPM).
<code>assetPropertyAlias</code>	Alias yang mengidentifikasi properti aset, seperti jalur aliran data server OPC-UA (misalnya,). /company/windfarm/3/turbine/7/temperature
<code>timeSeriesId</code>	ID yang mengidentifikasi data deret waktu dari peralatan, metrik, atau transformasi.

	Anda dapat menggunakan bidang ini untuk menggabungkan data mentah dan metadata aset dalam kueri.
<code>timeSeriesBucket</code>	Angka heksadesimal antara 00 dan ff. Angka ini berasal dari <code>timeSeriesId</code> . Partisi ini digunakan untuk meningkatkan throughput saat AWS IoT SiteWise menulis ke tingkat dingin. Saat Anda menggunakan Amazon Athena untuk menjalankan kueri, Anda dapat menggunakan partisi untuk partisi butir halus guna meningkatkan kinerja kueri. <code>timeSeriesBucket</code> dan <code>seriesBucket</code> di jalur file ke data mentah adalah nomor yang sama.
<code>assetCompositeModelId</code>	ID dari model komposit.
<code>assetCompositeModelExternalId</code>	ID eksternal dari model komposit.
<code>assetCompositeModelDescription</code>	Deskripsi dari model komposit.
<code>assetCompositeModelName</code>	Nama dari model komposit.
<code>assetCompositeModelType</code>	Jenis dari model komposit. Untuk model komposit alarm, jenis ini berupa <code>AWS/ALARM</code> .
<code>assetCreationDate</code>	Tanggal aset dibuat, dalam waktu zaman Unix.
<code>assetLastUpdateDate</code>	Tanggal aset terakhir diperbarui, dalam waktu zaman Unix.
<code>assetStatusErrorCode</code>	Kode kesalahan.
<code>assetStatusErrorMessage</code>	Pesan kesalahan.
<code>assetStatusState</code>	Status aset saat ini.
Nama bidang	Deskripsi

Example metadata aset di tingkat dingin

<code>{"assetId":"7020c8e2-e6db-40fa-9845-</code>
<code>ed0dddd4c77d","assetExternalId":null,"assetName":"Wind Turbine Asset</code>
<code>2","assetModelId":"ec1d924f-f07d-444f-b072-</code>
<code>e2994c165d35","assetModelExternalId":null,"assetModelName":"Wind</code>
<code>Turbine Asset Model","assetPropertyId":"95e63da7-d34e-43e1-</code>
<code>bc6f-1b490154b07a","assetPropertyExternalId":null,"assetPropertyName":"Temperature","assetPrope</code>
<code>Washington/Seattle/WT2/temp","timeSeriesId":"7020c8e2-e6db-40fa-9845-</code>
<code>ed0dddd4c77d_95e63da7-d34e-43e1-</code>
<code>bc6f-1b490154b07a","timeSeriesBucket":"f6","assetArn":null,"assetCompositeModelDescription":nul</code>
<code>{"assetId":"7020c8e2-e6db-40fa-9845-</code>
<code>ed0dddd4c77d","assetExternalId":null,"assetName":"Wind Turbine Asset</code>
<code>2","assetModelId":"ec1d924f-f07d-444f-b072-</code>
<code>e2994c165d35","assetModelExternalId":null,"assetModelName":"Wind Turbine Asset</code>
<code>Model","assetPropertyId":"c706d54d-4c11-42dc-9a01-63662fc697b4","assetPropertyExternalId":null</code>
<code>Washington/Seattle/WT2/pressure","timeSeriesId":"7020c8e2-e6db-40fa-9845-</code>
<code>ed0dddd4c77d_c706d54d-4c11-42dc-9a01-63662fc697b4","timeSeriesBucket":"1e","assetArn":null,"ass</code>
<code>{"assetId":"7020c8e2-e6db-40fa-9845-</code>
<code>ed0dddd4c77d","assetExternalId":null,"assetName":"Wind Turbine Asset</code>
<code>2","assetModelId":"ec1d924f-f07d-444f-b072-</code>
<code>e2994c165d35","assetModelExternalId":null,"assetModelName":"Wind</code>
<code>Turbine Asset Model","assetPropertyId":"8cf1162f-dead-4fbe-b468-</code>

Mengintegrasikan dengan layanan lainnya

AWS IoT SiteWise terintegrasi dengan beberapa AWS layanan untuk mengembangkan AWS IoT solusi lengkap di AWS Cloud. Untuk informasi selengkapnya, lihat [Berinteraksi dengan layanan lain AWS](#)

AWS IoT SiteWise konsep

Berikut ini adalah konsep inti dari AWS IoT SiteWise:

Agregat

Agregat adalah metrik fundamental, atau pengukuran, yang AWS IoT SiteWise secara otomatis menghitung untuk semua data deret waktu. Untuk informasi selengkapnya, lihat [Kueri agregat properti aset](#).

Aset

Saat Anda memasukkan, atau menelan, data AWS IoT SiteWise dari peralatan industri Anda, perangkat, peralatan, dan proses Anda masing-masing ditampilkan sebagai aset. Setiap aset memiliki data terkait. Misalnya, peralatan mungkin memiliki nomor seri, lokasi, merek dan model, dan tanggal pemasangan. Mungkin juga memiliki nilai deret waktu untuk ketersediaan, kinerja, kualitas, suhu, tekanan, dan banyak lagi. Kelompokkan aset ke dalam hierarki, memungkinkan aset untuk mengakses data yang disimpan dalam aset anak mereka. Untuk informasi selengkapnya, lihat [Pemodelan aset industri](#).

Hirarki aset

Siapkan hierarki aset untuk membuat representasi logis dari operasi industri Anda. Untuk melakukan ini, tentukan hierarki dalam model aset dan kaitkan aset yang dibuat dari model itu dengan hierarki yang ditentukan. Metrik dalam aset induk dapat menggabungkan data dari properti aset anak, memungkinkan Anda menghitung metrik yang menawarkan wawasan tentang keseluruhan operasi Anda atau bagian tertentu darinya. Untuk informasi selengkapnya, lihat [Mendefinisikan hierarki model aset](#).

Model aset

Setiap aset dibuat menggunakan model aset. Model aset adalah struktur yang menentukan dan menstandarisasi format aset Anda. Mereka memastikan informasi yang konsisten di beberapa aset dengan jenis yang sama, memungkinkan Anda menangani data dalam aset yang mewakili grup perangkat. [Di setiap model aset, Anda dapat menentukan atribut, input deret waktu](#)

[\(pengukuran\), transformasi deret waktu \(transformasi\), agregasi deret waktu \(metrik\), dan hierarki aset](#). Untuk informasi selengkapnya, lihat [Pemodelan aset industri](#).

Tentukan di mana properti model aset Anda diproses dengan mengonfigurasi model aset Anda untuk edge. Manfaatkan fitur ini untuk menangani dan memantau data aset di perangkat lokal Anda.

Properti aset

Properti aset adalah struktur dalam setiap aset yang menyimpan data industri. Setiap properti memiliki tipe data dan juga dapat memiliki unit. Properti dapat berupa [atribut](#), [pengukuran](#), [transformasi](#), atau [metrik](#). Untuk informasi selengkapnya, lihat [Mendefinisikan properti data](#).

Konfigurasi properti aset untuk menghitung di tepi. Untuk informasi selengkapnya tentang pemrosesan data di edge, lihat [the section called “Mengaktifkan pemrosesan data tepi”](#).

Atribut

Atribut adalah properti aset yang biasanya tetap konstan, seperti produsen perangkat atau lokasi perangkat. Atribut dapat memiliki nilai preset. Setiap aset yang dibuat dari model aset mencakup nilai default dari atribut yang ditentukan dalam model tersebut. Untuk informasi selengkapnya, lihat [Mendefinisikan data statis \(atribut\)](#).

Dasbor

Setiap proyek berisi satu set dasbor. Dasbor menyediakan satu set visualisasi untuk nilai-nilai dari satu set aset. Pemilik proyek membuat dasbor dan visualisasi yang dikandungnya. Ketika pemilik proyek siap untuk membagikan kumpulan dasbor, pemilik dapat mengundang pemirsa ke proyek, yang memberi mereka akses ke semua dasbor dalam proyek. Jika Anda menginginkan kumpulan pemirsa yang berbeda untuk dasbor yang berbeda, Anda harus membagi dasbor di antara proyek. Saat pemirsa melihat dasbor, mereka dapat menyesuaikan rentang waktu untuk melihat data tertentu.

Aliran data

Masukan, atau konsumsi, data industri AWS IoT SiteWise bahkan sebelum membuat model aset dan aset. AWS IoT SiteWise secara otomatis menghasilkan aliran data untuk mengumpulkan aliran data mentah dari peralatan Anda.

Alias aliran data

Alias aliran data membantu Anda mengidentifikasi aliran data dengan mudah. Misalnya, alias `server1-windfarm/3/turbine/7/temperature` menunjukkan nilai suhu yang berasal dari turbin #7 di ladang angin #3. Istilah `server1` ini adalah nama sumber data yang membantu

mengidentifikasi server OPC-UA, dan `server1`- merupakan awalan yang dilampirkan ke semua aliran data yang dilaporkan dari server OPC-UA ini.

Asosiasi aliran data

Setelah Anda membuat model aset dan aset, kaitkan aliran data dengan properti aset yang ditentukan dalam aset Anda untuk menyusun data Anda. AWS IoT SiteWise kemudian dapat menggunakan model aset dan aset untuk menangani data yang masuk dari aliran data Anda. Anda juga dapat memisahkan aliran data dari properti aset. Untuk informasi selengkapnya, lihat [Mengelola aliran data](#).

Formula

Setiap properti [transformasi](#) dan [metrik](#) dilengkapi dengan rumus yang menguraikan bagaimana properti mengubah atau mengumpulkan data. Rumus ini mencakup input properti, operator, dan fungsi yang ditawarkan oleh. AWS IoT SiteWise Untuk informasi selengkapnya, lihat [Menggunakan ekspresi rumus](#).

Pengukuran

Pengukuran adalah properti aset yang menggambarkan aliran data deret waktu sensor mentah dari perangkat atau peralatan. Untuk informasi selengkapnya, lihat [Mendefinisikan aliran data dari peralatan \(pengukuran\)](#).

Metrik

Metrik adalah properti aset yang mewakili data deret waktu agregat. Setiap metrik disertai dengan ekspresi matematika ([rumus](#)) yang menguraikan cara mengumpulkan titik data dan interval waktu untuk menghitung agregasi tersebut. Metrik menghasilkan satu titik data untuk setiap interval waktu yang ditentukan. Untuk informasi selengkapnya, lihat [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#).

Paket

SiteWise Edge gateway menggunakan paket untuk menentukan cara mengumpulkan, memproses, dan merutekan data. Saat ini, AWS IoT SiteWise mendukung paket pengumpulan data dan paket pemrosesan data. Untuk informasi selengkapnya tentang paket yang tersedia untuk gateway SiteWise Edge Anda, lihat [the section called "Menggunakan paket"](#).

Paket pengumpulan data

Gunakan paket pengumpulan data sehingga gateway SiteWise Edge Anda dapat mengumpulkan data industri Anda dan mengarahkannya ke AWS tujuan pilihan Anda. Paket ini secara otomatis ditambahkan ke gateway SiteWise Edge Anda dan tidak dapat dihapus.

Paket pengolahan data

Gunakan paket pemrosesan data untuk memproses data Anda di tepi dan menyimpannya selama 30 hari untuk digunakan dalam aplikasi lokal.

Portal

AWS IoT SiteWise Monitor Portal adalah aplikasi web yang dapat Anda gunakan untuk memvisualisasikan dan berbagi AWS IoT SiteWise data Anda. Portal memiliki satu atau lebih administrator dan berisi nol atau lebih proyek.

Administrator portal

Setiap portal SiteWise Monitor memiliki satu atau lebih administrator portal. Administrator portal menggunakan portal untuk membuat proyek yang berisi koleksi aset dan dasbor. Administrator portal kemudian menetapkan aset dan pemilik untuk setiap proyek. Dengan mengontrol akses ke proyek, administrator portal menentukan aset mana yang dapat dilihat oleh pemilik proyek dan pemirsa.

Proyek

Setiap portal SiteWise Monitor berisi serangkaian proyek. Setiap proyek memiliki subset AWS IoT SiteWise aset Anda yang terkait dengannya. Pemilik proyek membuat satu atau beberapa dasbor untuk menyediakan cara yang konsisten untuk melihat data yang terkait dengan aset tersebut. Pemilik proyek dapat mengundang pemirsa ke proyek untuk memungkinkan mereka melihat aset dan dasbor dalam proyek. Proyek ini adalah unit dasar berbagi dalam SiteWise Monitor. Pemilik proyek dapat mengundang pengguna yang diberi akses ke portal oleh AWS administrator. Pengguna harus memiliki akses ke portal sebelum proyek di portal itu dapat dibagikan dengan pengguna tersebut.

Pemilik proyek

Setiap proyek SiteWise Monitor memiliki pemilik. Pemilik proyek membuat visualisasi dalam bentuk dasbor untuk mewakili data operasional secara konsisten. Ketika dasbor siap untuk dibagikan, pemilik proyek dapat mengundang pemirsa ke proyek. Pemilik proyek juga dapat menetapkan pemilik lain untuk proyek. Pemilik proyek dapat mengonfigurasi ambang batas dan pengaturan notifikasi untuk alarm.

Penampil proyek

Setiap proyek SiteWise Monitor memiliki pemirsa. Pemirsa proyek dapat terhubung ke portal untuk melihat dasbor yang dibuat oleh pemilik proyek. Di setiap dasbor, pemirsa proyek dapat

menyesuaikan rentang waktu untuk lebih memahami data operasional. Pemirsa proyek hanya dapat melihat dasbor di proyek yang dapat mereka akses. Pemirsa proyek dapat mengenali dan menunda alarm.

Alias properti

Anda memiliki opsi untuk membuat alias pada properti aset, seperti jalur aliran data server OPC-UA (misalnya, /company/windfarm/3/turbine/7/temperature), menyederhanakan identifikasi properti aset selama konsumsi atau pengambilan data aset. Saat Anda menggunakan [gateway SiteWise Edge](#) untuk menyerap data dari server, alias properti Anda harus cocok dengan jalur aliran data mentah Anda. Untuk informasi selengkapnya, lihat [Memetakan aliran data industri ke properti aset](#).

Pemberitahuan properti

Saat Anda mengaktifkan pemberitahuan properti untuk properti aset, AWS IoT SiteWise menerbitkan pesan MQTT ke AWS IoT Core setiap kali properti tersebut menerima nilai baru. Payload pesan mencakup detail tentang pembaruan ke nilai properti tersebut. Gunakan notifikasi nilai properti untuk membuat solusi yang menghubungkan data industri Anda AWS IoT SiteWise dengan AWS layanan lain. Untuk informasi selengkapnya, lihat [Berinteraksi dengan layanan lain AWS](#).

SiteWise Gerbang tepi

Gateway SiteWise Edge terletak di tempat pelanggan untuk mengumpulkan, menangani, dan mengarahkan data. Gateway SiteWise Edge terhubung ke sumber data industri Anda melalui protokol [OPC-UA](#) untuk mengumpulkan dan memproses data, mengirimkannya ke cloud. AWS SiteWise Edge gateway juga dapat terhubung ke sumber [data mitra](#). SiteWise Edge gateway menggunakan paket untuk pengumpulan data, pemrosesan tepi, dan banyak lagi. Untuk informasi selengkapnya tentang paket yang tersedia, lihat [the section called “Menggunakan paket”](#).

Anda memiliki fleksibilitas untuk membuat gateway SiteWise Edge pada perangkat atau platform apa pun yang mampu berjalan AWS IoT Greengrass. Untuk informasi selengkapnya, lihat [Menggunakan SiteWise gateway Edge](#).

Transformasi

Transformasi adalah properti aset yang mewakili data deret waktu yang ditransformasikan. Setiap transformasi disertai dengan ekspresi matematika ([rumus](#)) yang menentukan bagaimana mengkonversi titik data dari satu bentuk ke bentuk lainnya. Titik data yang ditransformasikan memiliki one-to-one hubungan dengan titik data input. Untuk informasi selengkapnya, lihat [Mengubah data \(transformasi\)](#).

Visualisasi

Di setiap dasbor, pemilik proyek memutuskan bagaimana menampilkan properti dan alarm aset yang terkait dengan proyek. Ketersediaan dapat direpresentasikan sebagai grafik garis, sementara nilai lainnya mungkin ditampilkan sebagai grafik batang atau indikator kinerja utama (KPI). Alarm paling baik ditampilkan sebagai kisi-kisi status dan jadwal status. Pemilik proyek menyesuaikan setiap visualisasi untuk memberikan pemahaman terbaik tentang data untuk aset tersebut.

Gunakan kasus untuk AWS IoT SiteWise

AWS IoT SiteWise digunakan di berbagai industri untuk banyak pengumpulan data industri dan aplikasi analisis.

Kumpulkan data secara konsisten dari semua sumber Anda untuk membantu menyelesaikan masalah dengan cepat. AWS IoT SiteWise menawarkan pemantauan jarak jauh untuk mengumpulkan data secara langsung di tempat atau mengumpulkannya dari berbagai sumber di banyak fasilitas. AWS IoT SiteWise memberikan fleksibilitas yang diperlukan untuk solusi data IoT industri.

Manufaktur

AWS IoT SiteWise dapat menyederhanakan proses pengumpulan dan pemanfaatan data dari peralatan Anda untuk menentukan dan meminimalkan inefisiensi, meningkatkan operasi industri. AWS IoT SiteWise membantu Anda mengumpulkan data dari jalur manufaktur dan peralatan. Dengan AWS IoT SiteWise, Anda dapat mentransfer data ke AWS Cloud dan membuat metrik kinerja untuk peralatan dan proses spesifik Anda. Anda dapat menggunakan metrik yang dihasilkan untuk memahami efektivitas keseluruhan operasi Anda dan mengidentifikasi peluang untuk inovasi dan peningkatan. Anda juga dapat melihat proses manufaktur Anda dan mengidentifikasi kekurangan peralatan dan proses, kesenjangan produksi, atau cacat produk.

Makanan dan minuman

Fasilitas industri makanan dan minuman menangani berbagai macam pengolahan makanan, termasuk menggiling biji-bijian menjadi tepung, menyembelih dan mengemas daging, dan merakit, memasak, dan membekukan makanan microwave. Pabrik pengolahan makanan sering menjangkau beberapa lokasi dengan operator pabrik dan peralatan di lokasi terpusat untuk memantau proses dan peralatan. Misalnya, unit pendingin menilai penanganan dan kedaluwarsa bahan. Mereka memantau

pembuatan limbah di seluruh fasilitas untuk memastikan efisiensi operasional. Dengan AWS IoT SiteWise, Anda dapat mengelompokkan aliran data sensor dari beberapa lokasi berdasarkan jalur produksi, dan fasilitas sehingga teknisi proses Anda dapat lebih memahami dan melakukan perbaikan di seluruh fasilitas.

Energi dan utilitas

Dengan AWS IoT SiteWise, Anda dapat menyelesaikan masalah peralatan dengan lebih mudah dan lebih efisien. Anda dapat memantau kinerja aset dari jarak jauh dan real time. Akses data peralatan historis dari mana saja untuk menentukan potensi masalah, mengirimkan sumber daya yang akurat, dan mencegah dan memperbaiki masalah dengan lebih cepat.

Memulai dengan AWS IoT SiteWise

Dengan AWS IoT SiteWise, Anda dapat mengumpulkan, mengatur, menganalisis, dan memvisualisasikan data Anda.

AWS IoT SiteWise menyediakan demo yang dapat Anda gunakan untuk menjelajahi layanan tanpa mengonfigurasi sumber data nyata. Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT SiteWise demo](#).

Anda dapat menyelesaikan tutorial berikut untuk menjelajahi fitur-fitur tertentu dari AWS IoT SiteWise:

- [Menelan data dari AWS IoT berbagai hal](#)
- [Memvisualisasikan dan berbagi data ladang angin di Monitor SiteWise](#)
- [Menerbitkan pembaruan nilai properti ke Amazon DynamoDB](#)

Lihat topik berikut untuk mempelajari lebih lanjut tentang AWS IoT SiteWise:

- [Menelan data ke AWS IoT SiteWise](#)
- [Pemodelan aset industri](#)
- [Mengaktifkan pemrosesan data tepi](#)
- [Memantau data dengan AWS IoT SiteWise Monitor](#)
- [Data kueri dari AWS IoT SiteWise](#)
- [Berinteraksi dengan layanan lain AWS](#)

Topik

- [Persyaratan](#)
- [Menyiapkan sebuah Akun AWS](#)
- [Menggunakan AWS IoT SiteWise demo](#)

Persyaratan

Anda harus memiliki Akun AWS untuk memulai AWS IoT SiteWise. Jika Anda tidak memilikinya, lihat [Menyiapkan sebuah Akun AWS](#).

Gunakan Wilayah di mana AWS IoT SiteWise tersedia. Untuk informasi lebih lanjut, lihat [AWS IoT SiteWise kuota dan titik akhir](#). Anda dapat menggunakan pemilih Wilayah di AWS Management Console untuk beralih ke salah satu Wilayah ini.

Menyiapkan sebuah Akun AWS

Topik

- [Mendaftar untuk Akun AWS](#)
- [Membuat pengguna administratif](#)

Mendaftar untuk Akun AWS

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar untuk Akun AWS

1. Buka <https://portal.aws.amazon.com/billing/signup>.
2. Ikuti petunjuk secara online.

Anda akan diminta untuk menerima panggilan telepon dan memasukkan kode verifikasi pada keypad telepon sebagai bagian dari prosedur pendaftaran.

Saat Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWS dibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya dalam akun. Sebagai praktik terbaik keamanan, [tetapkan akses administratif ke pengguna administratif](#), dan hanya gunakan pengguna root untuk melakukan [tugas yang memerlukan akses pengguna root](#).

AWS mengirimkan Anda email konfirmasi setelah proses pendaftaran selesai. Anda dapat melihat aktivitas akun saat ini dan mengelola akun dengan mengunjungi <https://aws.amazon.com/> dan memilih Akun Saya.

Membuat pengguna administratif

Setelah Anda mendaftar Akun AWS, amankan Pengguna root akun AWS, aktifkan AWS IAM Identity Center, dan buat pengguna administratif sehingga Anda tidak menggunakan pengguna root untuk tugas sehari-hari.

Amankan Anda Pengguna root akun AWS

1. Masuk ke [AWS Management Console](#) sebagai pemilik akun dengan memilih pengguna Root dan memasukkan alamat Akun AWS email Anda. Di halaman berikutnya, masukkan kata sandi Anda.

Untuk bantuan masuk menggunakan pengguna root, lihat [Masuk sebagai pengguna root](#) dalam Panduan Pengguna AWS Sign-In .

2. Aktifkan autentikasi multi-faktor (MFA) untuk pengguna root Anda.

Untuk petunjuk, lihat [Mengaktifkan perangkat MFA virtual untuk pengguna Akun AWS root \(konsol\) Anda](#) di Panduan Pengguna IAM.

Membuat pengguna administratif

1. Aktifkan Pusat Identitas IAM.

Untuk mendapatkan petunjuk, silakan lihat [Mengaktifkan AWS IAM Identity Center](#) di Panduan Pengguna AWS IAM Identity Center .

2. Di Pusat Identitas IAM, berikan akses administratif ke sebuah pengguna administratif.

Untuk tutorial tentang menggunakan Direktori Pusat Identitas IAM sebagai sumber identitas Anda, lihat [Mengkonfigurasi akses pengguna dengan default Direktori Pusat Identitas IAM](#) di Panduan AWS IAM Identity Center Pengguna.

Masuk sebagai pengguna administratif

- Untuk masuk dengan pengguna Pusat Identitas IAM, gunakan URL masuk yang dikirim ke alamat email Anda saat Anda membuat pengguna Pusat Identitas IAM.

Untuk bantuan masuk menggunakan pengguna Pusat Identitas IAM, lihat [Masuk ke portal AWS akses](#) di Panduan AWS Sign-In Pengguna.

Menggunakan AWS IoT SiteWise demo

Anda dapat dengan mudah menjelajahi AWS IoT SiteWise dengan menggunakan AWS IoT SiteWise demo. AWS IoT SiteWise menyediakan demo sebagai AWS CloudFormation template yang dapat Anda terapkan untuk membuat model aset, aset, dan portal SiteWise Monitor, dan menghasilkan data sampel hingga satu minggu.

⚠ Important

Setelah Anda membuat demo, Anda akan mulai dikenakan biaya untuk sumber daya yang dibuat dan dikonsumsi demo ini.

Topik

- [Membuat AWS IoT SiteWise demo](#)
- [Menghapus demo AWS IoT SiteWise](#)

Membuat AWS IoT SiteWise demo

Anda dapat membuat AWS IoT SiteWise demo dari AWS IoT SiteWise konsol.

ℹ Note

Demo membuat fungsi Lambda, satu aturan CloudWatch Acara, dan peran AWS Identity and Access Management (IAM) yang diperlukan untuk demo. Anda mungkin melihat sumber daya ini di Anda Akun AWS. Kami menyarankan Anda menyimpan sumber daya ini sampai Anda selesai dengan demo. Jika Anda menghapus sumber daya, demo mungkin berhenti bekerja dengan benar.

Untuk membuat demo di AWS IoT SiteWise konsol

1. Arahkan ke [AWS IoT SiteWise konsol](#) dan temukan SiteWise demo di sudut kanan atas halaman.
2. (Opsional) Di bawah SiteWise demo, ubah bidang Days to keep demo assets untuk menentukan berapa hari untuk menyimpan demo sebelum menghapusnya.
3. (Opsional) Untuk membuat portal SiteWise Monitor untuk memantau data sampel, lakukan hal berikut.

Note

Anda akan dikenakan biaya untuk sumber daya SiteWise Monitor yang dibuat dan dikonsumsi oleh demo ini. Untuk informasi selengkapnya, lihat [SiteWise Monitor](#) di AWS IoT SiteWise Harga.

- a. Pilih Monitor Sumber Daya.
- b. Pilih Izin.
- c. Pilih peran IAM yang ada yang memberikan akses kepada pengguna IAM federasi Anda ke portal.

⚠ Important

Peran IAM Anda harus memiliki izin berikut.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:Describe*",
        "iotsitewise:List*",
        "iotsitewise:Get*",
        "cloudformation:DescribeStacks",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedRolePolicies",
        "sso:DescribeRegisteredRegions",
        "organizations:DescribeOrganization"
      ],
      "Resource": "*"
    }
  ]
}
```


Untuk informasi selengkapnya tentang cara bekerja dengan SiteWise Monitor, lihat [Apa itu AWS IoT SiteWise Monitor?](#) dalam Panduan AWS IoT SiteWise Monitor Aplikasi.

4. Pilih Buat demo.

Demo membutuhkan waktu sekitar 3 menit untuk dibuat. Jika demo gagal dibuat, akun Anda mungkin memiliki izin yang tidak mencukupi. Beralih ke akun yang memiliki izin administratif, atau gunakan langkah-langkah berikut untuk menghapus demo dan coba lagi:

a. Pilih Hapus demo.

Demo membutuhkan waktu sekitar 15 menit untuk dihapus.

b. Jika demo tidak dihapus, buka [AWS CloudFormation konsol](#), pilih tumpukan bernama IoT SiteWiseDemoAssets, dan pilih Hapus di sudut kanan atas.

c. Jika demo gagal dihapus lagi, ikuti langkah-langkah di AWS CloudFormation konsol untuk melewati sumber daya yang gagal dihapus, dan coba lagi.

5. Setelah demo berhasil dibuat, Anda dapat menjelajahi aset dan data demo di [AWS IoT SiteWise konsol](#).

Menghapus demo AWS IoT SiteWise

AWS IoT SiteWise Demo menghapus dirinya sendiri setelah seminggu, atau jumlah hari yang Anda pilih jika Anda membuat tumpukan demo dari AWS CloudFormation konsol. Anda dapat menghapus demo sebelumnya jika Anda selesai menggunakan sumber daya demo. Anda juga dapat menghapus demo jika demo gagal dibuat. Gunakan langkah-langkah berikut untuk menghapus demo secara manual.

Untuk menghapus AWS IoT SiteWise demo

1. Navigasikan ke [konsol AWS CloudFormation](#) tersebut.
2. Pilih IoTSiteWiseDemoAssets dari daftar Stacks.
3. Pilih Hapus.

Saat Anda menghapus tumpukan, semua sumber daya yang dibuat untuk demo akan dihapus.

4. Dalam dialog konfirmasi, pilih Hapus tumpukan.

Tumpukan membutuhkan waktu sekitar 15 menit untuk dihapus. Jika demo gagal dihapus, pilih Hapus di sudut kanan atas lagi. Jika demo gagal dihapus lagi, ikuti langkah-langkah di AWS CloudFormation konsol untuk melewati sumber daya yang gagal dihapus, dan coba lagi.

AWS IoT SiteWise tutorial

Selamat datang di halaman AWS IoT SiteWise tutorial. Kumpulan tutorial yang berkembang ini memberdayakan Anda dengan pengetahuan dan keterampilan yang dibutuhkan untuk menavigasi seluk-beluk. AWS IoT SiteWise Tutorial ini menawarkan beragam topik dasar untuk memenuhi kebutuhan Anda. Saat Anda mempelajari tutorial, temukan wawasan yang tak ternilai ke dalam berbagai aspek. AWS IoT SiteWise

Setiap tutorial menggunakan contoh peralatan tertentu. Tutorial ini ditujukan untuk lingkungan pengujian, dan mereka menggunakan nama perusahaan fiktif, model, aset, properti, dan sebagainya. Tujuannya adalah untuk memberikan pedoman umum. Tutorial tidak dimaksudkan untuk penggunaan langsung dalam lingkungan produksi tanpa tinjauan dan adaptasi yang cermat untuk memenuhi kebutuhan unik organisasi Anda.

Topik

- [Menghitung OEE di AWS IoT SiteWise](#)
- [Menelan data dari AWS IoT berbagai hal](#)
- [Memvisualisasikan dan berbagi data ladang angin di Monitor SiteWise](#)
- [Menerbitkan pembaruan nilai properti ke Amazon DynamoDB](#)

Menghitung OEE di AWS IoT SiteWise

Tutorial ini memberikan contoh bagaimana menghitung efektivitas peralatan secara keseluruhan (OEE) untuk proses manufaktur. Akibatnya, perhitungan atau rumus OEE Anda mungkin berbeda dari yang ditunjukkan di sini. Secara umum, OEE didefinisikan sebagai $Availability * Quality * Performance$. Untuk mempelajari lebih lanjut tentang menghitung OEE, lihat [Efektivitas peralatan secara keseluruhan](#) di Wikipedia.

Prasyarat

Untuk menyelesaikan tutorial ini, Anda harus mengonfigurasi konsumsi data untuk perangkat yang memiliki tiga aliran data berikut:

- `Equipment_State`— Kode numerik yang mewakili keadaan mesin, seperti idle, kesalahan, penghentian yang direncanakan, atau operasi normal.

- **Good_Count**— Aliran data di mana setiap titik data berisi jumlah operasi yang berhasil sejak titik data terakhir.
- **Bad_Count**— Aliran data di mana setiap titik data berisi jumlah operasi yang gagal sejak titik data terakhir.

Untuk mengonfigurasi konsumsi data, lihat [Menelan data ke AWS IoT SiteWise](#). Jika Anda tidak memiliki operasi industri yang tersedia, Anda dapat menulis skrip yang menghasilkan dan mengunggah data sampel melalui AWS IoT SiteWise API.

Cara menghitung OEE

Dalam tutorial ini, Anda membuat model aset yang menghitung OEE dari tiga aliran input data: `Equipment_State`, `Good_Count` dan `Bad_Count`. Dalam contoh ini, pertimbangkan mesin pengemas generik, seperti yang digunakan untuk mengemas gula, keripik kentang, atau cat. Di [AWS IoT SiteWise konsol](#), buat model AWS IoT SiteWise aset dengan pengukuran, transformasi, dan metrik berikut. Kemudian, Anda dapat membuat aset untuk mewakili mesin pengemas dan mengamati bagaimana AWS IoT SiteWise menghitung OEE.

Tentukan [pengukuran](#) berikut untuk mewakili aliran data mentah dari mesin pengemas.

Pengukuran

- **Equipment_State**— Aliran data (atau pengukuran) yang menyediakan keadaan mesin pengemas saat ini dalam kode numerik:
 - `1024`— Mesin dalam keadaan idle.
 - `1020`— Kesalahan, seperti kesalahan atau penundaan.
 - `1000`— Sebuah pemberhentian yang direncanakan.
 - `1111`— Sebuah operasi yang normal.
- **Good_Count**— Aliran data di mana setiap titik data berisi jumlah operasi yang berhasil sejak titik data terakhir.
- **Bad_Count**— Aliran data di mana setiap titik data berisi jumlah operasi yang gagal sejak titik data terakhir.

Dengan menggunakan aliran data `Equipment_State` pengukuran dan kode yang dikandungnya, tentukan [transformasi](#) berikut (atau pengukuran turunan). Transformasi memiliki one-to-one hubungan dengan pengukuran mentah.

Transformasi

- `Idle = eq(Equipment_State, 1024)`— Aliran data yang diubah yang berisi keadaan idle mesin.
- `Fault = eq(Equipment_State, 1020)`— Aliran data yang diubah yang berisi status kesalahan mesin.
- `Stop = eq(Equipment_State, 1000)`— Aliran data yang diubah yang berisi status berhenti yang direncanakan mesin.
- `Running = eq(Equipment_State, 1111)`— Aliran data yang diubah yang berisi keadaan operasional normal mesin.

Dengan menggunakan pengukuran mentah dan pengukuran yang ditransformasikan, tentukan [metrik](#) berikut yang menggabungkan data mesin selama interval waktu yang ditentukan. Pilih interval waktu yang sama untuk setiap metrik saat Anda menentukan metrik di bagian ini.

Metrik

- `Successes = sum(Good_Count)`— Jumlah paket yang berhasil diisi selama interval waktu yang ditentukan.
- `Failures = sum(Bad_Count)`— Jumlah paket yang tidak berhasil diisi selama interval waktu yang ditentukan.
- `Idle_Time = statetime(Idle)`— Total waktu idle mesin (dalam detik) per interval waktu yang ditentukan.
- `Fault_Time = statetime(Fault)`— Total waktu kesalahan mesin (dalam detik) per interval waktu yang ditentukan.
- `Stop_Time = statetime(Stop)`— Total waktu berhenti yang direncanakan mesin (dalam detik) per interval waktu yang ditentukan.
- `Run_Time = statetime(Running)`— Total waktu mesin (dalam detik) berjalan tanpa masalah per interval waktu yang ditentukan.
- `Down_Time = Idle_Time + Fault_Time + Stop_Time`— Total downtime mesin (dalam detik) selama interval waktu yang ditentukan, dihitung sebagai jumlah status mesin selain `Run_Time`.
- `Availability = Run_Time / (Run_Time + Down_Time)`— Waktu kerja mesin atau persentase waktu terjadwal bahwa mesin tersedia untuk beroperasi selama interval waktu yang ditentukan.

- $Quality = \text{Successes} / (\text{Successes} + \text{Failures})$ — Persentase mesin dari paket yang berhasil diisi selama interval waktu yang ditentukan.
- $Performance = ((\text{Successes} + \text{Failures}) / \text{Run_Time}) / \text{Ideal_Run_Rate}$ — Kinerja mesin selama interval waktu yang ditentukan sebagai persentase dari laju lari ideal (dalam detik) untuk proses Anda.

Misalnya, Anda `Ideal_Run_Rate` mungkin 60 paket per menit (1 paket per detik). Jika Anda `Ideal_Run_Rate` per menit atau per jam, Anda perlu membaginya dengan faktor konversi unit yang sesuai `Run_Time` karena dalam hitungan detik.

- $OEE = \text{Availability} * \text{Quality} * \text{Performance}$ - Efektivitas peralatan mesin secara keseluruhan selama interval waktu yang ditentukan. Rumus ini menghitung OEE sebagai pecahan dari 1.

Menelan data dari AWS IoT berbagai hal

Pelajari cara menyerap data AWS IoT SiteWise dari armada AWS IoT benda dengan menggunakan bayangan perangkat dalam tutorial ini. Bayangan perangkat adalah objek JSON yang menyimpan informasi status saat ini untuk AWS IoT perangkat. Untuk informasi selengkapnya, lihat [Layanan bayangan perangkat](#) di Panduan AWS IoT Pengembang.

Setelah Anda menyelesaikan tutorial ini, Anda dapat mengatur operasi AWS IoT SiteWise berdasarkan AWS IoT hal-hal. Dengan menggunakan AWS IoT berbagai hal, Anda dapat mengintegrasikan operasi Anda dengan fitur berguna lainnya dari AWS IoT. Misalnya, Anda dapat mengonfigurasi AWS IoT fitur untuk melakukan tugas-tugas berikut:

- Konfigurasi aturan tambahan untuk mengalirkan data ke [AWS IoT Events](#), [Amazon DynamoDB](#), dan lainnya. Layanan AWS Untuk informasi selengkapnya, lihat [Aturan](#) di Panduan AWS IoT Pengembang.
- Indeks, telusuri, dan agregat data perangkat Anda dengan layanan pengindeksan AWS IoT armada. Untuk informasi selengkapnya, lihat [layanan pengindeksan armada](#) di AWS IoT Panduan Pengembang.
- Audit dan amankan perangkat Anda dengan AWS IoT Device Defender. Untuk informasi lebih lanjut, lihat [AWS IoT Device Defender](#) dalam Panduan Pengembang AWS IoT .

Dalam tutorial ini, Anda mempelajari cara menelan data dari bayangan perangkat AWS IoT benda ke aset. AWS IoT SiteWise Untuk melakukannya, Anda membuat satu atau beberapa AWS IoT hal dan

menjalankan skrip yang memperbarui bayangan perangkat setiap hal dengan data penggunaan CPU dan memori. Anda menggunakan data penggunaan CPU dan memori dalam tutorial ini untuk meniru data sensor yang realistis. Kemudian, Anda membuat aturan dengan AWS IoT SiteWise tindakan yang mengirimkan data ini ke aset AWS IoT SiteWise setiap kali bayangan perangkat diperbarui. Untuk informasi selengkapnya, lihat [Menelan data menggunakan aturan AWS IoT Core](#).

Topik

- [Prasyarat](#)
- [Langkah 1: Membuat AWS IoT kebijakan](#)
- [Langkah 2: Membuat dan mengkonfigurasi sesuatu AWS IoT](#)
- [Langkah 3: Membuat model aset perangkat](#)
- [Langkah 4: Membuat model aset armada perangkat](#)
- [Langkah 5: Membuat dan mengonfigurasi aset perangkat](#)
- [Langkah 6: Membuat dan mengonfigurasi aset armada perangkat](#)
- [Langkah 7: Membuat aturan di AWS IoT Core untuk mengirim data ke aset perangkat](#)
- [Langkah 8: Menjalankan skrip klien perangkat](#)
- [Langkah 9: Membersihkan sumber daya setelah tutorial](#)

Prasyarat

Untuk menyelesaikan tutorial ini, Anda memerlukan hal berikut:

- Sebuah Akun AWS. Jika Anda tidak memilikinya, lihat [Menyiapkan sebuah Akun AWS](#).
- Sebuah komputer pengembangan yang berjalan Windows/macOS/Linux, atau Unix untuk mengakses AWS Management Console. Untuk informasi lebih lanjut, lihat [Memulai dengan AWS Management Console](#).
- Pengguna AWS Identity and Access Management (IAM) dengan izin administrator.
- Python3 diinstal pada komputer pengembangan Anda atau diinstal pada perangkat yang ingin Anda daftarkan sebagai AWS IoT sesuatu.

Langkah 1: Membuat AWS IoT kebijakan

Dalam prosedur ini, buat AWS IoT kebijakan yang memungkinkan AWS IoT barang-barang Anda mengakses sumber daya yang digunakan dalam tutorial ini.

Untuk membuat AWS IoT kebijakan

1. Masuk ke [AWS Management Console](#).
2. Tinjau [AWS Wilayah](#) di AWS IoT SiteWise mana didukung. Beralih ke salah satu Wilayah yang didukung ini, jika perlu.
3. Navigasikan ke [konsol AWS IoT](#) tersebut. Jika tombol Connect device muncul, pilih tombol tersebut.
4. Di panel navigasi kiri, pilih Keamanan, lalu pilih Kebijakan.
5. Pilih Buat.
6. Masukkan nama untuk AWS IoT kebijakan (misalnya, **SiteWiseTutorialDevicePolicy**).
7. Di bawah Dokumen kebijakan, pilih JSON untuk memasukkan kebijakan berikut dalam formulir JSON. Ganti *region* dan *account-id* dengan Region dan ID akun Anda, seperti **us-east-1** dan **123456789012**

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:Connect",
      "Resource": "arn:aws:iot:region:account-id:client/SiteWiseTutorialDevice*"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Publish",
      "Resource": [
        "arn:aws:iot:region:account-id:topic/$aws/things/
        ${iot:Connection.Thing.ThingName}/shadow/update",
        "arn:aws:iot:region:account-id:topic/$aws/things/
        ${iot:Connection.Thing.ThingName}/shadow/delete",
        "arn:aws:iot:region:account-id:topic/$aws/things/
        ${iot:Connection.Thing.ThingName}/shadow/get"
      ]
    },
    {
      "Effect": "Allow",
      "Action": "iot:Receive",
      "Resource": [
        "arn:aws:iot:region:account-id:topic/$aws/things/
        ${iot:Connection.Thing.ThingName}/shadow/update/accepted",
```



```

    "arn:aws:iot:region:account-id:topic/$aws/things/
    ${iot:Connection.Thing.ThingName}/shadow/delete/accepted",
    "arn:aws:iot:region:account-id:topic/$aws/things/
    ${iot:Connection.Thing.ThingName}/shadow/get/accepted",
    "arn:aws:iot:region:account-id:topic/$aws/things/
    ${iot:Connection.Thing.ThingName}/shadow/update/rejected",
    "arn:aws:iot:region:account-id:topic/$aws/things/
    ${iot:Connection.Thing.ThingName}/shadow/delete/rejected"
  ]
},
{
  "Effect": "Allow",
  "Action": "iot:Subscribe",
  "Resource": [
    "arn:aws:iot:region:account-id:topicfilter/$aws/things/
    ${iot:Connection.Thing.ThingName}/shadow/update/accepted",
    "arn:aws:iot:region:account-id:topicfilter/$aws/things/
    ${iot:Connection.Thing.ThingName}/shadow/delete/accepted",
    "arn:aws:iot:region:account-id:topicfilter/$aws/things/
    ${iot:Connection.Thing.ThingName}/shadow/get/accepted",
    "arn:aws:iot:region:account-id:topicfilter/$aws/things/
    ${iot:Connection.Thing.ThingName}/shadow/update/rejected",
    "arn:aws:iot:region:account-id:topicfilter/$aws/things/
    ${iot:Connection.Thing.ThingName}/shadow/delete/rejected"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "iot:GetThingShadow",
    "iot:UpdateThingShadow",
    "iot>DeleteThingShadow"
  ],
  "Resource": "arn:aws:iot:region:account-id:thing/SiteWiseTutorialDevice*"
}
]
}

```

Kebijakan ini memungkinkan AWS IoT perangkat Anda untuk membuat koneksi dan berkomunikasi dengan bayangan perangkat menggunakan pesan MQTT. Untuk informasi selengkapnya tentang pesan MQTT, lihat [Apa itu MQTT?](#) . Untuk berinteraksi dengan bayangan perangkat, AWS IoT barang-barang Anda mempublikasikan dan menerima pesan MQTT tentang topik yang dimulai. `$aws/things/thing-name/`

shadow/ Kebijakan ini menggabungkan variabel kebijakan sesuatu yang dikenal sebagai `{iot:Connection.Thing.ThingName}`. Variabel ini menggantikan nama benda yang terhubung di setiap topik. `iot:ConnectPernyataan` tersebut menetapkan batasan pada perangkat mana yang dapat membuat koneksi, memastikan bahwa variabel kebijakan hal hanya dapat menggantikan nama yang dimulai dengan `SiteWiseTutorialDevice`.

Untuk informasi selengkapnya, lihat [Variabel kebijakan Thing](#) di Panduan AWS IoT Pengembang.

Note

Kebijakan ini berlaku untuk hal-hal yang namanya dimulai `SiteWiseTutorialDevice`. Untuk menggunakan nama yang berbeda untuk barang-barang Anda, Anda harus memperbarui kebijakan yang sesuai.

8. Pilih Buat.

Langkah 2: Membuat dan mengkonfigurasi sesuatu AWS IoT

Dalam prosedur ini, Anda membuat dan mengkonfigurasi AWS IoT sesuatu. Anda dapat menunjuk komputer pengembangan Anda sebagai AWS IoT sesuatu. Saat Anda maju, ingatlah bahwa prinsip-prinsip yang Anda pelajari di sini dapat diterapkan pada proyek yang sebenarnya. Anda memiliki fleksibilitas untuk membuat dan mengatur AWS IoT hal-hal di perangkat apa pun yang mampu menjalankan AWS IoT SDK, termasuk AWS IoT Greengrass dan FreeRTOS. Untuk informasi selengkapnya, lihat [AWS IoT SDK](#) di Panduan AWS IoT Pengembang.

Untuk membuat dan mengkonfigurasi AWS IoT sesuatu

1. Buka baris perintah dan jalankan perintah berikut untuk membuat direktori untuk tutorial ini.


```
mkdir iot-sitewise-rule-tutorial
cd iot-sitewise-rule-tutorial
```

2. Jalankan perintah berikut untuk membuat direktori untuk sertifikat barang Anda.

```
mkdir device1
```

Jika Anda membuat hal-hal tambahan, tambahkan nomor dalam nama direktori yang sesuai untuk melacak sertifikat mana yang termasuk dalam hal itu.

3. Navigasikan ke [konsol AWS IoT](#) tersebut.
4. Di panel navigasi kiri, pilih Semua perangkat di bagian Kelola. Lalu, pilih Berbagai hal.
5. Jika kotak dialog Anda belum memiliki hal apa pun muncul, pilih Buat sesuatu. Jika tidak, pilih Buat sesuatu.
6. Pada halaman Creating things, pilih Create a single dan kemudian pilih Next.
7. Pada halaman Tentukan properti benda, masukkan nama untuk AWS IoT barang Anda (misalnya, **SiteWiseTutorialDevice1**) lalu pilih Berikutnya. Jika Anda membuat hal-hal tambahan, tambahkan nomor dalam nama benda yang sesuai.

 Important

Nama benda harus sesuai dengan nama yang digunakan dalam kebijakan yang Anda buat di Langkah 1: Membuat AWS IoT kebijakan. Jika tidak, perangkat Anda tidak dapat terhubung AWS IoT.

8. Pada halaman Konfigurasi sertifikat perangkat - opsional, pilih Buat otomatis sertifikat baru (disarankan) pilih Berikutnya. Sertifikat memungkinkan AWS IoT untuk mengidentifikasi perangkat Anda dengan aman.
9. Pada halaman Lampirkan kebijakan ke sertifikat - opsional, pilih kebijakan yang Anda buat di Langkah 1: Membuat AWS IoT kebijakan dan pilih Buat sesuatu.
10. Pada kotak dialog Unduh sertifikat dan kunci, lakukan hal berikut:
 - a. Pilih tautan Unduh untuk mengunduh sertifikat barang Anda, kunci publik, dan kunci pribadi. Simpan ketiga file ke direktori yang Anda buat untuk sertifikat barang Anda (misalnya, `iot-sitewise-rule-tutorial/device1`).

 Important

Ini adalah satu-satunya saat Anda dapat mengunduh sertifikat dan kunci barang Anda, yang Anda perlukan agar perangkat Anda berhasil terhubung AWS IoT.

- b. Pilih tautan Unduh untuk mengunduh sertifikat CA root. Simpan sertifikat CA root ke file `iot-sitewise-rule-tutorial`. Kami merekomendasikan mengunduh Amazon Root CA 1.

11. Pilih Selesai.

Anda sekarang telah mendaftarkan AWS IoT sesuatu di komputer Anda. Ambil salah satu langkah berikut:

- Lanjutkan ke Langkah 3: Membuat model aset perangkat tanpa membuat AWS IoT hal-hal tambahan. Anda dapat menyelesaikan tutorial ini hanya dengan satu hal.
- Ulangi langkah-langkah di bagian ini di komputer atau perangkat lain untuk membuat lebih banyak AWS IoT hal. Untuk tutorial ini, kami sarankan Anda mengikuti opsi ini sehingga Anda dapat menelan data penggunaan CPU dan memori yang unik dari beberapa perangkat.
- Ulangi langkah-langkah di bagian ini pada perangkat yang sama (komputer Anda) untuk membuat lebih banyak AWS IoT hal. Setiap AWS IoT benda menerima data penggunaan CPU dan memori yang serupa dari komputer Anda, jadi gunakan pendekatan ini untuk menunjukkan pengambilan data non-unik dari beberapa perangkat.

Langkah 3: Membuat model aset perangkat

Dalam prosedur ini, Anda membuat model aset AWS IoT SiteWise untuk mewakili perangkat Anda yang mengalirkan data penggunaan CPU dan memori. Untuk memproses data dalam aset yang mewakili kelompok perangkat, model aset menerapkan informasi yang konsisten di beberapa aset dengan jenis yang sama. Untuk informasi selengkapnya, lihat [Pemodelan aset industri](#).

Untuk membuat model aset yang mewakili perangkat

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Model.
3. Pilih Buat model.
4. Di bawah Detail model, masukkan nama untuk model Anda. Misalnya, **SiteWise Tutorial Device Model**.
5. Di bawah definisi Pengukuran, lakukan hal berikut:
 - a. Di Nama, masukkan **CPU Usage**.
 - b. Di Unit, masukkan %.
 - c. Biarkan tipe Data sebagai Ganda.

Properti pengukuran mewakili aliran data mentah perangkat. Untuk informasi selengkapnya, lihat [Mendefinisikan aliran data dari peralatan \(pengukuran\)](#).

6. Pilih Tambahkan pengukuran baru untuk menambahkan properti pengukuran kedua.
7. Di baris kedua di bawah Definisi pengukuran, lakukan hal berikut:
 - a. Di Nama, masukkan **Memory Usage**.
 - b. Di Unit, masukkan%.
 - c. Biarkan tipe Data sebagai Ganda.
8. Di bawah definisi Metrik, lakukan hal berikut:
 - a. Di Nama, masukkan **Average CPU Usage**.
 - b. Dalam Formula, masukkan`avg(CPU Usage)`. Pilih CPU Usagedari daftar pelengkapan otomatis saat muncul.
 - c. Dalam Interval waktu, masukkan**5 minutes**.

Properti metrik mendefinisikan perhitungan agregasi yang memproses semua titik data input selama interval dan mengeluarkan satu titik data per interval. Properti metrik ini menghitung rata-rata penggunaan CPU setiap perangkat setiap 5 menit. Untuk informasi selengkapnya, lihat [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#).

9. Pilih Tambahkan metrik baru untuk menambahkan properti metrik kedua.
10. Di baris kedua di bawah Definisi metrik, lakukan hal berikut:
 - a. Di Nama, masukkan **Average Memory Usage**.
 - b. Dalam Formula, masukkan`avg(Memory Usage)`. Pilih Memory Usagedari daftar pelengkapan otomatis saat muncul.
 - c. Dalam Interval waktu, masukkan**5 minutes**.

Properti metrik ini menghitung penggunaan memori rata-rata setiap perangkat setiap 5 menit.

11. (Opsional) Tambahkan metrik tambahan lain yang ingin Anda hitung per perangkat. Beberapa fungsi menarik termasuk `min` dan`max`. Untuk informasi selengkapnya, lihat [Menggunakan ekspresi rumus](#). Pada Langkah 4: Membuat model aset armada perangkat, Anda membuat aset induk yang dapat menghitung metrik menggunakan data dari seluruh armada perangkat Anda.
12. Pilih Buat model.

Langkah 4: Membuat model aset armada perangkat

Dalam prosedur ini, Anda membuat model aset AWS IoT SiteWise untuk melambungkan koleksi perangkat Anda. Dalam model aset ini, Anda membuat struktur yang memungkinkan Anda menautkan banyak aset perangkat ke satu aset armada menyeluruh. Setelah itu, Anda menguraikan metrik dalam model aset armada untuk mengkonsolidasikan data dari semua aset perangkat yang terhubung. Pendekatan ini memberi Anda wawasan komprehensif tentang kinerja kolektif seluruh armada Anda.

Untuk membuat model aset yang mewakili armada perangkat

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
 2. Di panel navigasi kiri, pilih Model.
 3. Pilih Buat model.
 4. Di bawah Detail model, masukkan nama untuk model Anda. Misalnya, **SiteWise Tutorial Device Fleet Model**.
 5. Di bawah definisi Hierarki, lakukan hal berikut:
 - a. Dalam nama Hierarki, masukkan **Device**.
 - b. Dalam model Hierarki, pilih model aset perangkat Anda (**SiteWise Tutorial Device Model**).
- Hirarki mendefinisikan hubungan antara model aset induk (armada) dan model aset anak (perangkat). Aset induk dapat mengakses data properti aset anak. Saat Anda membuat aset nanti, Anda perlu mengaitkan aset anak ke aset induk sesuai dengan definisi hierarki dalam model aset induk. Untuk informasi selengkapnya, lihat [Mendefinisikan hierarki model aset](#).
6. Di bawah definisi Metrik, lakukan hal berikut:
 - a. Di Nama, masukkan **Average CPU Usage**.
 - b. Dalam Formula, masukkan **avg(Device | Average CPU Usage)**. Ketika daftar pelengkapan otomatis muncul, pilih Device untuk memilih hierarki, lalu pilih Average CPU Usage untuk memilih metrik dari aset perangkat yang Anda buat sebelumnya.
 - c. Dalam Interval waktu, masukkan **5 minutes**.

Properti metrik ini menghitung penggunaan CPU rata-rata dari semua aset perangkat yang terkait dengan aset armada melalui **Device** hierarki.

7. Pilih Tambahkan metrik baru untuk menambahkan properti metrik kedua.
8. Di baris kedua di bawah Definisi metrik, lakukan hal berikut:
 - a. Di Nama, masukkan **Average Memory Usage**.
 - b. Dalam Formula, masukkan **avg(Device | Average Memory Usage)**. Ketika daftar pelengkapan otomatis muncul, pilih Device untuk memilih hierarki, lalu pilih Average Memory Usage untuk memilih metrik dari aset perangkat yang Anda buat sebelumnya.
 - c. Dalam Interval waktu, masukkan **5 minutes**.

Properti metrik ini menghitung penggunaan memori rata-rata dari semua aset perangkat yang terkait dengan aset armada melalui **Device** hierarki.

9. (Opsional) Tambahkan metrik tambahan lain yang ingin Anda hitung di seluruh armada perangkat Anda.
10. Pilih Buat model.

Langkah 5: Membuat dan mengonfigurasi aset perangkat

Dalam prosedur ini, Anda menghasilkan aset perangkat yang didasarkan pada model aset perangkat Anda. Kemudian, Anda menentukan alias properti untuk setiap properti pengukuran. Alias properti adalah string unik yang mengidentifikasi properti aset. Kemudian, Anda dapat mengidentifikasi properti untuk upload data dengan menggunakan alias, bukan ID aset dan ID properti. Untuk informasi selengkapnya, lihat [Memetakan aliran data industri ke properti aset](#).

Untuk membuat aset perangkat dan menentukan alias properti

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Aset.
3. Pilih Buat aset.
4. Di bawah Informasi model, pilih model aset perangkat Anda **SiteWise Tutorial Device Model**.
5. Di bawah informasi Aset, masukkan nama untuk aset Anda. Misalnya, **SiteWise Tutorial Device 1**.
6. Pilih Buat aset.
7. Untuk aset perangkat baru Anda, pilih Edit.

8. Di bawah CPU Usage, masukkan **/tutorial/device/SiteWiseTutorialDevice1/cpu** sebagai alias properti. Anda menyertakan nama AWS IoT benda dalam alias properti, sehingga Anda dapat menelan data dari semua perangkat Anda menggunakan satu AWS IoT aturan.
9. Di bawah Memory Usage, masukkan **/tutorial/device/SiteWiseTutorialDevice1/memory** sebagai alias properti.
10. Pilih Simpan.

Jika Anda membuat beberapa AWS IoT hal sebelumnya, ulangi langkah 3 hingga 10 untuk setiap perangkat, dan tambahkan nomor dalam nama aset dan alias properti yang sesuai. Misalnya, nama aset perangkat kedua harus **SiteWise Tutorial Device 2**, dan alias propertinya harus **/tutorial/device/SiteWiseTutorialDevice2/cpu**, dan **/tutorial/device/SiteWiseTutorialDevice2/memory**.

Langkah 6: Membuat dan mengonfigurasi aset armada perangkat

Dalam prosedur ini, Anda membentuk aset armada perangkat yang berasal dari model aset armada perangkat Anda. Kemudian, Anda menautkan aset perangkat individual Anda ke aset armada. Asosiasi ini memungkinkan properti metrik aset armada untuk mengkompilasi dan menganalisis data dari beberapa perangkat. Data ini memberi Anda pandangan konsolidasi tentang kinerja kolektif seluruh armada.

Untuk membuat aset armada perangkat dan mengaitkan aset perangkat

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Aset.
3. Pilih Buat aset.
4. Di bawah Informasi model, pilih model aset armada perangkat Anda **SiteWise Tutorial Device Fleet Model**.
5. Di bawah informasi Aset, masukkan nama untuk aset Anda. Misalnya, **SiteWise Tutorial Device Fleet 1**.
6. Pilih Buat aset.
7. Untuk aset armada perangkat baru Anda, pilih Edit.
8. Di bawah Aset yang terkait dengan aset ini, pilih Tambahkan aset terkait dan lakukan hal berikut:

- a. Di bawah Hierarki, pilih Device. Hirarki ini mengidentifikasi hubungan hierarkis antara perangkat dan aset armada perangkat. Anda mendefinisikan hierarki ini dalam model aset armada perangkat sebelumnya dalam tutorial ini.
 - b. Di bawah Aset, pilih aset perangkat Anda, SiteWise Tutorial Device1.
9. (Opsional) Jika Anda membuat beberapa aset perangkat lebih awal, ulangi langkah 8 hingga 10 untuk setiap aset perangkat yang Anda buat.
 10. Pilih Simpan.

Anda sekarang akan melihat aset perangkat Anda diatur sebagai hierarki.

Langkah 7: Membuat aturan di AWS IoT Core untuk mengirim data ke aset perangkat

Dalam prosedur ini, Anda membuat aturan di AWS IoT Core. Aturan ini dirancang untuk menafsirkan pesan notifikasi dari bayangan perangkat dan mengirimkan data ke aset perangkat Anda di AWS IoT SiteWise. Setiap kali bayangan perangkat Anda diperbarui, AWS IoT mengirimkan pesan MQTT. Anda dapat membuat aturan yang mengambil tindakan saat bayangan perangkat berubah berdasarkan pesan MQTT. Dalam hal ini, tujuannya adalah untuk menangani pesan pembaruan, mengekstrak nilai properti, dan mengirimkannya ke aset perangkat Anda AWS IoT SiteWise.

Untuk membuat aturan dengan AWS IoT SiteWise tindakan

1. Navigasikan ke [konsol AWS IoT](#) tersebut.
2. Di panel navigasi kiri, pilih Perutean pesan, lalu pilih Aturan.
3. Pilih Buat aturan.
4. Masukkan nama dan deskripsi untuk aturan Anda dan pilih Berikutnya.
5. Masukkan pernyataan SQL berikut dan pilih Berikutnya.


```
SELECT
  *
FROM
  '$aws/things/+/shadow/update/accepted'
WHERE
  startsWith(topic(3), "SiteWiseTutorialDevice")
```

Pernyataan kueri aturan ini berfungsi karena layanan bayangan perangkat menerbitkan pembaruan bayangan ke `$aws/things/thingName/shadow/update/accepted`. Untuk informasi selengkapnya tentang bayangan [perangkat](#), lihat [Layanan bayangan perangkat](#) di Panduan AWS IoT Pengembang.

Dalam WHERE klausa, pernyataan kueri aturan ini menggunakan `topic(3)` fungsi untuk mendapatkan nama benda dari segmen ketiga topik. Kemudian, pernyataan tersebut menyaring perangkat yang memiliki nama yang tidak cocok dengan perangkat tutorial. Untuk informasi selengkapnya tentang AWS IoT SQL, lihat [referensi AWS IoT SQL](#) di Panduan AWS IoT Pengembang.

6. Di bawah Tindakan aturan, pilih Kirim data pesan ke properti aset AWS IoT SiteWise dan lakukan hal berikut:
 - a. Pilih Berdasarkan alias properti.
 - b. Dalam alias Properti, masukkan `/tutorial/device/${topic(3)}/cpu`.

`${...}` Sintaks adalah template substitusi. AWS IoT mengevaluasi isi dalam kawat gigi. Template substitusi ini menarik nama benda dari topik untuk membuat alias unik untuk setiap hal. Untuk informasi selengkapnya, lihat [Templat substitusi](#) di Panduan AWS IoT Pengembang.

 Note

Karena ekspresi dalam templat substitusi dievaluasi secara terpisah dari SELECT pernyataan, Anda tidak dapat menggunakan templat substitusi untuk mereferensikan alias yang dibuat menggunakan klausa. AS Anda hanya dapat mereferensikan informasi yang ada di muatan asli, selain fungsi dan operator yang didukung.

- c. Di ID Entri - opsional, masukkan `${concat(topic(3), "-cpu-", floor(state.reported.timestamp))}`.


ID entri secara unik mengidentifikasi setiap upaya entri nilai. Jika entri mengembalikan kesalahan, Anda dapat menemukan ID entri di output kesalahan untuk memecahkan masalah. Template substitusi dalam ID entri ini menggabungkan nama benda dan stempel waktu perangkat yang dilaporkan. Misalnya, ID entri yang dihasilkan mungkin terlihat seperti `SiteWiseTutorialDevice1-cpu-1579808494`.

- d. Dalam Waktu dalam hitungan detik, masukkan **`${floor(state.reported.timestamp)}`**.

Template substitusi ini menghitung waktu dalam hitungan detik dari stempel waktu yang dilaporkan perangkat. Dalam tutorial ini, perangkat melaporkan stempel waktu dalam hitungan detik dalam waktu epoch Unix sebagai nomor floating point.

- e. Di Offset dalam nanos - opsional, masukkan. **`${floor((state.reported.timestamp % 1) * 1E9)}`**

Template substitusi ini menghitung offset nanodetik dari waktu dalam hitungan detik dengan mengubah bagian desimal dari stempel waktu yang dilaporkan perangkat.

 Note

AWS IoT SiteWise mengharuskan data Anda memiliki stempel waktu saat ini dalam waktu epoch Unix. Jika perangkat Anda tidak melaporkan waktu secara akurat, Anda bisa mendapatkan waktu saat ini dari mesin AWS IoT aturan dengan [stempel waktu \(\)](#). Fungsi ini melaporkan waktu dalam milidetik, jadi Anda harus memperbarui parameter waktu tindakan aturan Anda ke nilai berikut:

- Dalam Waktu dalam hitungan detik, masukkan **`${floor(timestamp() / 1E3)}`**.
- Di Offset di nanos, masukkan. **`${(timestamp() % 1E3) * 1E6}`**

- f. Di Tipe data, pilih Ganda.

Tipe data ini harus cocok dengan tipe data properti aset yang Anda tentukan dalam model aset.

- g. Dalam Value (Nilai), masukkan **`${state.reported.cpu}`**. Dalam template substitusi, Anda menggunakan `.` operator untuk mengambil nilai dari dalam struktur JSON.
- h. Pilih Tambah entri untuk menambahkan entri baru untuk properti penggunaan memori, dan selesaikan langkah-langkah berikut lagi untuk properti itu:
- Pilih Berdasarkan alias properti.
 - Dalam alias Properti, masukkan **`/tutorial/device/${topic(3)}/memory`**.
 - Di ID Entri - opsional, masukkan **`${concat(topic(3), "-memory-", floor(state.reported.timestamp))}`**.

- iv. Dalam Waktu dalam hitungan detik, masukkan `floor(state.reported.timestamp)`.
 - v. Di Offset dalam nanos - opsional, masukkan `floor((state.reported.timestamp % 1) * 1E9)`
 - vi. Di Tipe data, pilih Ganda.
 - vii. Dalam Value (Nilai), masukkan `state.reported.memory`.
 - i. Di bawah Peran IAM, pilih Buat peran baru untuk membuat peran IAM untuk tindakan aturan ini. Peran ini memungkinkan AWS IoT untuk mendorong data ke properti di aset armada perangkat Anda dan hierarki asetnya.
 - j. Masukkan nama peran dan pilih Buat.
7. (Opsional) Konfigurasi tindakan kesalahan yang dapat Anda gunakan untuk memecahkan masalah aturan Anda. Untuk informasi selengkapnya, lihat [Memecahkan masalah aturan](#).
 8. Pilih Berikutnya.
 9. Tinjau pengaturan dan pilih Buat untuk membuat aturan.

Langkah 8: Menjalankan skrip klien perangkat

Untuk tutorial ini, Anda tidak menggunakan perangkat yang sebenarnya untuk melaporkan data. Sebagai gantinya, Anda menjalankan skrip untuk memperbarui bayangan perangkat AWS IoT benda Anda dengan CPU dan penggunaan memori untuk meniru data sensor nyata. Untuk menjalankan skrip, Anda harus terlebih dahulu menginstal Python paket yang diperlukan. Dalam prosedur ini, Anda menginstal Python paket yang diperlukan dan kemudian menjalankan skrip klien perangkat.

Untuk mengkonfigurasi dan menjalankan skrip klien perangkat

1. Navigasikan ke [konsol AWS IoT](#) tersebut.
2. Di bagian bawah panel navigasi kiri, pilih Pengaturan.
3. Simpan titik akhir kustom Anda untuk digunakan dengan skrip klien perangkat. Anda menggunakan titik akhir ini untuk berinteraksi dengan bayangan benda Anda. Titik akhir ini unik untuk akun Anda di Wilayah saat ini.

Titik akhir kustom Anda akan terlihat seperti contoh berikut.

```
identifier.iot.region.amazonaws.com
```

4. Buka baris perintah dan jalankan perintah berikut untuk menavigasi ke direktori tutorial yang Anda buat sebelumnya.

```
cd iot-sitewise-rule-tutorial
```

5. Jalankan perintah berikut untuk menginstal file AWS IoT Device SDK for Python.

```
pip3 install AWSIoTPythonSDK
```

Untuk informasi selengkapnya, lihat [AWS IoT Device SDK for Python](#) di Panduan AWS IoT Pengembang

6. Jalankan perintah berikut untuk menginstal psutil, proses lintas platform dan perpustakaan utilitas sistem.

```
pip3 install psutil
```

Untuk informasi selengkapnya, lihat [psutil](#) di Indeks Paket Python.

7. Buat file yang disebut `thing_performance.py` dalam `iot-sitewise-rule-tutorial` direktori dan kemudian salin kode Python berikut ke dalam file.

```
import AWSIoTPythonSDK.MQTTLib as AWSIoTPyMQTT

import json
import psutil
import argparse
import logging
import time

# Configures the argument parser for this program.
def configureParser():
    parser = argparse.ArgumentParser()
    parser.add_argument(
        "-e",
        "--endpoint",
        action="store",
        required=True,
        dest="host",
        help="Your AWS IoT custom endpoint",
    )
```

```
parser.add_argument(
    "-r",
    "--rootCA",
    action="store",
    required=True,
    dest="rootCAPath",
    help="Root CA file path",
)
parser.add_argument(
    "-c",
    "--cert",
    action="store",
    required=True,
    dest="certificatePath",
    help="Certificate file path",
)
parser.add_argument(
    "-k",
    "--key",
    action="store",
    required=True,
    dest="privateKeyPath",
    help="Private key file path",
)
parser.add_argument(
    "-p",
    "--port",
    action="store",
    dest="port",
    type=int,
    default=8883,
    help="Port number override",
)
parser.add_argument(
    "-n",
    "--thingName",
    action="store",
    required=True,
    dest="thingName",
    help="Targeted thing name",
)
parser.add_argument(
    "-d",
    "--requestDelay",
```

```
        action="store",
        dest="requestDelay",
        type=float,
        default=1,
        help="Time between requests (in seconds)",
    )
    parser.add_argument(
        "-v",
        "--enableLogging",
        action="store_true",
        dest="enableLogging",
        help="Enable logging for the AWS IoT Device SDK for Python",
    )
    return parser

# An MQTT shadow client that uploads device performance data to AWS IoT at a
# regular interval.
class PerformanceShadowClient:
    def __init__(
        self,
        thingName,
        host,
        port,
        rootCAPath,
        privateKeyPath,
        certificatePath,
        requestDelay,
    ):
        self.thingName = thingName
        self.host = host
        self.port = port
        self.rootCAPath = rootCAPath
        self.privateKeyPath = privateKeyPath
        self.certificatePath = certificatePath
        self.requestDelay = requestDelay

    # Updates this thing's shadow with system performance data at a regular
    # interval.
    def run(self):
        print("Connecting MQTT client for {}".format(self.thingName))
        mqttClient = self.configureMQTTClient()
        mqttClient.connect()
        print("MQTT client for {} connected".format(self.thingName))
```

```
deviceShadowHandler = mqttClient.createShadowHandlerWithName(
    self.thingName, True
)

print("Running performance shadow client for {}...
\n".format(self.thingName))
while True:
    performance = self.readPerformance()
    print("[{}]" .format(self.thingName))
    print("CPU:\t{}%".format(performance["cpu"]))
    print("Memory:\t{}%\n".format(performance["memory"]))
    payload = {"state": {"reported": performance}}
    deviceShadowHandler.shadowUpdate(
        json.dumps(payload), self.shadowUpdateCallback, 5
    )
    time.sleep(args.requestDelay)

# Configures the MQTT shadow client for this thing.
def configureMQTTClient(self):
    mqttClient = AWSIoTPyMQTT.AWSIoTMQTTShadowClient(self.thingName)
    mqttClient.configureEndpoint(self.host, self.port)
    mqttClient.configureCredentials(
        self.rootCAPath, self.privateKeyPath, self.certificatePath
    )
    mqttClient.configureAutoReconnectBackoffTime(1, 32, 20)
    mqttClient.configureConnectDisconnectTimeout(10)
    mqttClient.configureMQTTOperationTimeout(5)
    return mqttClient

# Returns the local device's CPU usage, memory usage, and timestamp.
def readPerformance(self):
    cpu = psutil.cpu_percent()
    memory = psutil.virtual_memory().percent
    timestamp = time.time()
    return {"cpu": cpu, "memory": memory, "timestamp": timestamp}

# Prints the result of a shadow update call.
def shadowUpdateCallback(self, payload, responseStatus, token):
    print("[{}]" .format(self.thingName))
    print("Update request {} {}\n".format(token, responseStatus))

# Configures debug logging for the AWS IoT Device SDK for Python.
def configureLogging():
```



```
logger = logging.getLogger("AWSIoTPythonSDK.core")
logger.setLevel(logging.DEBUG)
streamHandler = logging.StreamHandler()
formatter = logging.Formatter(
    "%(asctime)s - %(name)s - %(levelname)s - %(message)s"
)
streamHandler.setFormatter(formatter)
logger.addHandler(streamHandler)

# Runs the performance shadow client with user arguments.
if __name__ == "__main__":
    parser = configureParser()
    args = parser.parse_args()
    if args.enableLogging:
        configureLogging()
    thingClient = PerformanceShadowClient(
        args.thingName,
        args.host,
        args.port,
        args.rootCAPath,
        args.privateKeyPath,
        args.certificatePath,
        args.requestDelay,
    )
    thingClient.run()
```

8. Jalankan `thing_performance.py` dari baris perintah dengan parameter berikut:

- `-n, --thingName` - Nama benda Anda, seperti **SiteWiseTutorialDevice1**.
- `-e, --endpoint` — AWS IoT Titik akhir kustom Anda yang Anda simpan sebelumnya dalam prosedur ini.
- `-r, --rootCA` — Jalur ke sertifikat CA AWS IoT root Anda.
- `-c, --cert` — Jalan menuju sertifikat AWS IoT barang Anda.
- `-k, --key` — Jalan menuju kunci pribadi sertifikat AWS IoT barang Anda.
- `-d, --requestDelay` — (Opsional) Waktu dalam hitungan detik untuk menunggu di antara setiap pembaruan bayangan perangkat. Default ke 1 detik.
- `-v, --enableLogging` — (Opsional) Jika parameter ini ada, skrip mencetak pesan debug dari AWS IoT Device SDK for Python.

Perintah Anda akan terlihat mirip dengan contoh berikut.

```
python3 thing_performance.py \  
  --thingName SiteWiseTutorialDevice1 \  
  --endpoint identifier.iot.region.amazonaws.com \  
  --rootCA AmazonRootCA1.pem \  
  --cert device1/thing-id-certificate.pem.crt \  
  --key device1/thing-id-private.pem.key
```

Jika Anda menjalankan skrip untuk AWS IoT hal-hal tambahan, perbarui nama benda dan direktori sertifikat yang sesuai.

9. Coba buka dan tutup program di perangkat Anda untuk melihat bagaimana penggunaan CPU dan memori berubah. Skrip mencetak setiap pembacaan penggunaan CPU dan memori. Jika skrip mengunggah data ke layanan bayangan perangkat berhasil, output skrip akan terlihat seperti contoh berikut.

```
[SiteWiseTutorialDevice1]  
CPU:    24.6%  
Memory: 85.2%  
  
[SiteWiseTutorialDevice1]  
Update request e6686e44-fca0-44db-aa48-3ca81726f3e3 accepted
```

10. Ikuti langkah-langkah ini untuk memverifikasi bahwa skrip memperbarui bayangan perangkat:
 - a. Navigasikan ke [konsol AWS IoT](#) tersebut.
 - b. Di panel navigasi kiri, pilih Semua perangkat dan kemudian pilih Things.
 - c. Pilih barang Anda, SiteWiseTutorialDevice.
 - d. Pilih tab Device Shadows, pilih Classic Shadow, dan verifikasi bahwa status Shadow terlihat seperti contoh berikut.

```
{  
  "reported": {  
    "cpu": 24.6,  
    "memory": 85.2,  
    "timestamp": 1579567542.2835066  
  }  
}
```

Jika status bayangan benda Anda kosong atau tidak terlihat seperti contoh sebelumnya, periksa apakah skrip sedang berjalan dan berhasil terhubung AWS IoT. Jika skrip terus habis saat menghubungkan ke AWS IoT, periksa apakah [kebijakan hal](#) Anda dikonfigurasi sesuai dengan tutorial ini.

11. Ikuti langkah-langkah ini untuk memverifikasi bahwa tindakan aturan mengirim data ke AWS IoT SiteWise:
 - a. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
 - b. Di panel navigasi kiri, pilih Aset.
 - c. Pilih tanda panah di samping aset armada perangkat Anda (SiteWise Tutorial Device Fleet 11) untuk memperluas hierarki asetnya, lalu pilih aset perangkat Anda (SiteWise Tutorial Device1).
 - d. Pilih Pengukuran.
 - e. Verifikasi bahwa sel nilai terbaru memiliki nilai untuk CPU Usagedan Memory Usage properti.

Measurements				
Name	Alias	Notification status	Notification topic	Latest value
CPU Usage	/tutorial/device/SiteWiseTutorialDevice1/cpu	⊖ Disabled	-	24.6
Memory Usage	/tutorial/device/SiteWiseTutorialDevice1/memory	⊖ Disabled	-	85.2

- f. Jika Memory Usage properti CPU Usagedan tidak memiliki nilai terbaru, segarkan halaman. Jika nilai tidak muncul setelah beberapa menit, lihat [Memecahkan masalah aturan](#).
12. Anda telah menyelesaikan tutorial ini. Jika Anda ingin menjelajahi visualisasi langsung data Anda, Anda dapat mengonfigurasi portal di AWS IoT SiteWise Monitor Untuk informasi selengkapnya, lihat [Memantau data dengan AWS IoT SiteWise Monitor](#). Jika tidak, Anda dapat menekan CTRL+C di prompt perintah Anda untuk menghentikan skrip klien perangkat. Tidak mungkin program Python akan mengirim pesan yang cukup untuk dikenakan biaya, tetapi ini adalah praktik terbaik untuk menghentikan program ketika Anda selesai.

Langkah 9: Membersihkan sumber daya setelah tutorial

Setelah Anda menyelesaikan tutorial tentang menelan data dari AWS IoT berbagai hal, bersihkan sumber daya Anda untuk menghindari biaya tambahan.

Untuk menghapus aset hierarkis di AWS IoT SiteWise

1. Arahkan ke [AWS IoT SiteWise konsol](#)
2. Di panel navigasi kiri, pilih Aset.
3. Saat Anda menghapus aset AWS IoT SiteWise, Anda harus terlebih dahulu memisahkannya.

Selesaikan langkah-langkah berikut untuk memisahkan aset perangkat Anda dari aset armada perangkat Anda:

- a. Pilih aset armada perangkat Anda (SiteWise Tutorial Device Fleet 1).
- b. Pilih Edit.
- c. Di bawah Aset yang terkait dengan aset ini, pilih Putuskan hubungan untuk setiap aset perangkat yang terkait dengan aset armada perangkat ini.
- d. Pilih Simpan.

Sekarang Anda akan melihat aset perangkat Anda tidak lagi terorganisir sebagai hierarki.

4. Pilih aset perangkat Anda (SiteWise Tutorial Device1).
5. Pilih Hapus.
6. Pada dialog konfirmasi, masukkan **Delete** lalu pilih Hapus.
7. Ulangi langkah 4 hingga 6 untuk setiap aset perangkat dan aset armada perangkat (SiteWise Tutorial Device Fleet 1).

Untuk menghapus model aset hierarkis di AWS IoT SiteWise

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Jika Anda belum melakukannya, hapus aset armada perangkat dan perangkat Anda. Untuk informasi lebih lanjut, lihat [prosedur sebelumnya](#). Anda tidak dapat menghapus model jika Anda memiliki aset yang dibuat dari model tersebut.
3. Di panel navigasi kiri, pilih Model.
4. Pilih model aset armada perangkat Anda (SiteWise Tutorial Device Fleet Model).

Saat menghapus model aset hierarkis, mulailah dengan menghapus model aset induk terlebih dahulu.

5. Pilih Hapus.
6. Pada dialog konfirmasi, masukkan **Delete** lalu pilih Hapus.

7. Ulangi langkah 4 hingga 6 untuk model aset perangkat Anda (SiteWise Tutorial Device Model).

Untuk menonaktifkan atau menghapus aturan di AWS IoT Core

1. Navigasikan ke [konsol AWS IoT](#) tersebut.
2. Di panel navigasi kiri, pilih Perutean pesan, lalu pilih Aturan.
3. Pilih aturan Anda dan pilih Hapus.
4. Dalam dialog konfirmasi, masukkan nama aturan dan kemudian pilih Hapus.

Memvisualisasikan dan berbagi data ladang angin di Monitor SiteWise

Tutorial ini menjelaskan cara menggunakan AWS IoT SiteWise Monitor untuk memvisualisasikan dan berbagi data industri melalui aplikasi web yang dikelola, yang dikenal sebagai portal. Setiap portal mencakup proyek, memberi Anda fleksibilitas untuk memilih data mana yang dapat diakses dalam setiap proyek. Kemudian, tentukan orang-orang di organisasi Anda yang dapat mengakses setiap portal. Pengguna Anda masuk ke portal menggunakan AWS IAM Identity Center akun, sehingga Anda dapat menggunakan toko identitas yang ada atau toko yang dikelola oleh AWS.

Anda, dan pengguna Anda dengan izin yang memadai, dapat membuat dasbor di setiap proyek untuk memvisualisasikan data industri Anda dengan cara yang berarti. Kemudian, pengguna Anda dapat melihat dasbor ini untuk dengan cepat mendapatkan wawasan tentang data Anda dan memantau operasi Anda. Anda dapat mengonfigurasi izin administratif atau hanya baca untuk setiap proyek untuk setiap pengguna di perusahaan Anda. Untuk informasi selengkapnya, lihat [Memantau data dengan AWS IoT SiteWise Monitor](#).

Sepanjang tutorial, Anda meningkatkan AWS IoT SiteWise demo, menyediakan contoh dataset untuk ladang angin. Anda mengonfigurasi portal di SiteWise Monitor, membuat proyek, dan dasbor untuk memvisualisasikan data ladang angin. Tutorial ini juga mencakup pembuatan pengguna tambahan, bersama dengan penetapan izin untuk memiliki atau melihat proyek dan dasbor terkait.

Note

Saat Anda menggunakan SiteWise Monitor, Anda dikenakan biaya per pengguna yang masuk ke portal (per bulan). Dalam tutorial ini, Anda membuat tiga pengguna, tetapi Anda hanya perlu masuk dengan satu pengguna. Setelah Anda menyelesaikan tutorial ini, Anda

dikenakan biaya untuk satu pengguna. Untuk informasi selengkapnya, silakan lihat [Harga AWS IoT SiteWise](#).

Topik

- [Prasyarat](#)
- [Langkah 1: Buat portal di SiteWise Monitor](#)
- [Langkah 2: Masuk ke portal](#)
- [Langkah 3: Buat proyek ladang angin](#)
- [Langkah 4: Buat dasbor untuk memvisualisasikan data ladang angin](#)
- [Langkah 5: Jelajahi portal](#)
- [Langkah 6: Bersihkan sumber daya setelah tutorial](#)

Prasyarat

Untuk menyelesaikan tutorial ini, Anda memerlukan hal berikut:

- Sebuah Akun AWS. Jika Anda tidak memilikinya, lihat [Menyiapkan sebuah Akun AWS](#).
- Sebuah komputer pengembangan yang berjalan Windows/macOS/Linux, atau Unix untuk mengakses AWS Management Console. Untuk informasi lebih lanjut, lihat [Memulai dengan AWS Management Console](#).
- Pengguna AWS Identity and Access Management (IAM) dengan izin administrator.
- Demo peternakan AWS IoT SiteWise angin yang sedang berjalan. Ketika Anda mengatur demo, itu mendefinisikan model dan aset AWS IoT SiteWise dan mengalirkan data ke mereka untuk mewakili ladang angin. Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT SiteWise demo](#).
- Jika Anda mengaktifkan Pusat Identitas IAM di akun Anda, masuk ke akun AWS Organizations manajemen Anda. Untuk informasi lebih lanjut, lihat [AWS Organizations terminologi dan konsep](#). Jika Anda belum mengaktifkan IAM Identity Center, Anda akan mengaktifkannya dalam tutorial ini dan menetapkan akun Anda sebagai akun manajemen.

Jika Anda tidak dapat masuk ke akun AWS Organizations manajemen Anda, Anda dapat menyelesaikan sebagian tutorial selama Anda memiliki pengguna Pusat Identitas IAM di organisasi Anda. Dalam hal ini, Anda dapat membuat portal dan dasbor, tetapi Anda tidak dapat membuat pengguna Pusat Identitas IAM baru untuk ditetapkan ke proyek.

Langkah 1: Buat portal di SiteWise Monitor

Dalam prosedur ini, Anda membuat portal di AWS IoT SiteWise Monitor. Setiap portal adalah aplikasi web terkelola yang dapat Anda dan pengguna masuk dengan AWS IAM Identity Center akun. Dengan IAM Identity Center, Anda dapat menggunakan toko identitas perusahaan Anda yang ada atau membuatnya dikelola oleh AWS. Karyawan perusahaan Anda dapat masuk tanpa membuat terpisah Akun AWS.

Untuk membuat portal

1. Masuk ke [konsol AWS IoT SiteWise](#) tersebut.
2. Tinjau [AWS IoT SiteWise titik akhir dan kuota](#) di mana AWS IoT SiteWise didukung dan alihkan Wilayah, jika diperlukan. Anda harus menjalankan AWS IoT SiteWise demo di Wilayah yang sama.
3. Di panel navigasi kiri, pilih Portal.
4. Pilih Buat portal.
5. Jika Anda sudah mengaktifkan IAM Identity Center, lewati ke langkah 6. Jika tidak, selesaikan langkah-langkah berikut untuk mengaktifkan Pusat Identitas IAM:
 - a. Pada halaman Aktifkan AWS IAM Identity Center (SSO), masukkan alamat Email Anda, Nama depan, dan Nama belakang untuk membuat pengguna Pusat Identitas IAM bagi Anda untuk menjadi administrator portal. Gunakan alamat email yang dapat Anda akses sehingga Anda dapat menerima email untuk mengatur kata sandi bagi pengguna Pusat Identitas IAM baru Anda.

Di portal, administrator portal membuat proyek dan menugaskan pengguna ke proyek. Anda dapat membuat lebih banyak pengguna nanti.

AWS IoT SiteWise > Monitor > Portals > Create portal

Step 1
Enable SSO

Step 2
Portal configuration

Step 3
Invite administrators

Step 4
Assign users

Enable AWS Single Sign-On (SSO)

AWS IoT SiteWise Monitor requires SSO to create a portal and invite users. Create your first user below to enable AWS Single-Sign On. Later in this process, you'll have the opportunity to create other users by using the AWS SSO console. [Learn more](#)

Create a user

Email address
john.doe@example.com

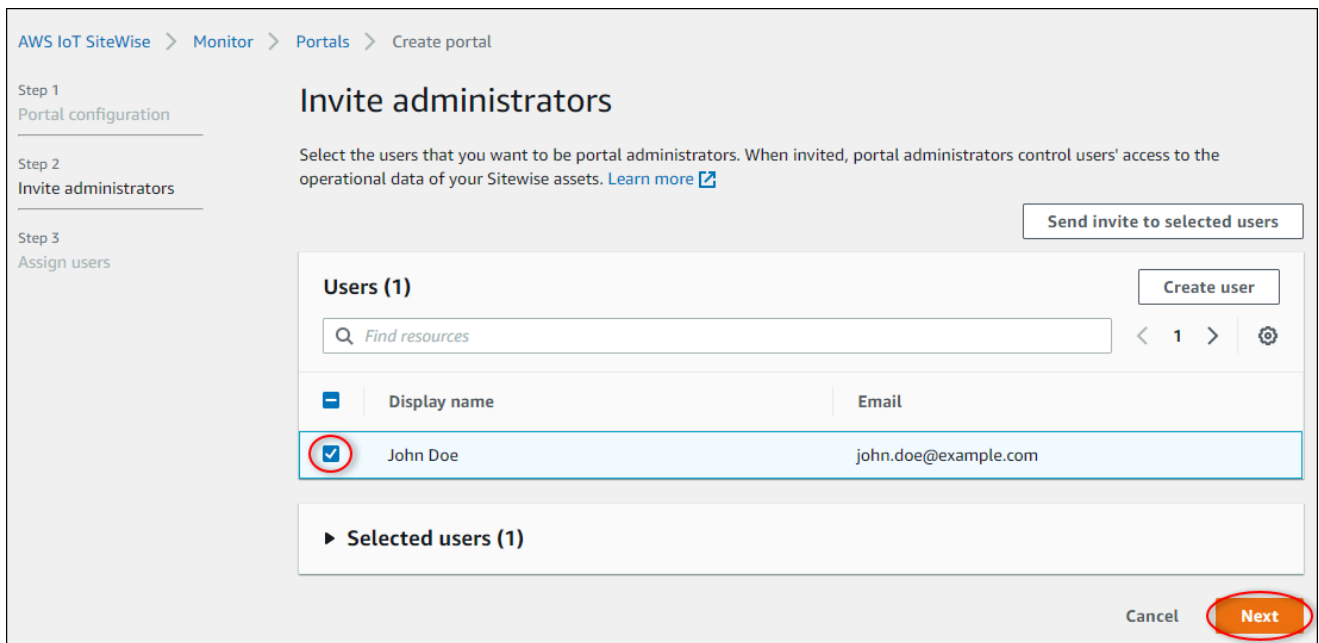
First name
John

Last name
Doe

Upon creation this application will enable AWS Organizations and Single Sign-On. [Learn more](#)

Cancel **Create user**

- b. Pilih Create user (Buat pengguna).
6. Pada halaman konfigurasi Portal, selesaikan langkah-langkah berikut:
- a. Masukkan nama untuk portal Anda, seperti **WindFarmPortal**.
 - b. (Opsional) Masukkan deskripsi untuk portal Anda. Jika Anda memiliki beberapa portal, gunakan deskripsi yang bermakna untuk melacak isi setiap portal.
 - c. (Opsional) Unggah gambar untuk ditampilkan di portal.
 - d. Masukkan alamat email yang dapat dihubungi pengguna portal ketika mereka memiliki masalah dengan portal dan memerlukan bantuan dari AWS administrator perusahaan Anda untuk menyelesaikannya.
 - e. Pilih Buat portal.
7. Pada halaman Undang administrator, Anda dapat menetapkan pengguna Pusat Identitas IAM ke portal sebagai administrator. Administrator portal mengelola izin dan proyek dalam portal. Di halaman ini, lakukan hal berikut:
- a. Pilih pengguna untuk menjadi administrator portal. Jika Anda mengaktifkan IAM Identity Center sebelumnya dalam tutorial ini, pilih pengguna yang Anda buat.



- b. (Optional) Pilih Kirim undangan ke pengguna yang dipilih. Klien email Anda terbuka, dan undangan muncul di badan pesan. Anda dapat menyesuaikan email sebelum mengirimkannya ke administrator portal Anda. Anda juga dapat mengirim email ke administrator portal Anda nanti. Jika Anda mencoba SiteWise Monitor untuk pertama kalinya dan akan menjadi administrator portal, Anda tidak perlu mengirim email sendiri.
 - c. Pilih Berikutnya.
8. Pada halaman Tetapkan pengguna, Anda dapat menetapkan pengguna Pusat Identitas IAM ke portal. Administrator portal nantinya dapat menetapkan pengguna ini sebagai pemilik proyek atau pemirsa. Pemilik proyek dapat membuat dasbor dalam proyek. Pemirsa proyek memiliki akses hanya-baca ke proyek yang ditetapkan. Di halaman ini, Anda dapat membuat pengguna IAM Identity Center untuk ditambahkan ke portal.

Note

Jika Anda tidak masuk ke akun AWS Organizations manajemen, Anda tidak dapat membuat pengguna Pusat Identitas IAM. Pilih Tetapkan pengguna untuk membuat portal tanpa pengguna portal, lalu lewati langkah ini.

Di halaman ini, lakukan hal berikut:

- a. Selesaikan langkah-langkah berikut dua kali untuk membuat dua pengguna IAM Identity Center:

- i. Pilih Buat pengguna untuk membuka kotak dialog tempat Anda memasukkan detail untuk pengguna baru.
- ii. Masukkan alamat Email, Nama depan, dan Nama belakang untuk pengguna baru. IAM Identity Center mengirimkan email kepada pengguna untuk mengatur kata sandi mereka. Jika Anda ingin masuk ke portal sebagai pengguna ini, pilih alamat email yang dapat Anda akses. Setiap alamat email harus unik. Pengguna Anda masuk ke portal menggunakan alamat email mereka sebagai nama pengguna mereka.

Create user ✕

Create a new AWS user. You can assign this user access to AWS applications and services

Email address
mary.major@example.com

First name: Mary Last name: Major

Cancel **Create user**

- iii. Pilih Create user (Buat pengguna).
- b. Pilih dua pengguna IAM Identity Center yang Anda buat pada langkah sebelumnya.

AWS IoT SiteWise > Monitor > Portals > WindFarmPortal > Assign users

Assign users

Users (3) Create user

Find resources

	Display name	Email
<input type="checkbox"/>	John Doe	john.doe@example.com
<input checked="" type="checkbox"/>	Mary Major	mary.major@example.com
<input checked="" type="checkbox"/>	Mateo Jackson	mateo.jackson@example.com

Selected users (2)

Cancel Assign users

- c. Pilih Tetapkan pengguna untuk menambahkan pengguna ini ke portal.

Halaman portal terbuka dengan portal baru Anda terdaftar.

Langkah 2: Masuk ke portal

Dalam prosedur ini, Anda masuk ke portal baru Anda menggunakan AWS IAM Identity Center pengguna yang Anda tambahkan ke portal.

Untuk masuk ke portal

1. Pada halaman Portal, pilih Tautan portal baru Anda untuk membuka portal Anda di tab baru.

AWS IoT SiteWise > Monitor > Portals

Portals (1) Delete View details Create portal


Your employees can use web portals to access your AWS IoT SiteWise asset data. This lets them analyze your operation and draw insights. You configure who has access to each portal.

Filter portals

	Name	Link	Date last modified	Date created	Status
<input type="radio"/>	WindFarmPortal	https://a1b2c3d4-5678-90ab-cdef-1111EXAMPLE.app.iotsitewise.aws	04-28-2020	04-20-2020	Active

2. Jika Anda membuat pengguna IAM Identity Center pertama Anda di awal tutorial, gunakan langkah-langkah berikut untuk membuat kata sandi untuk pengguna Anda:
 - a. Periksa email Anda untuk baris subjek Invitation to join AWS IAM Identity Center.
 - b. Buka email undangan itu dan pilih Accept invitation.
 - c. Di jendela baru, atur kata sandi untuk pengguna Pusat Identitas IAM Anda.

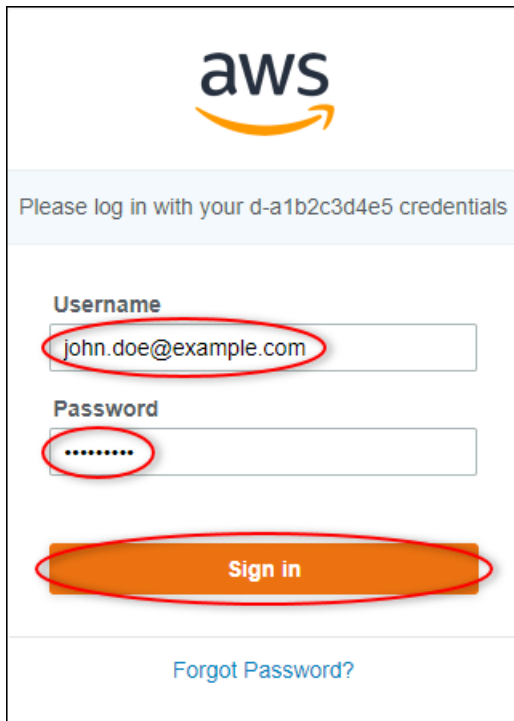
Jika Anda ingin masuk nanti ke portal sebagai pengguna Pusat Identitas IAM kedua dan ketiga yang Anda buat sebelumnya, Anda juga dapat menyelesaikan langkah-langkah ini untuk mengatur kata sandi bagi pengguna tersebut.

 Note

Jika Anda tidak menerima email, Anda dapat membuat kata sandi untuk pengguna Anda di konsol Pusat Identitas IAM. Untuk informasi selengkapnya, lihat [Mengatur ulang kata sandi pengguna](#) di Panduan AWS IAM Identity Center Pengguna.

3. Masukkan Pusat Identitas IAM Anda Username dan Password. Jika Anda membuat pengguna IAM Identity Center Anda sebelumnya dalam tutorial ini, Anda Username adalah alamat email dari pengguna administrator portal yang Anda buat.

Semua pengguna portal, termasuk administrator portal, harus masuk dengan kredensial pengguna Pusat Identitas IAM mereka. Kredensial ini biasanya bukan kredensial yang sama dengan yang Anda gunakan untuk masuk ke AWS Management Console



aws

Please log in with your d-a1b2c3d4e5 credentials

Username
john.doe@example.com

Password
.....

Sign in

[Forgot Password?](#)

4. Pilih Sign in.

Portal Anda terbuka.

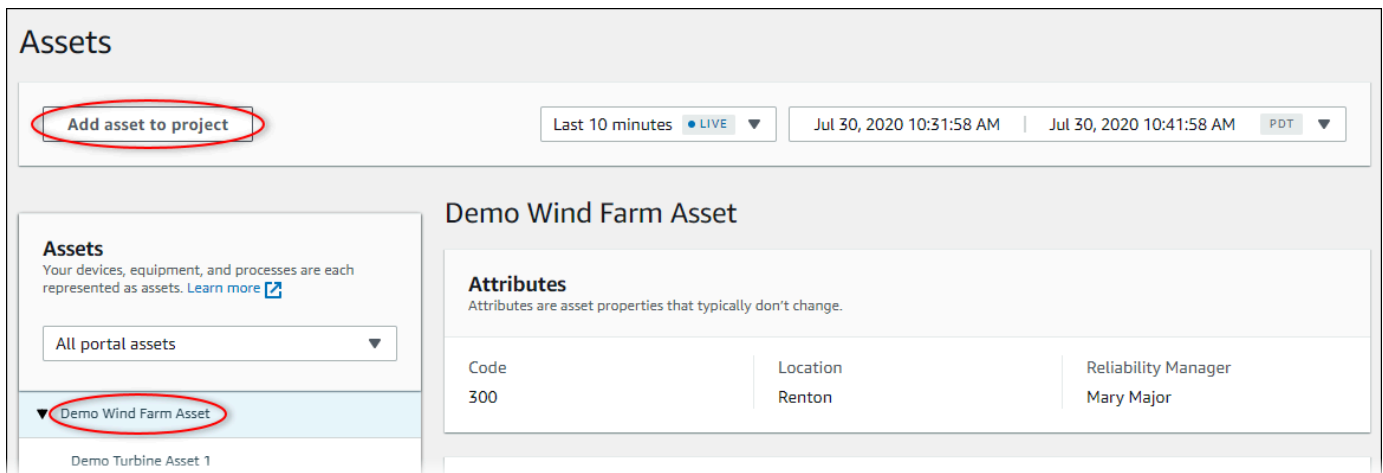
Langkah 3: Buat proyek ladang angin

Dalam prosedur ini, Anda membuat proyek di portal Anda. Proyek adalah sumber daya yang menentukan sekumpulan izin, aset, dan dasbor, yang dapat Anda konfigurasi untuk memvisualisasikan data aset dalam proyek tersebut. Dengan proyek, Anda menentukan siapa yang memiliki akses ke subset operasi Anda dan bagaimana data subset tersebut divisualisasikan. Anda dapat menetapkan pengguna portal sebagai pemilik atau pemirsa setiap proyek. Pemilik proyek dapat membuat dasbor untuk memvisualisasikan data dan berbagi proyek dengan pengguna lain. Pemirsa proyek dapat melihat dasbor tetapi tidak mengeditnya. Untuk informasi selengkapnya tentang peran di SiteWise Monitor, lihat [SiteWise Memantau peran](#).

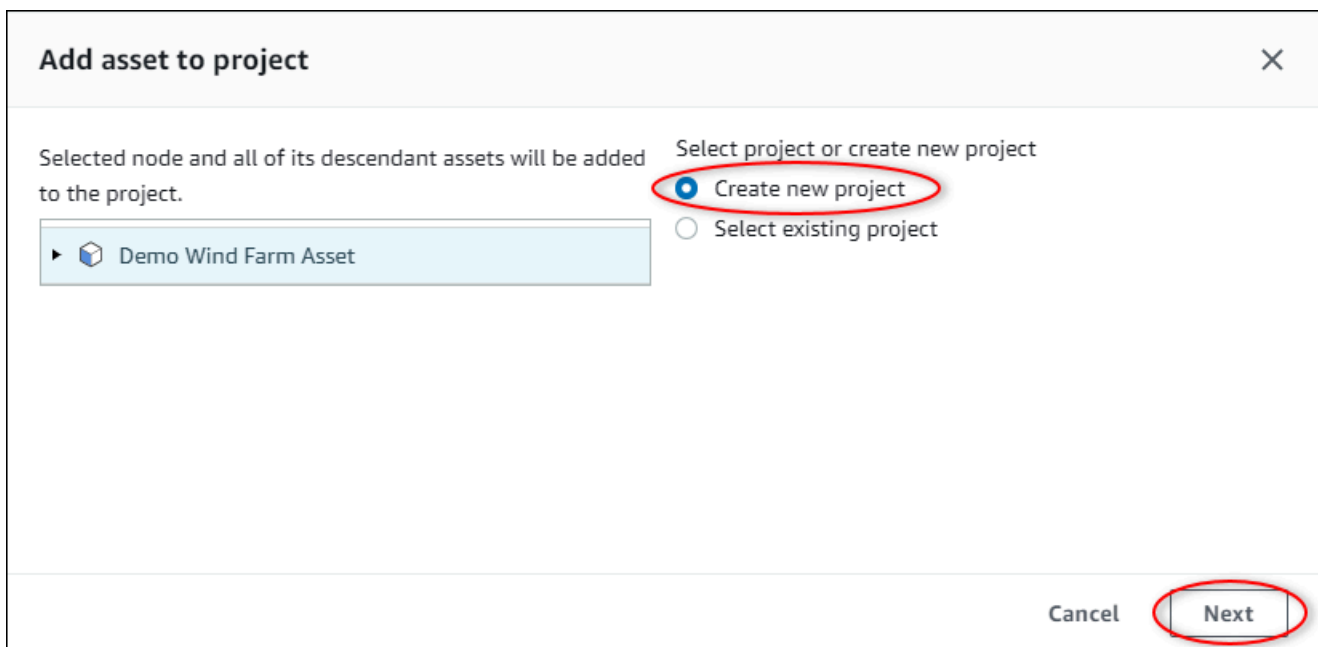
Untuk membuat proyek ladang angin

1. Di panel navigasi kiri di portal Anda, pilih tab Aset. Di halaman Aset, Anda dapat menjelajahi semua aset yang tersedia di portal dan menambahkan aset ke proyek.

- Di browser aset, pilih Demo Wind Farm Asset. Saat memilih aset, Anda dapat menjelajahi data langsung dan historis aset tersebut. Anda juga dapat menekan Shift untuk memilih beberapa aset dan membandingkan datanya side-by-side.
- Pilih Tambahkan aset ke proyek di kiri atas. Proyek berisi dashboard yang dapat dilihat pengguna portal Anda untuk menjelajahi data Anda. Setiap proyek memiliki akses ke subset aset Anda di AWS IoT SiteWise. Saat Anda menambahkan aset ke proyek, semua pengguna yang memiliki akses ke proyek tersebut juga dapat mengakses data untuk aset tersebut dan anak-anaknya.



- Dalam kotak dialog Tambahkan aset ke proyek, pilih Buat proyek baru, lalu pilih Berikutnya.



- Di kotak dialog Buat proyek baru, masukkan nama Proyek dan deskripsi Proyek untuk proyek Anda, lalu pilih Tambahkan aset ke proyek.



Create new project ✕

Project name
Wind Farm 1
The project name can have up to 256 characters.

Project description
A project that contains dashboards for wind farm #1.
The project description can have up to 2048 characters.

Cancel Previous **Add asset to project**

Halaman proyek baru Anda terbuka.

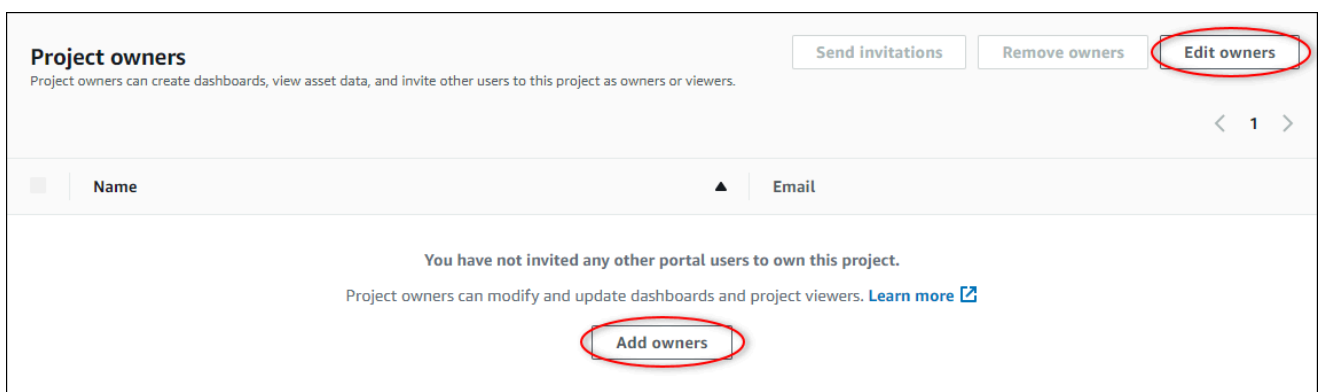
6. Pada halaman proyek, Anda dapat menambahkan pengguna portal sebagai pemilik atau pemirsa proyek ini.

Note

Jika Anda tidak masuk ke akun AWS Organizations manajemen, Anda mungkin tidak memiliki pengguna portal untuk menetapkan proyek ini, sehingga Anda dapat melewati langkah ini.

Di halaman ini, lakukan hal berikut:

- a. Di bawah Pemilik proyek, pilih Tambah pemilik atau Edit pengguna.



Project owners Send invitations Remove owners **Edit owners**

Project owners can create dashboards, view asset data, and invite other users to this project as owners or viewers.

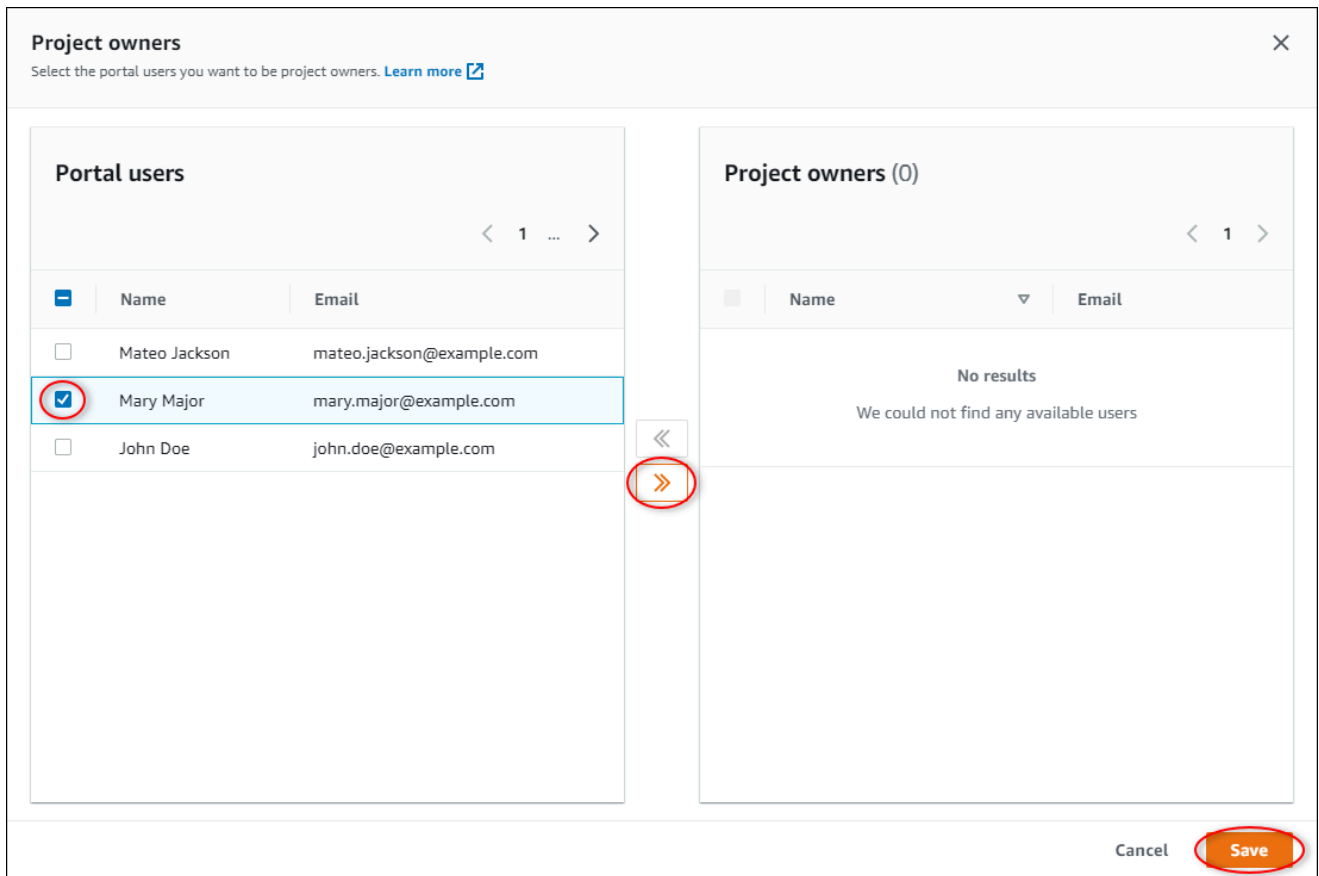
< 1 >

Name	Email
------	-------

You have not invited any other portal users to own this project.
Project owners can modify and update dashboards and project viewers. [Learn more](#)

Add owners

- b. Pilih pengguna untuk ditambahkan sebagai pemilik proyek (misalnya, Mary Major), dan kemudian pilih ikon >>.



- c. Pilih Simpan.

Pengguna IAM Identity Center Anda Mary Majordapat masuk ke portal ini untuk mengedit dasbor dalam proyek ini dan berbagi proyek ini dengan pengguna lain di portal ini.

- d. Di bawah Penampil proyek, pilih Tambahkan pemirsa atau Edit pengguna.
- e. Pilih pengguna untuk ditambahkan sebagai penampil proyek (misalnya, Mateo Jackson), dan kemudian pilih ikon >>.
- f. Pilih Simpan.

Pengguna Pusat Identitas IAM Anda Mateo Jackson dapat masuk ke portal ini untuk melihat, tetapi tidak mengedit, dasbor di proyek ladang angin.

Langkah 4: Buat dasbor untuk memvisualisasikan data ladang angin

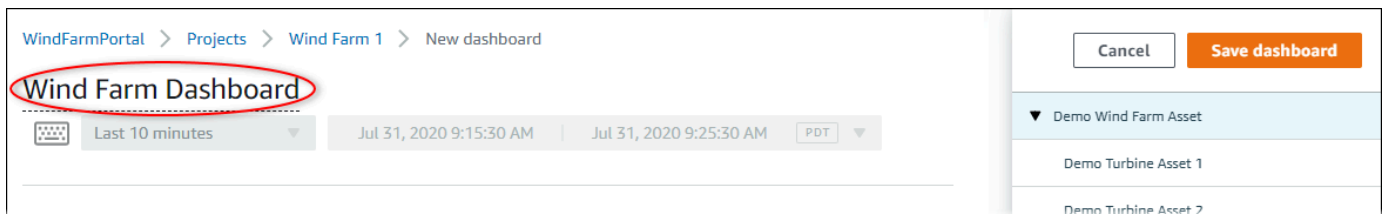
Dalam prosedur ini, Anda membuat dasbor untuk memvisualisasikan data ladang angin demo. Dasbor berisi visualisasi yang dapat disesuaikan dari data aset proyek Anda. Setiap visualisasi dapat memiliki jenis yang berbeda, seperti diagram garis, diagram batang, atau tampilan indikator kinerja kunci (KPI). Anda dapat memilih jenis visualisasi yang paling sesuai untuk data Anda. Pemilik proyek dapat mengedit dasbor, sedangkan pemirsa proyek hanya dapat melihat dasbor untuk mendapatkan wawasan.

Untuk membuat dasbor dengan visualisasi

1. Pada halaman proyek baru Anda, pilih Buat dasbor untuk membuat dasbor dan buka halaman editnya.

Di halaman edit dasbor, Anda dapat menyeret properti aset dari hierarki aset ke dasbor untuk membuat visualisasi. Kemudian, Anda dapat mengedit setiap judul visualisasi, judul legenda, jenis, ukuran, dan lokasi di dasbor.

2. Masukkan nama dasbor Anda.



3. Seret Total Average Power dari dasbor Demo Wind Farm Asset ke dasbor untuk membuat visualisasi.

The screenshot displays the 'Wind Farm Dashboard' interface. At the top, the breadcrumb navigation shows 'WindFarmPortal > Projects > Wind Farm 1 > New dashboard'. The dashboard title is 'Wind Farm Dashboard'. Below the title, there are filters for 'Last 10 minutes', a date range from 'Jul 31, 2020 9:15:30 AM' to 'Jul 31, 2020 9:25:30 AM', and a time zone dropdown set to 'PDT'. The main area contains a grid of widgets. One widget, 'Total Average Power', is highlighted with a red oval and shows a value of '24038 Watts'. To the right, a sidebar panel titled 'Properties for "Demo Wind Farm Asset"' is visible. It lists 'Code' as '300' and 'Total Overdrive State Time' as '0 seconds'. A red oval highlights an empty field in the properties panel.

4. Pilih Demo Turbine Asset 1 untuk menampilkan properti untuk aset itu, lalu seret Wind Speed ke dasbor untuk membuat visualisasi kecepatan angin.

WindFarmPortal > Projects > Wind Farm 1 > New dashboard

Wind Farm Dashboard

Last 10 minutes | Jul 31, 2020 9:15:30 AM | Jul 31, 2020 9:25:30 AM | PDT

Total Average Po...

26,000
25,500
25,000
24,500
24,000
23,500
23,000
22,500
22,000

09:20 09:25

— Total Average Power (Demo Wind Farm Asset)
23420 Watts

Wind Speed 14.753 m/s

Cancel Save dashboard

▼ Demo Wind Farm Asset

- Demo Turbine Asset 1
- Demo Turbine Asset 2
- Demo Turbine Asset 3
- Demo Turbine Asset 4

Properties for "Demo Turbine Asset 1"

Overdrive State 0

Overdrive State Time 0 Seconds

RotationsPerMinute 27.143 RPM

RotationsPerSecond 4.524e-1 RPS

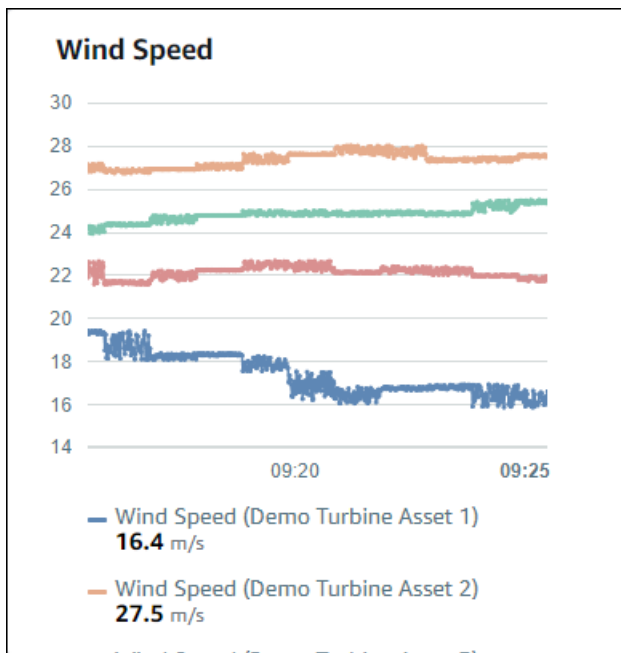
Torque (KiloNewton Meter) 2.5261 kNm

Torque (Newton Meter) 2526.1 Nm

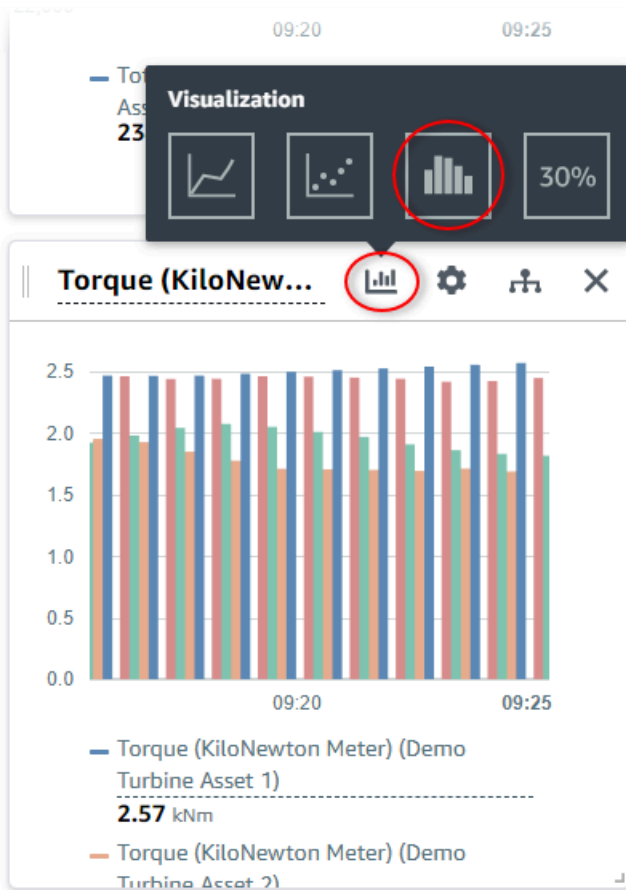
Wind Direction 7.4587 Degrees

5. Tambahkan Wind Speed ke visualisasi kecepatan angin baru untuk masing-masing Demo Turbine Asset 2, 3, dan 4 (dalam urutan itu).

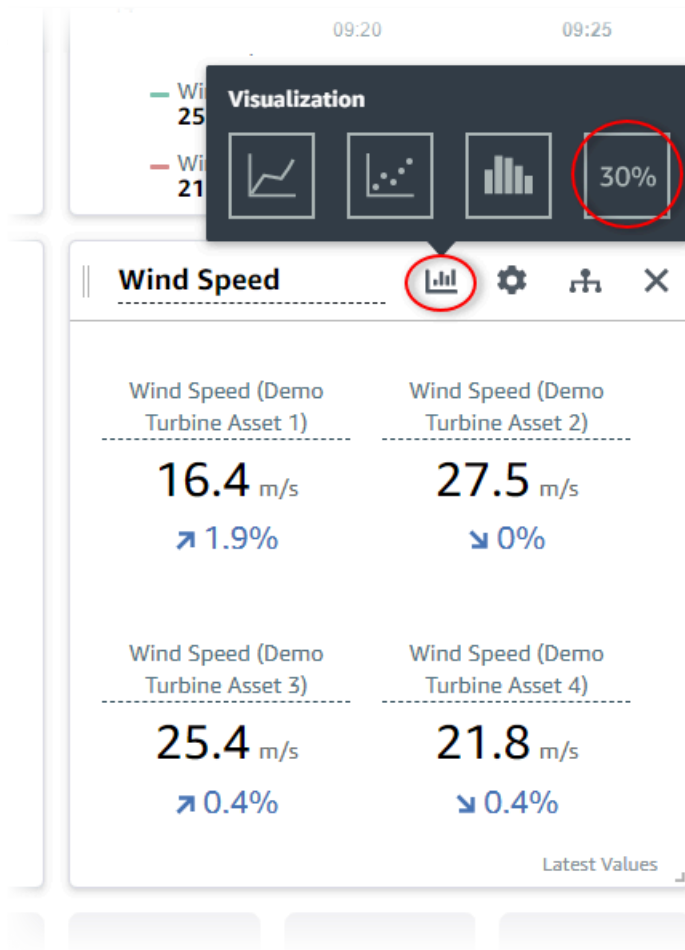
Wind Speed Visualisasi Anda akan terlihat mirip dengan tangkapan layar berikut.



6. Ulangi langkah 4 dan 5 untuk Torque (KiloNewton Meter) properti turbin angin untuk membuat visualisasi torsi turbin angin.
7. Pilih ikon jenis visualisasi untuk Torque (KiloNewton Meter) visualisasi, lalu pilih ikon bagan batang.



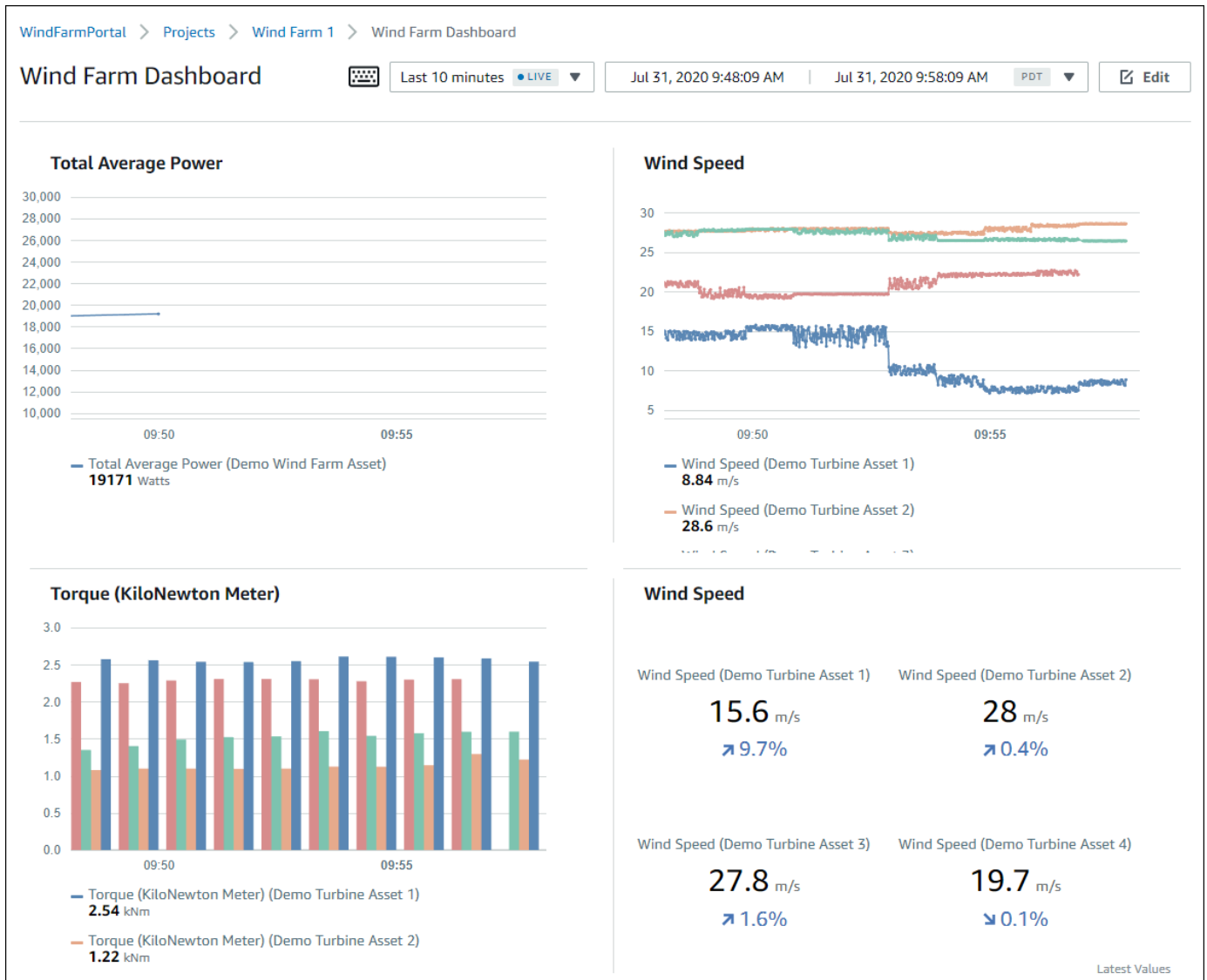
8. Ulangi langkah 4 dan 5 untuk Wind Direction properti turbin angin untuk membuat visualisasi arah angin.
9. Pilih ikon jenis visualisasi untuk Wind Direction visualisasi, lalu pilih ikon bagan KPI (). 30%



10. (Opsional) Buat perubahan lain pada setiap judul visualisasi, judul legenda, jenis, ukuran, dan lokasi sesuai kebutuhan.

11. Pilih Simpan dasbor di kanan atas untuk menyimpan dasbor Anda.

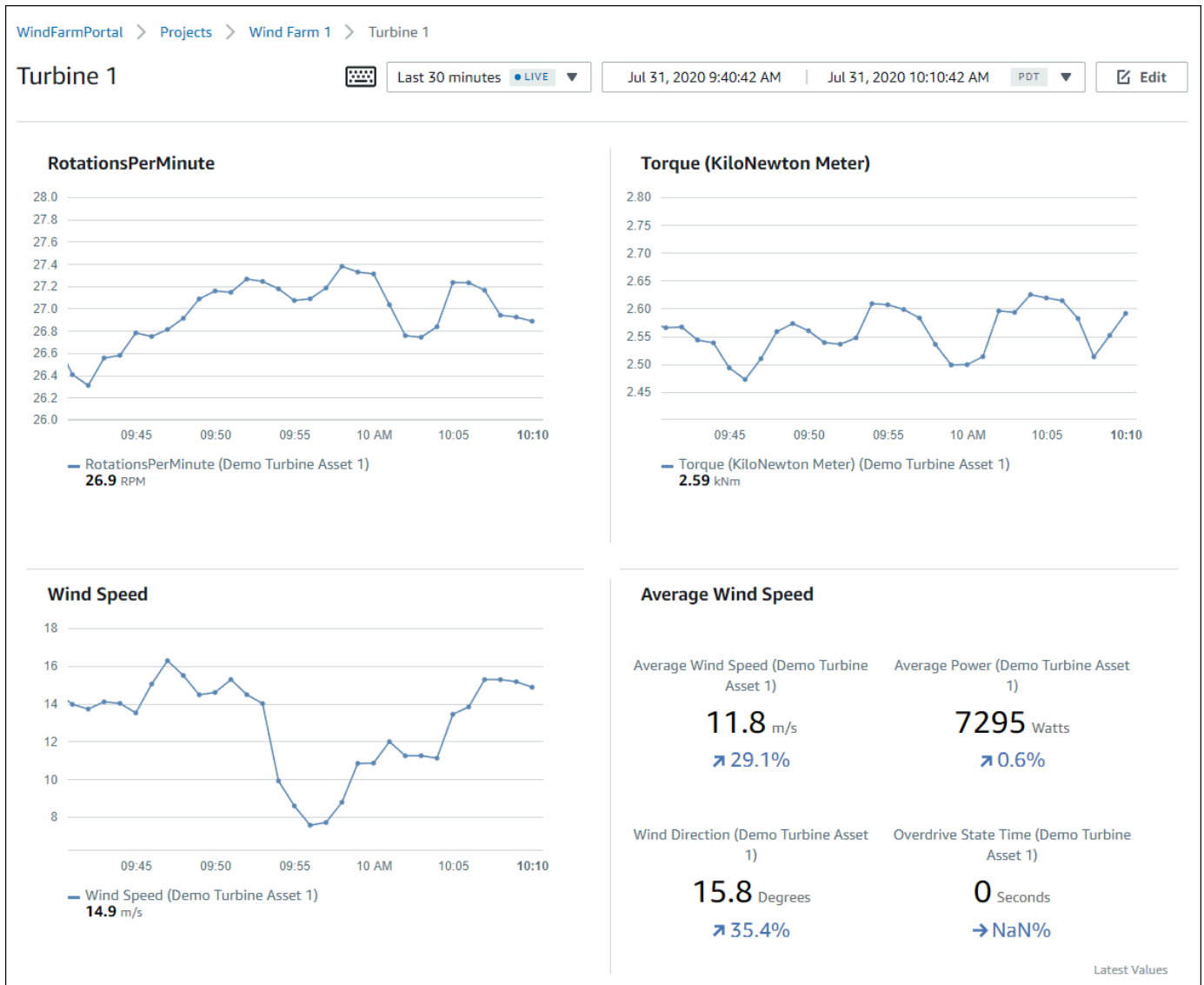
Dasbor Anda akan terlihat mirip dengan tangkapan layar berikut.



12. (Opsional) Buat dasbor tambahan untuk setiap aset turbin angin.

Sebagai praktik terbaik, kami menyarankan Anda membuat dasbor untuk setiap aset sehingga pemirsa proyek Anda dapat menyelidiki masalah apa pun dengan setiap aset individu. Anda hanya dapat menambahkan hingga 5 aset ke setiap visualisasi, jadi Anda harus membuat beberapa dasbor untuk aset hierarkis Anda dalam banyak skenario.

Dasbor untuk turbin angin demo mungkin terlihat mirip dengan tangkapan layar berikut.



- (Opsional) Ubah timeline atau pilih titik data pada visualisasi untuk menjelajahi data di dasbor Anda. Untuk informasi selengkapnya, lihat [Melihat dasbor](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.

Langkah 5: Jelajahi portal

Dalam prosedur ini, Anda dapat menjelajahi portal sebagai pengguna dengan izin lebih sedikit daripada administrator AWS IoT SiteWise portal.

Untuk menjelajahi portal dan menyelesaikan tutorial

- (Opsional) Jika Anda menambahkan pengguna lain ke proyek sebagai pemilik atau pemirsa, Anda dapat masuk ke portal sebagai pengguna ini. Ini memungkinkan Anda menjelajahi portal sebagai pengguna dengan izin lebih sedikit daripada administrator portal.

⚠ Important

Anda dikenakan biaya untuk setiap pengguna yang masuk ke portal. Untuk informasi selengkapnya, silakan lihat [Harga AWS IoT SiteWise](#).

Untuk menjelajahi portal sebagai pengguna lain, lakukan hal berikut:

- a. Pilih Keluar di kiri bawah portal untuk keluar dari aplikasi web.
- b. Pilih Keluar di kanan atas portal aplikasi Pusat Identitas IAM untuk keluar dari pengguna Pusat Identitas IAM Anda.
- c. Masuk ke portal sebagai pengguna Pusat Identitas IAM yang Anda tetapkan sebagai pemilik proyek atau penampil proyek. Untuk informasi selengkapnya, lihat [Langkah 2: Masuk ke portal](#).

Anda telah menyelesaikan tutorialnya. Ketika Anda selesai menjelajahi ladang angin demo Anda di SiteWise Monitor, ikuti prosedur selanjutnya untuk membersihkan sumber daya Anda.

Langkah 6: Bersihkan sumber daya setelah tutorial

Setelah Anda menyelesaikan tutorial, Anda dapat membersihkan sumber daya Anda. Anda tidak dikenakan biaya AWS IoT SiteWise jika pengguna tidak masuk ke portal Anda, tetapi Anda dapat menghapus portal dan Direktori Pusat Identitas AWS IAM pengguna Anda. Aset ladang angin demo Anda dihapus pada akhir durasi yang Anda pilih saat Anda membuat demo, atau Anda dapat menghapus demo secara manual. Untuk informasi selengkapnya, lihat [Menghapus demo AWS IoT SiteWise](#).

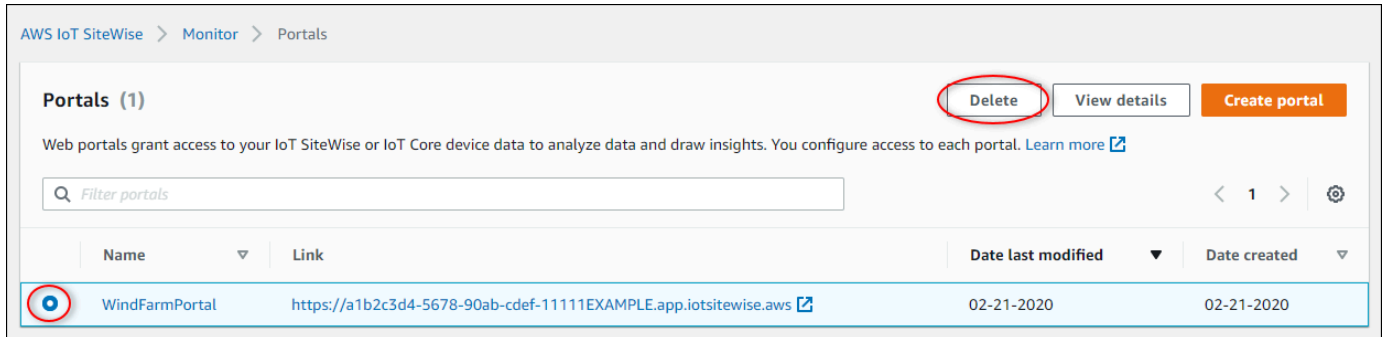
Gunakan prosedur berikut untuk menghapus portal Anda dan pengguna Pusat Identitas IAM.

Untuk menghapus portal

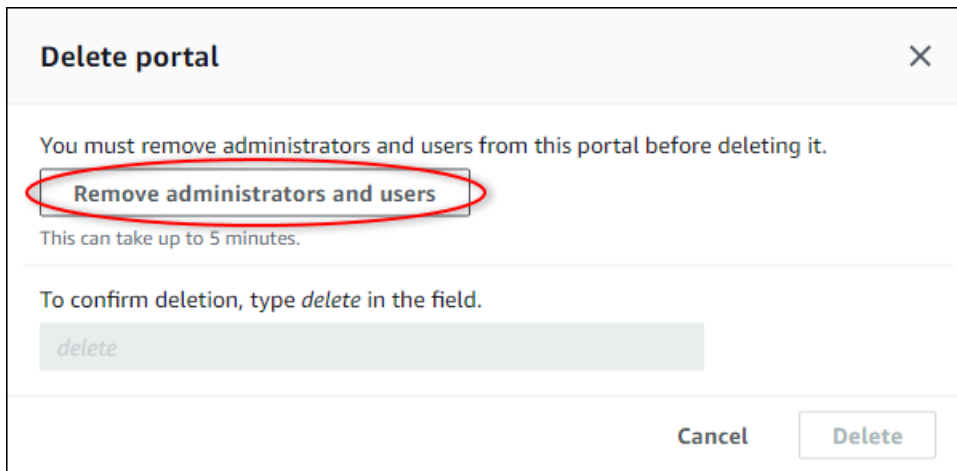
1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.

2. Di panel navigasi kiri, pilih Portal.
3. Pilih portal Anda WindFarmPortal, lalu pilih Hapus.

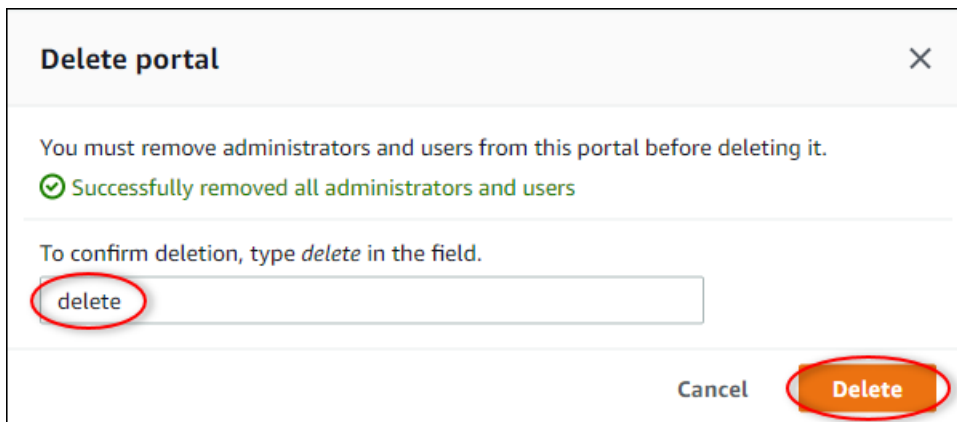
Saat Anda menghapus portal atau proyek, aset yang terkait dengan proyek yang dihapus tidak akan terpengaruh.



4. Di kotak dialog Hapus portal, pilih Hapus administrator dan pengguna.

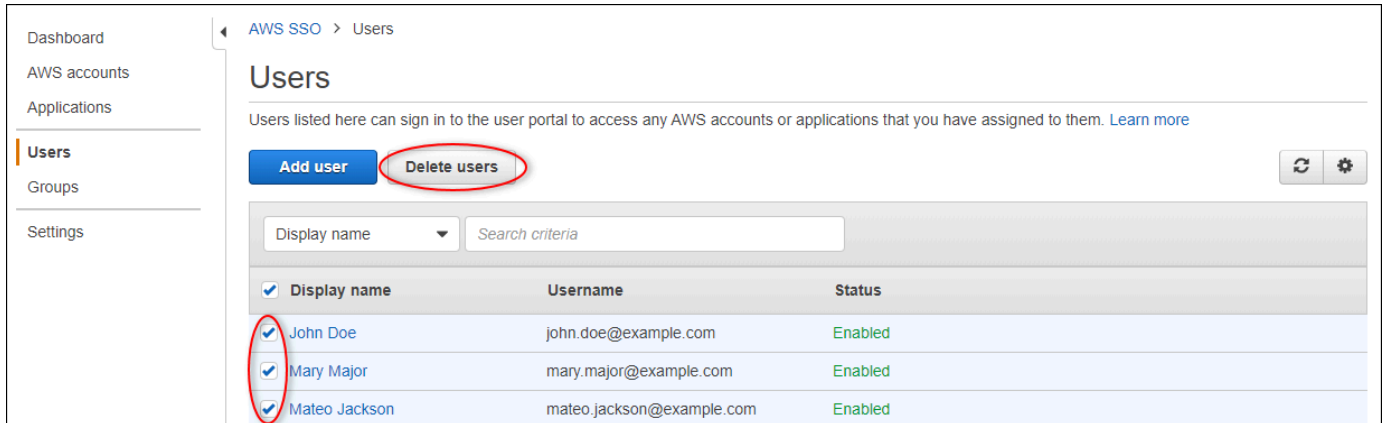


5. Masukkan **delete** untuk mengonfirmasi penghapusan, lalu pilih Hapus.

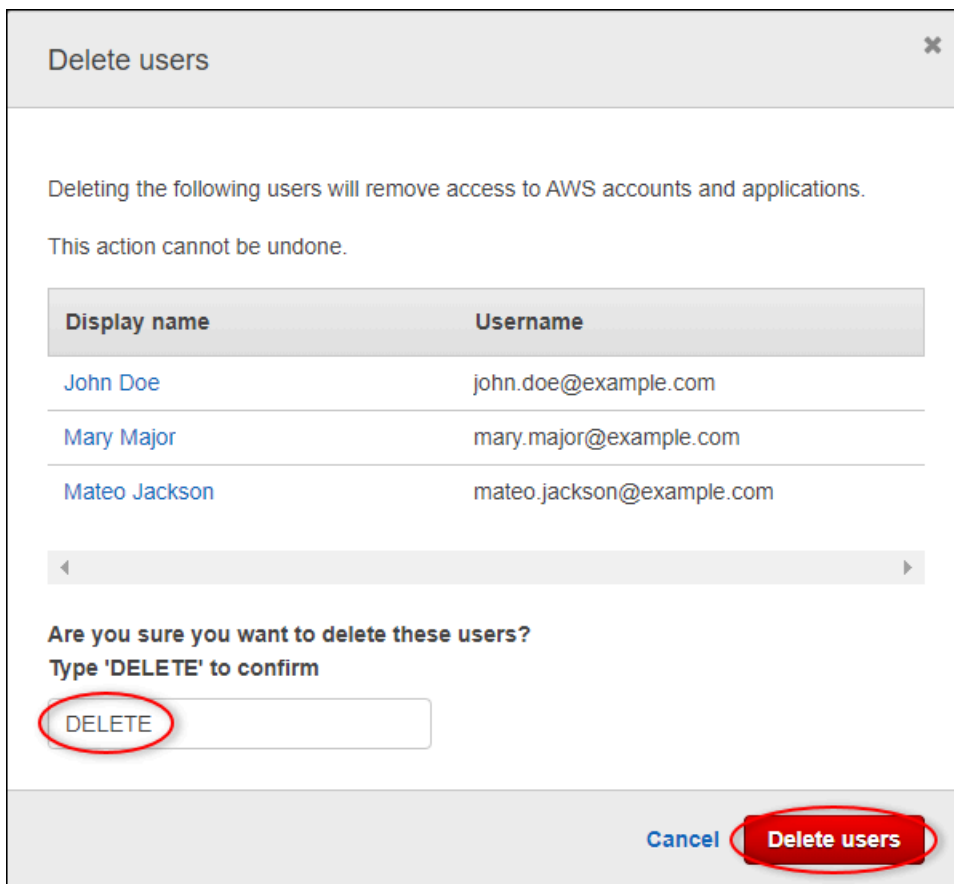


Untuk menghapus pengguna IAM Identity Center

1. Arahkan ke [konsol Pusat Identitas IAM](#).
2. Pada panel navigasi kiri, pilih Pengguna.
3. Pilih kotak centang untuk menghapus setiap pengguna, lalu pilih Hapus pengguna.



4. Di kotak dialog Hapus pengguna, masukkan **DELETE**, lalu pilih Hapus pengguna.



Menerbitkan pembaruan nilai properti ke Amazon DynamoDB

Tutorial ini memperkenalkan cara mudah untuk menyimpan data Anda dengan menggunakan [Amazon DynamoDB](#), sehingga lebih mudah untuk mengakses data aset historis tanpa berulang kali menanyakan API. AWS IoT SiteWise Setelah Anda menyelesaikan tutorial ini, Anda dapat membuat perangkat lunak khusus yang mengkonsumsi data aset Anda, seperti peta langsung kecepatan angin dan arah di seluruh ladang angin. Jika Anda ingin memantau dan memvisualisasikan data Anda tanpa menerapkan solusi perangkat lunak khusus, lihat [Memantau data dengan AWS IoT SiteWise Monitor](#).

Dalam tutorial ini, Anda membangun AWS IoT SiteWise demo yang menyediakan kumpulan sampel data untuk ladang angin. Anda mengonfigurasi pembaruan nilai properti dari demo ladang angin untuk mengirim data, melalui aturan AWS IoT Core, ke tabel DynamoDB yang Anda buat. Saat Anda mengaktifkan pembaruan nilai properti, AWS IoT SiteWise mengirimkan data Anda ke AWS IoT Core dalam pesan MQTT. Kemudian, tentukan aturan AWS IoT Core yang melakukan tindakan, seperti tindakan DynamoDB, tergantung pada isi pesan tersebut. Untuk informasi selengkapnya, lihat [Berinteraksi dengan layanan lain AWS](#).

Topik

- [Prasyarat](#)
- [Langkah 1: Konfigurasi AWS IoT SiteWise untuk mempublikasikan pembaruan nilai properti](#)
- [Langkah 2: Buat aturan di AWS IoT Core](#)
- [Langkah 3: Buat tabel DynamoDB](#)
- [Langkah 4: Konfigurasi tindakan aturan DynamoDB](#)
- [Langkah 5: Jelajahi data di DynamoDB](#)
- [Langkah 6: Bersihkan sumber daya setelah tutorial](#)

Prasyarat

Untuk menyelesaikan tutorial ini, Anda memerlukan hal berikut:

- AWS Akun. Jika Anda tidak memilikinya, lihat [Menyiapkan sebuah Akun AWS](#).
- Komputer pengembangan yang menjalankan Windows, macOS, Linux, atau Unix untuk mengakses file. AWS Management Console Untuk informasi lebih lanjut, lihat [Memulai dengan AWS Management Console](#).

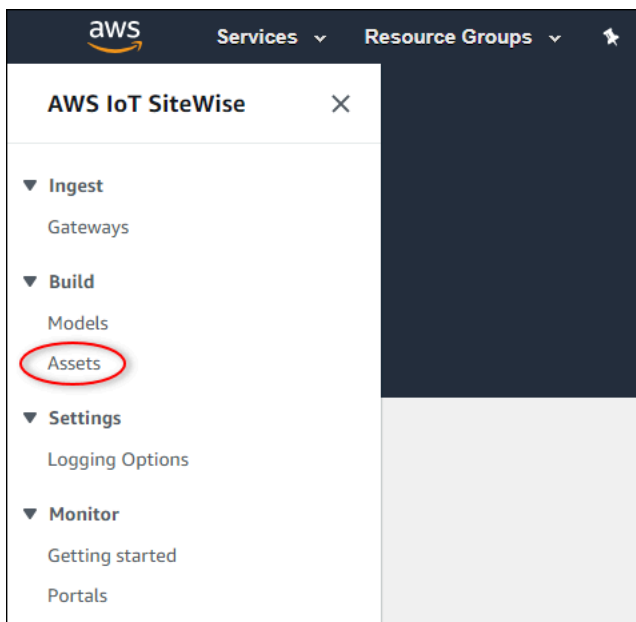
- Pengguna IAM dengan izin administrator.
- Demo ladang AWS IoT SiteWise angin yang sedang berjalan. Saat Anda mengatur demo, itu mendefinisikan model dan aset AWS IoT SiteWise dan mengalirkan data ke mereka untuk mewakili ladang angin. Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT SiteWise demo](#).

Langkah 1: Konfigurasikan AWS IoT SiteWise untuk mempublikasikan pembaruan nilai properti

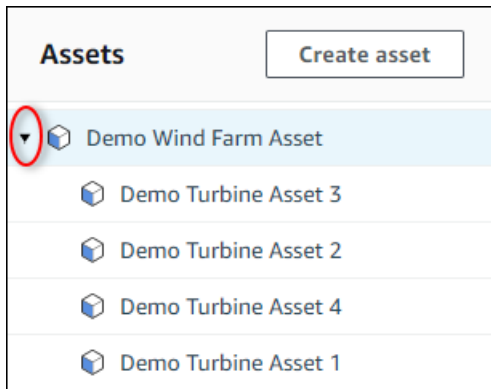
Dalam prosedur ini, Anda mengaktifkan pemberitahuan nilai properti pada properti aset Wind Speedturbin demo Anda. Setelah Anda mengaktifkan pemberitahuan nilai properti, AWS IoT SiteWise menerbitkan setiap pembaruan nilai dalam pesan MQTT ke Core. AWS IoT

Untuk mengaktifkan pemberitahuan pembaruan nilai properti pada properti aset

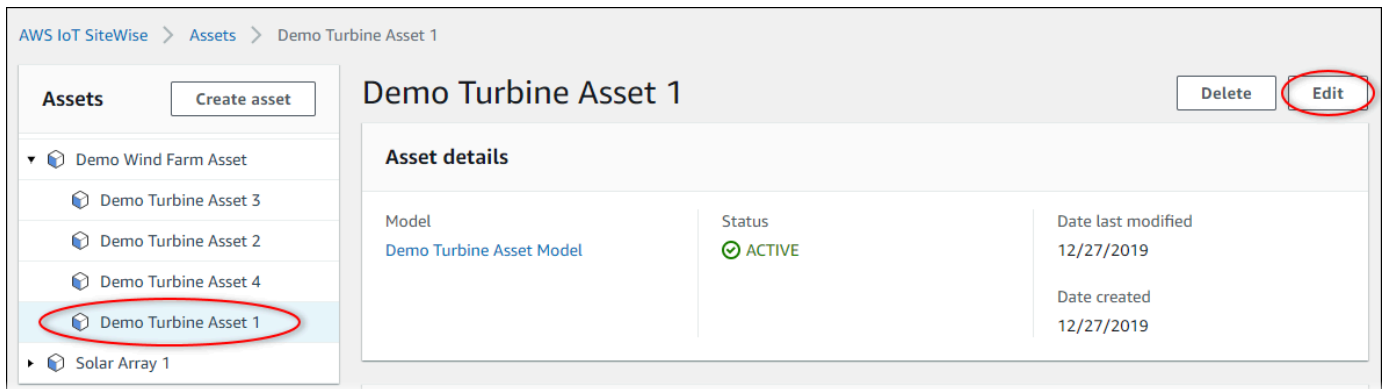
1. Masuk ke [konsol AWS IoT SiteWise](#) tersebut.
2. Tinjau [AWS IoT SiteWise titik akhir dan kuota](#) di mana AWS IoT SiteWise didukung dan alihkan AWS Wilayah, jika perlu. Beralih ke Wilayah tempat Anda menjalankan AWS IoT SiteWise demo.
3. Di panel navigasi kiri, pilih Aset.



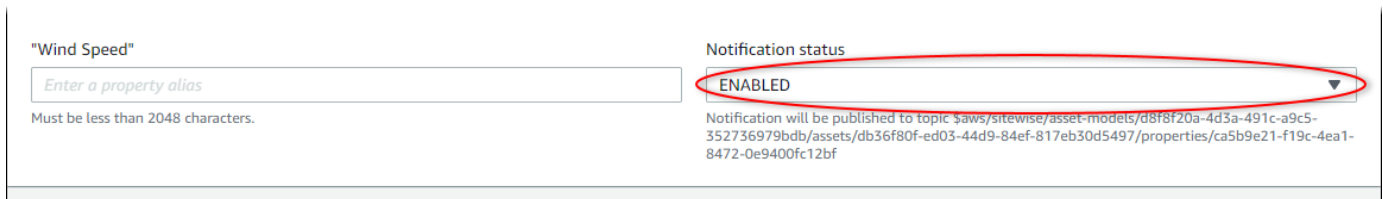
4. Pilih panah di sebelah Demo Wind Farm Asset untuk memperluas hierarki aset ladang angin.



5. Pilih turbin demo dan pilih Edit.



6. Perbarui status Pemberitahuan Wind Speedproperti ke ENABLED.



7. Pilih Simpan aset di bagian bawah halaman.

8. Ulangi langkah 5 hingga 7 untuk setiap aset turbin demo.

9. Pilih turbin demo (misalnya, Demo Turbine Asset 1).

10. Pilih Pengukuran.

11. Pilih ikon salin di sebelah Wind Speedproperti untuk menyalin topik notifikasi ke clipboard Anda. Simpan topik notifikasi untuk digunakan nanti dalam tutorial ini. Anda hanya perlu merekam topik notifikasi dari satu turbin.

Torque (KiloNewton Meter)	-	⊖ Disabled	-	2.128123
Wind Speed	-	✔ Enabled	<code>\$aws/sitewise/asset-models/d8f8f...</code>	26.49812

Topik notifikasi akan terlihat seperti contoh berikut.

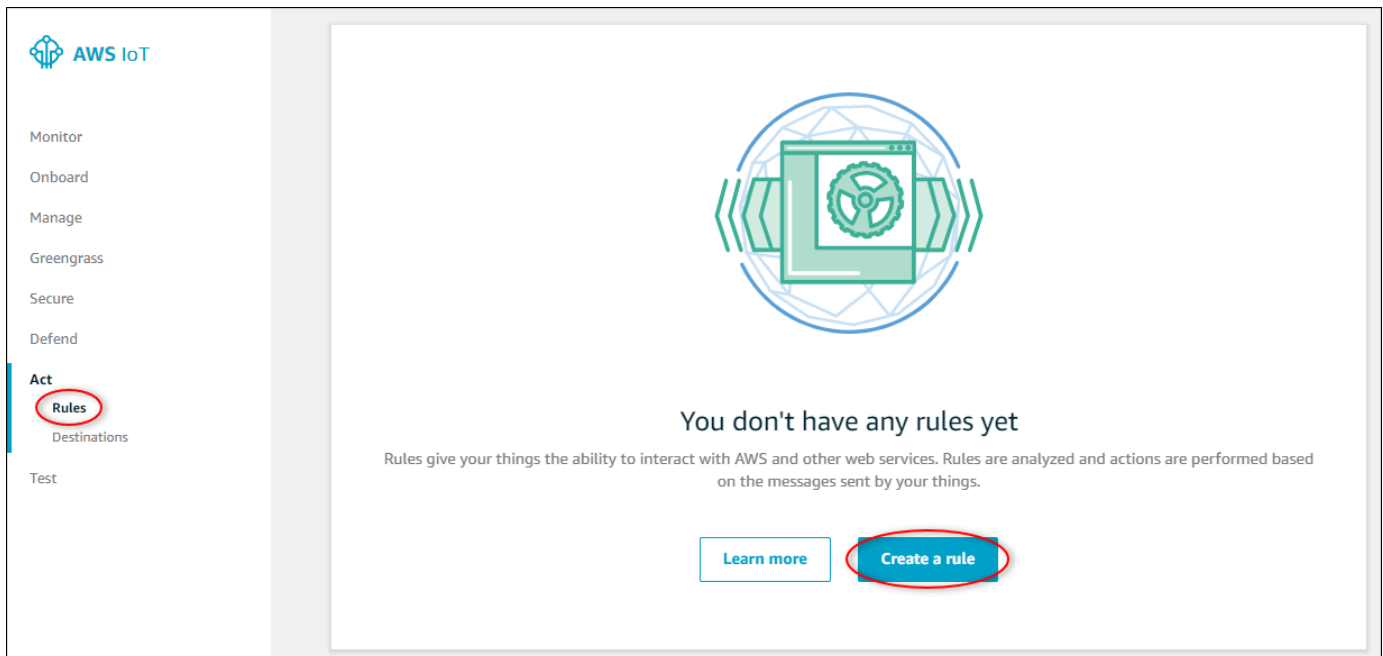
```
$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/  
assets/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE/properties/a1b2c3d4-5678-90ab-  
cdef-33333EXAMPLE
```

Langkah 2: Buat aturan di AWS IoT Core

Dalam prosedur ini, Anda membuat aturan di AWS IoT Core yang mem-parsing pesan notifikasi nilai properti dan menyisipkan data ke dalam tabel Amazon DynamoDB. AWS IoT Aturan inti mengurai pesan MQTT dan melakukan tindakan berdasarkan konten dan topik setiap pesan. Kemudian, Anda membuat aturan dengan tindakan DynamoDB untuk menyisipkan data ke tabel DynamoDB yang Anda buat sebagai bagian dari tutorial ini.

Untuk membuat aturan dengan tindakan DynamoDB

1. Navigasikan ke [konsol AWS IoT](#) tersebut. Jika tombol Mulai muncul, pilihlah.
2. Di panel navigasi kiri, pilih Bertindak lalu pilih Aturan.



3. Jika kotak dialog Anda belum memiliki aturan apa pun muncul, pilih Buat aturan. Jika tidak, pilih Buat.
4. Masukkan nama dan deskripsi aturan.

Create a rule

Create a rule to evaluate messages sent by your things and specify what to do when a message is received (for example, write data to a DynamoDB table or invoke a Lambda function).

Name

Description

5. Temukan topik notifikasi yang Anda simpan sebelumnya dalam tutorial ini.

```
$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/
assets/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE/properties/a1b2c3d4-5678-90ab-
cdef-33333EXAMPLE
```

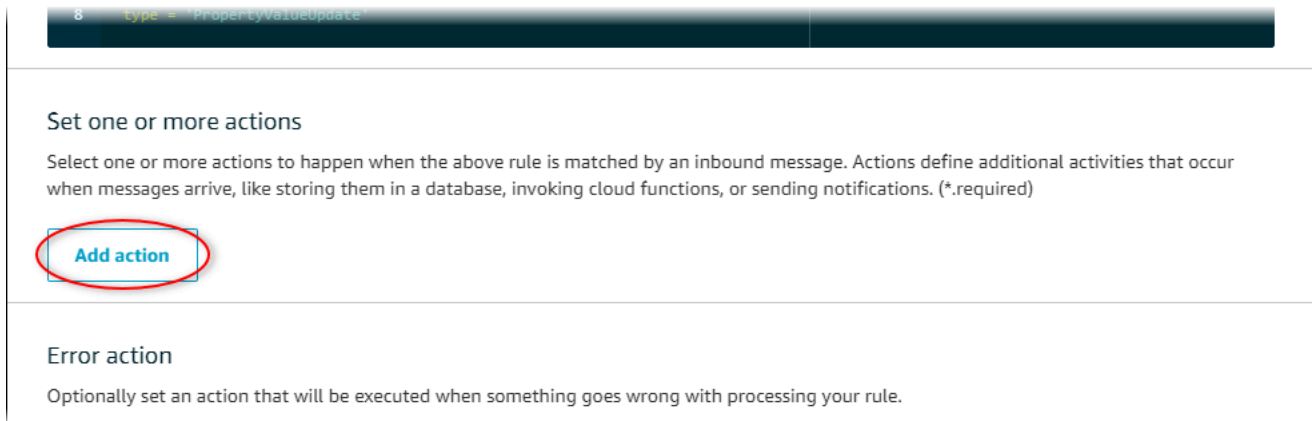
Ganti ID aset (ID setelah `assets/`) dalam topik dengan `a+`. Ini memilih properti kecepatan angin untuk semua aset turbin angin demo. Filter `+` topik menerima semua node dari satu tingkat dalam suatu topik. Topik Anda akan terlihat seperti contoh berikut.

```
$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/assets/+/
properties/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

6. Masukkan pernyataan query aturan berikut. Ganti topik di FROM bagian dengan topik notifikasi Anda.

```
SELECT
  payload.assetId AS asset,
  (SELECT VALUE (value.doubleValue) FROM payload.values) AS windspeed,
  timestamp() AS timestamp
FROM
  '$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/assets/+/'
  properties/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE'
WHERE
  type = 'PropertyValueUpdate'
```


7. Di bawah Setel satu atau beberapa tindakan, pilih Tambah tindakan.



8 type = "PropertyValueUpdate"

Set one or more actions

Select one or more actions to happen when the above rule is matched by an inbound message. Actions define additional activities that occur when messages arrive, like storing them in a database, invoking cloud functions, or sending notifications. (*.required)

Add action

Error action

Optionally set an action that will be executed when something goes wrong with processing your rule.

8. Pada halaman Pilih tindakan, pilih Pisahkan pesan menjadi beberapa kolom tabel DynamoDB (DynamoDBv2).



Select an action

Select an action.

-  Insert a message into a DynamoDB table
DYNAMODB
-  Split message into multiple columns of a DynamoDB table (DynamoDBv2)
DYNAMODBV2
-  Send a message to a Lambda function
LAMBDA

9. Pilih Konfigurasi tindakan di bagian bawah halaman.

10. Pada halaman Configure action, pilih Create a new resource.

Konsol DynamoDB terbuka di tab baru. Biarkan tab tindakan aturan tetap terbuka saat Anda menyelesaikan prosedur berikut.

Langkah 3: Buat tabel DynamoDB

Dalam prosedur ini, Anda membuat tabel Amazon DynamoDB untuk menerima data kecepatan angin dari tindakan aturan.

Untuk membuat tabel DynamoDB

1. Di dasbor konsol DynamoDB, pilih Buat tabel.
2. Masukkan nama untuk tabel Anda.

Create DynamoDB table Tutorial ?

DynamoDB is a schema-less database that only requires a table name and primary key. The table's primary key is made up of one or two attributes that uniquely identify items, partition the data, and sort data within each partition.

Table name* ⓘ

Primary key* Partition key

ⓘ

Add sort key

ⓘ

Table settings

Default settings provide the fastest way to get started with your table. You can modify these default settings now or after your table has been created.

Use default settings

- No secondary indexes.
- Provisioned capacity set to 5 reads and 5 writes.
- Basic alarms with 80% upper threshold using SNS topic "dynamodb".
- Encryption at Rest with DEFAULT encryption type.

ⓘ You do not have the required role to enable Auto Scaling by default. Please refer to [documentation](#).

+ Add tags **NEW!**

Additional charges may apply if you exceed the AWS Free Tier levels for CloudWatch or Simple Notification Service. Advanced alarm settings are available in the CloudWatch management console.

Cancel **Create**

3. Untuk kunci Primer, lakukan hal berikut:
 - a. Masukkan **timestamp** sebagai kunci partisi.
 - b. Pilih jenis Angka.
 - c. Pilih kotak centang Add sort key.
 - d. Masukkan **asset** sebagai kunci sortir, dan biarkan tipe kunci sortir default String.
4. Pilih Buat.

Ketika Tabel sedang dibuat pemberitahuan menghilang, tabel Anda siap.

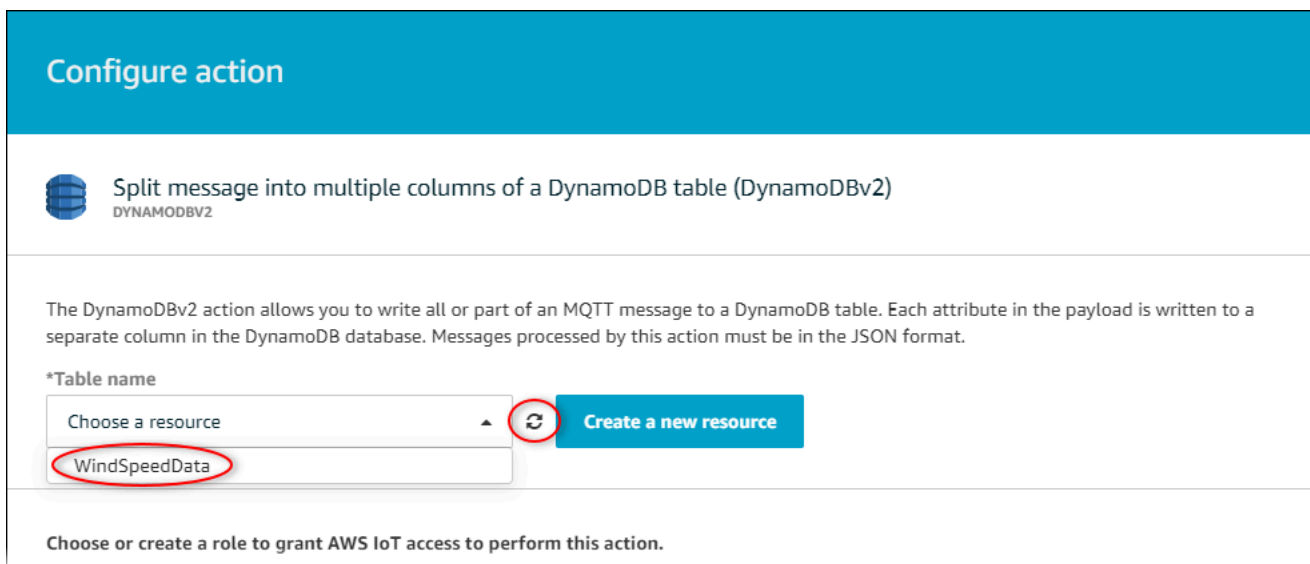
5. Kembali ke tab dengan halaman Configure action. Biarkan tab DynamoDB tetap terbuka saat Anda menyelesaikan prosedur berikut.

Langkah 4: Konfigurasi tindakan aturan DynamoDB


Dalam prosedur ini, Anda mengonfigurasi tindakan aturan Amazon DynamoDB untuk menyisipkan data dari pembaruan nilai properti ke tabel DynamoDB baru Anda.

Untuk mengkonfigurasi tindakan aturan DynamoDB

1. Pada halaman Configure action, refresh daftar nama Tabel, dan pilih tabel DynamoDB baru Anda.




Configure action

 Split message into multiple columns of a DynamoDB table (DynamoDBv2)
DYNAMODBv2

The DynamoDBv2 action allows you to write all or part of an MQTT message to a DynamoDB table. Each attribute in the payload is written to a separate column in the DynamoDB database. Messages processed by this action must be in the JSON format.

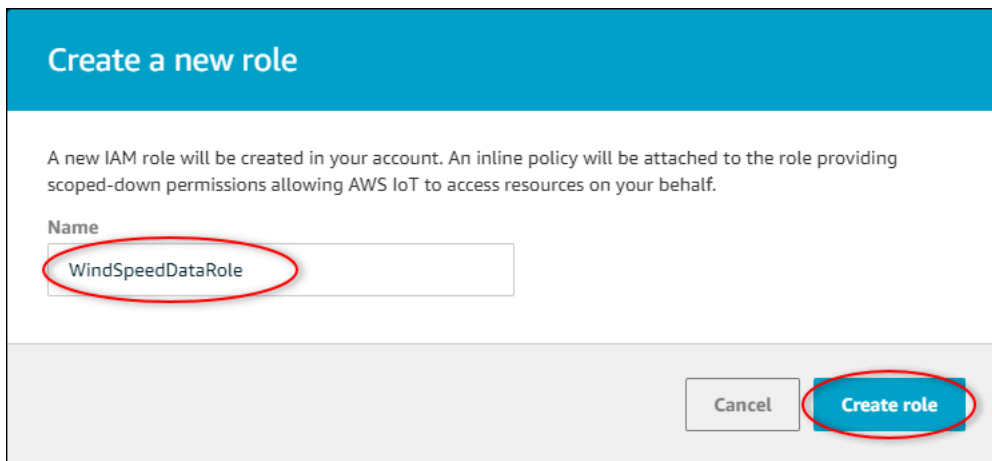
***Table name**

Choose a resource  **Create a new resource**

WindSpeedData

Choose or create a role to grant AWS IoT access to perform this action.

2. Pilih Buat peran untuk membuat peran IAM yang memberikan akses AWS IoT Core untuk melakukan tindakan aturan.
3. Masukkan nama peran dan pilih Buat peran.



4. Pilih Tambahkan tindakan.
5. Pilih Buat aturan di bagian bawah halaman untuk menyelesaikan pembuatan aturan.

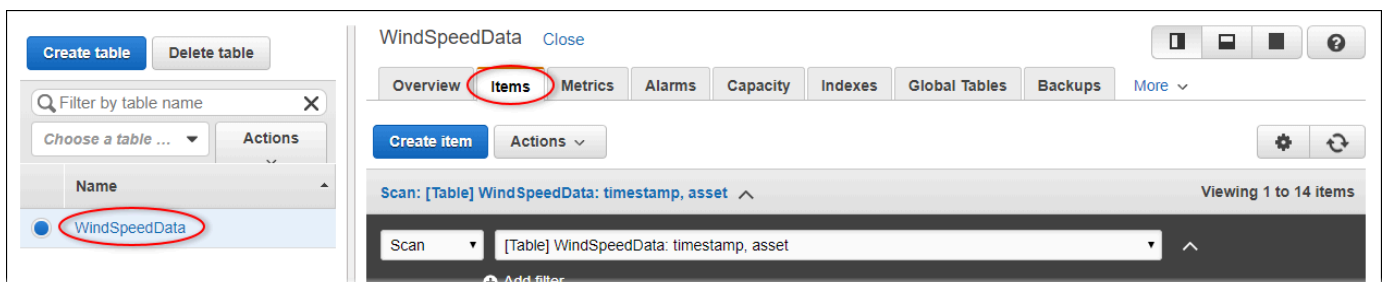
Data aset demo Anda akan mulai muncul di tabel DynamoDB Anda.

Langkah 5: Jelajahi data di DynamoDB

Dalam prosedur ini, Anda menjelajahi data kecepatan angin aset demo di tabel Amazon DynamoDB baru Anda.

Untuk mengeksplorasi data aset di DynamoDB

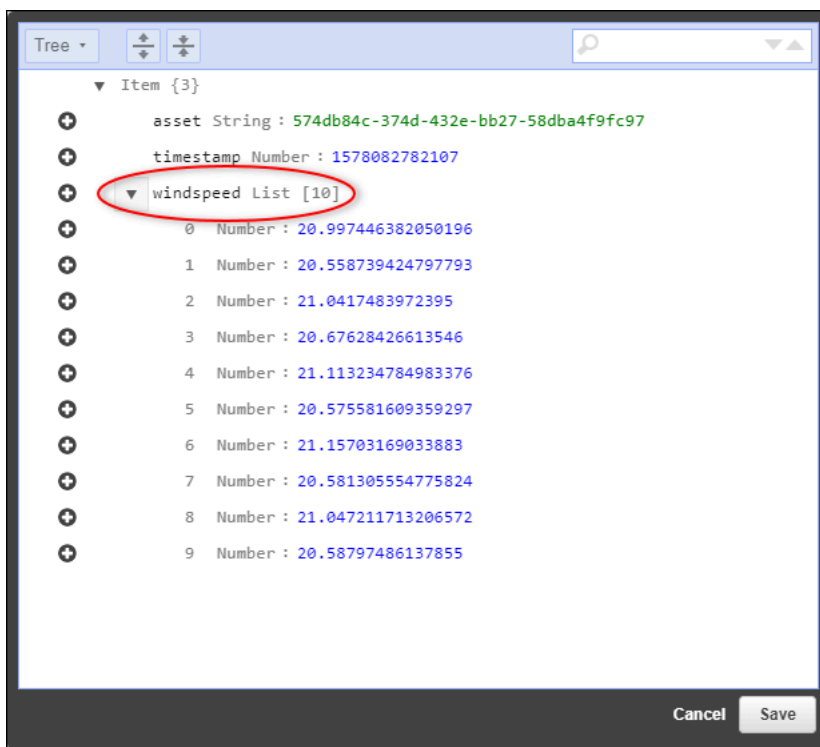
1. Kembali ke tab dengan tabel DynamoDB terbuka.
2. Dalam tabel yang Anda buat sebelumnya, pilih tab Item untuk melihat data dalam tabel. Segarkan halaman jika Anda tidak melihat baris dalam tabel. Jika baris tidak muncul setelah beberapa menit, lihat [Memecahkan masalah aturan](#).



3. Dalam satu baris dalam tabel, pilih ikon edit untuk memperluas data.

	timestamp	asset	windspeed
<input type="checkbox"/>	1578093637414	db36f80f-ed03-44d9-84ef-817eb30d5497	[{"N": "40.18707553698584"}, {"N": "40.20834808480326"}, {"N": "40..."}]
<input type="checkbox"/>	1578093637422	db36f80f-ed03-44d9-84ef-817eb30d5497	[{"N": "40.21081344172715"}, {"N": "40.218280888809424"}, {"N": "4..."}]
<input type="checkbox"/>	1578093637451	db36f80f-ed03-44d9-84ef-817eb30d5497	[{"N": "40.218912043562895"}, {"N": "40.22691091326525"}, {"N": "4..."}]
<input type="checkbox"/>	1578093637453	db36f80f-ed03-44d9-84ef-817eb30d5497	[{"N": "40.22876939941959"}, {"N": "40.21820505495924"}, {"N": "40..."}]

4. Pilih panah di sebelah windspeedstruktur untuk memperluas daftar titik data kecepatan angin. Setiap daftar mencerminkan sekumpulan titik data kecepatan angin yang dikirim AWS IoT SiteWise oleh demo ladang angin. Anda mungkin menginginkan format data yang berbeda jika Anda menyiapkan tindakan aturan untuk Anda gunakan sendiri. Untuk informasi selengkapnya, lihat [Menanyakan pesan pemberitahuan properti aset](#).



Sekarang setelah Anda menyelesaikan tutorial, nonaktifkan atau hapus aturan dan hapus tabel DynamoDB Anda untuk menghindari biaya tambahan. Untuk membersihkan sumber daya Anda, lihat [Langkah 6: Bersihkan sumber daya setelah tutorial](#).

Langkah 6: Bersihkan sumber daya setelah tutorial

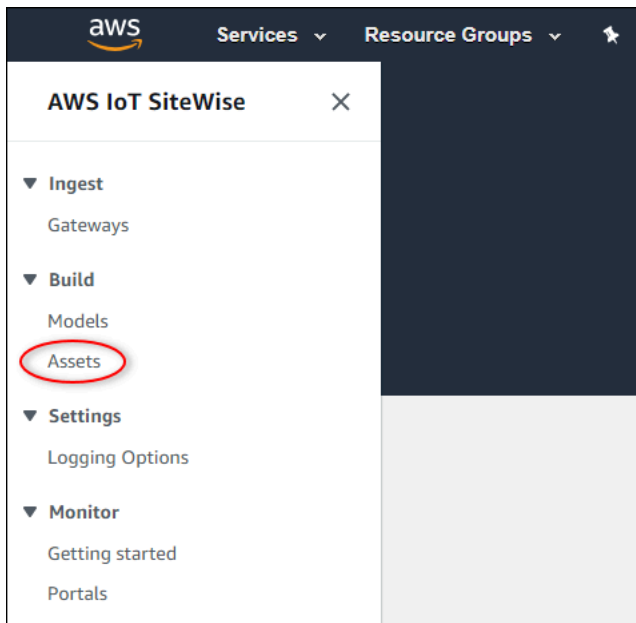
Setelah Anda menyelesaikan tutorial, bersihkan sumber daya Anda untuk menghindari biaya tambahan. Aset ladang angin demo Anda dihapus pada akhir durasi yang Anda pilih saat Anda

membuat demo. Anda juga dapat menghapus demo secara manual. Untuk informasi selengkapnya, lihat [Menghapus demo AWS IoT SiteWise](#).

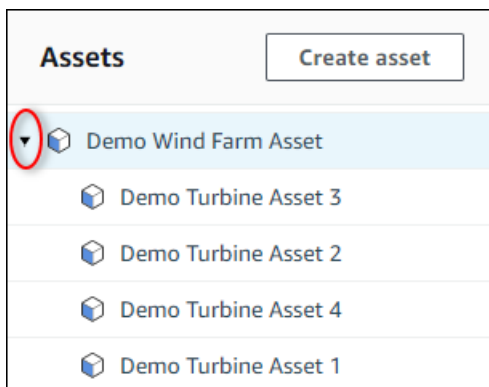
Gunakan prosedur berikut untuk menonaktifkan pemberitahuan pembaruan nilai properti (jika Anda tidak menghapus demo), menonaktifkan atau menghapus AWS IoT aturan Anda, dan menghapus tabel DynamoDB Anda.

Untuk menonaktifkan pemberitahuan pembaruan nilai properti pada properti aset

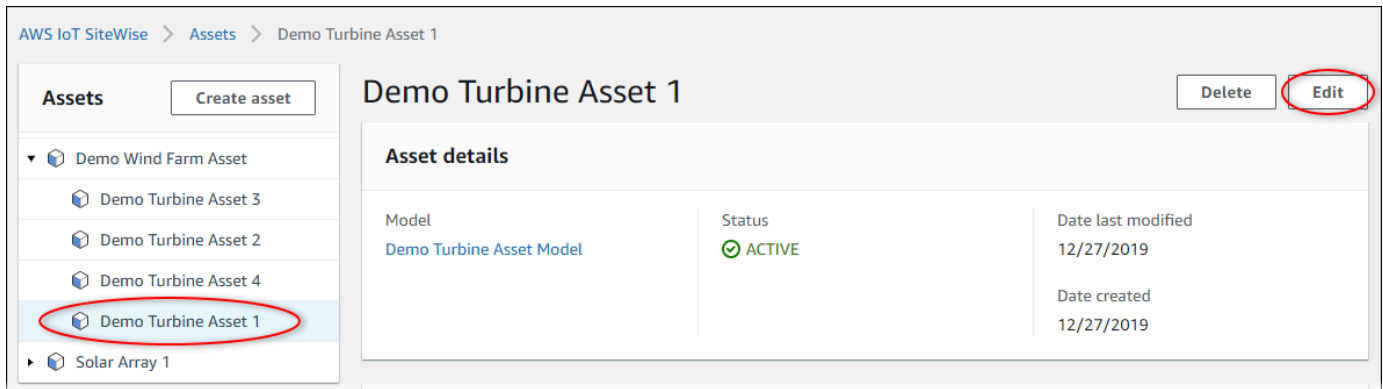
1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Aset.



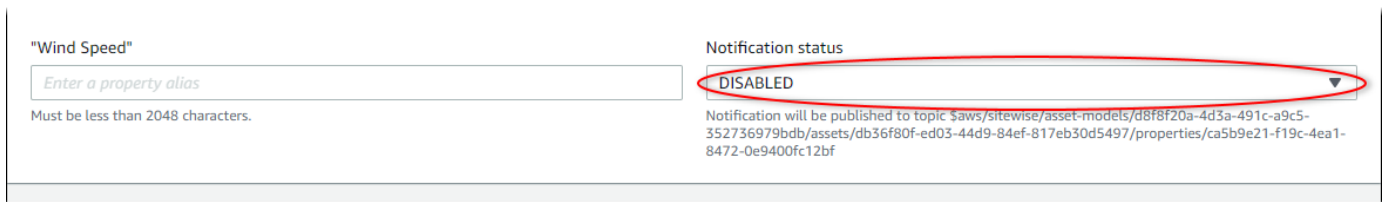
3. Pilih panah di sebelah Demo Wind Farm Asset untuk memperluas hierarki aset ladang angin.



4. Pilih turbin demo dan pilih Edit.



5. Perbarui status Pemberitahuan Wind Speed properti ke DISABLED.

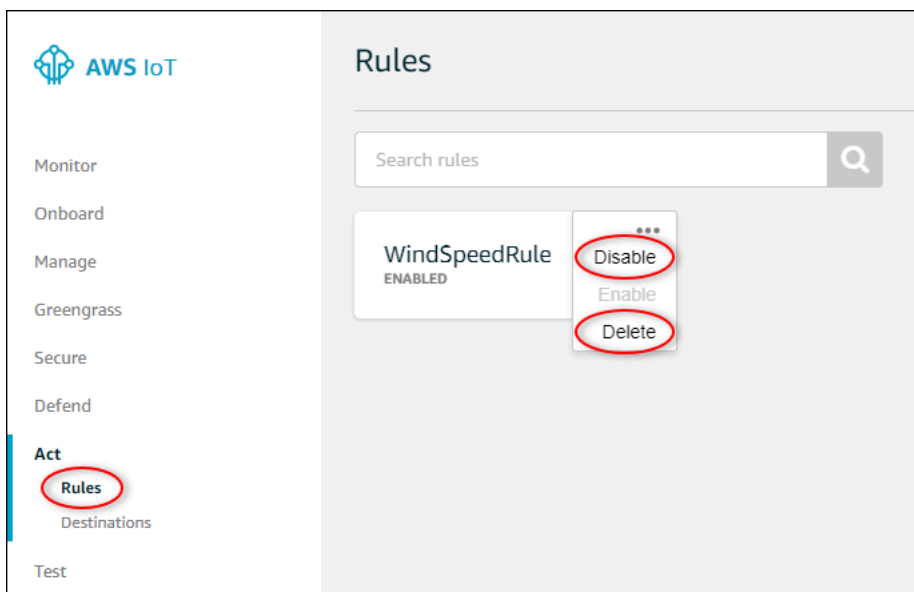


6. Pilih Simpan aset di bagian bawah halaman.

7. Ulangi langkah 4 hingga 6 untuk setiap aset turbin demo.

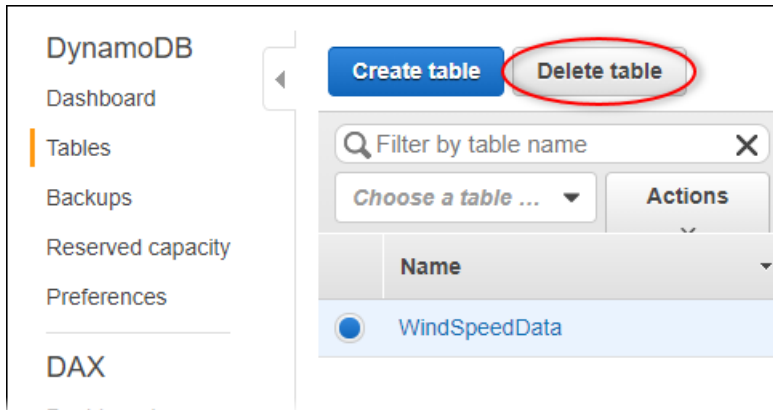
Untuk menonaktifkan atau menghapus aturan di AWS IoT Core

1. Navigasikan ke [konsol AWS IoT](#) tersebut.
2. Di panel navigasi kiri, pilih Bertindak lalu pilih Aturan.
3. Pilih menu pada aturan Anda dan pilih Nonaktifkan atau Hapus.

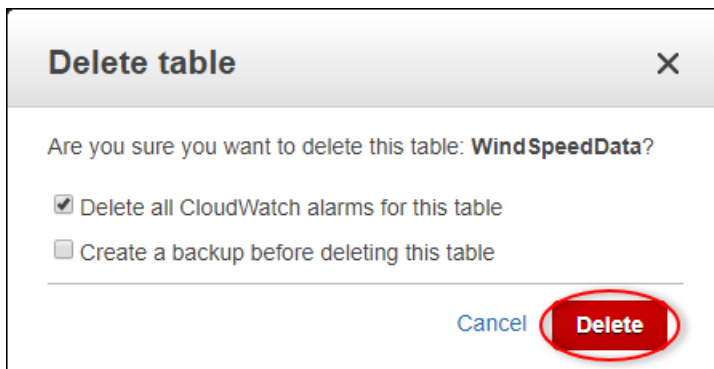


Untuk menghapus tabel DynamoDB

1. Arahkan ke konsol [DynamoDB](#).
2. Di panel navigasi kiri, pilih Tabel.
3. Pilih tabel yang Anda buat sebelumnya, WindSpeedData.
4. Pilih Hapus tabel.



5. Dalam dialog Hapus tabel, pilih Hapus.



Menelan data ke AWS IoT SiteWise

AWS IoT SiteWise dirancang untuk secara efisien mengumpulkan dan menghubungkan data industri dengan aset yang sesuai, mewakili berbagai aspek operasi industri. Dokumentasi ini berfokus pada aspek praktis dari menelan data ke dalam AWS IoT SiteWise, menawarkan beberapa metode yang disesuaikan dengan beragam kasus penggunaan industri. Untuk instruksi untuk membangun operasi industri virtual Anda, lihat [Pemodelan aset industri](#).

Anda dapat mengirim data industri untuk AWS IoT SiteWise menggunakan salah satu opsi berikut:

- AWS IoT SiteWise Edge —Gunakan [gateway SiteWise Edge](#) sebagai perantara antara AWS IoT SiteWise dan server data Anda. AWS IoT SiteWise menyediakan AWS IoT Greengrass komponen yang dapat Anda terapkan pada platform apa pun yang dapat dijalankan AWS IoT Greengrass untuk menyiapkan gateway SiteWise Edge. Opsi ini mendukung penautan dengan protokol server [OPC-UA](#).
- AWS IoT Aturan inti —Gunakan [aturan AWS IoT Inti](#) untuk mengunggah data dari pesan MQTT yang diterbitkan oleh AWS IoT sesuatu atau layanan lain. AWS
- AWS IoT Events tindakan —Gunakan [AWS IoT Events tindakan](#) yang dipicu oleh peristiwa tertentu di AWS IoT Events. Metode ini cocok untuk skenario di mana unggahan data terkait dengan kejadian peristiwa.
- AWS IoT Greengrass manajer aliran —Gunakan [pengelola AWS IoT Greengrass aliran](#) untuk mengunggah data dari sumber data lokal menggunakan perangkat edge. Opsi ini melayani situasi di mana data berasal dari lokasi lokal atau tepi.
- AWS IoT SiteWise API —Gunakan [AWS IoT SiteWise API](#) untuk mengunggah data dari sumber lain. Gunakan [BatchPutAssetPropertyValueAPI](#) streaming kami untuk konsumsi dalam hitungan detik, atau [CreateBulkImportJobAPI](#) berorientasi batch untuk memfasilitasi konsumsi hemat biaya dalam batch yang lebih besar.

Metode ini menawarkan berbagai solusi untuk mengelola data dari berbagai sumber. Selami detail setiap opsi untuk mendapatkan pemahaman komprehensif tentang kemampuan konsumsi data yang disediakan. AWS IoT SiteWise

Mengelola aliran data

Sebelum Anda terjun ke dalam pembuatan model aset dan aset AWS IoT SiteWise, mulailah dengan menyiapkan sumber data Anda untuk mengirim informasi langsung dari peralatan industri Anda

ke platform. AWS IoT SiteWise dirancang untuk secara otomatis menghasilkan aliran data yang mengumpulkan data mentah Anda. Setiap aliran data diidentifikasi oleh alias unik, sehingga lebih mudah untuk melacak asal untuk setiap bagian data.

Misalnya, pertimbangkan ladang angin menggunakan gateway AWS IoT SiteWise Edge untuk mengirim data tentang suhu udara, kecepatan putaran baling-baling, dan data deret waktu keluaran daya dari server OPC-UA ke. AWS IoT SiteWise Alias aliran `server1-windfarm/3/turbine/7/temperature` data mengidentifikasi nilai suhu yang berasal dari turbin #7 di ladang angin #3. `server1` adalah nama sumber data OPC-UA. `server1` awalan digunakan untuk semua aliran data yang berasal dari server ini, membantu mengatur data berdasarkan sumbernya.

Setelah Anda membuat model aset dan aset, atur masuknya data dengan mengaitkan setiap aliran data dengan properti aset tertentu. Asosiasi ini memungkinkan AWS IoT SiteWise untuk tidak hanya mengumpulkan, tetapi juga untuk memproses data sesuai dengan struktur aset Anda. Jika perlu, Anda juga dapat menghapus tautan antara aliran data dan properti aset.

Saat ini, Anda hanya dapat mengaitkan aliran data dengan pengukuran. Pengukuran adalah jenis properti aset yang mewakili aliran data sensor mentah perangkat, seperti nilai suhu stempel waktu atau nilai rotasi per menit (RPM) stempel waktu.

Ketika pengukuran ini menentukan metrik atau transformasi, data yang masuk memicu perhitungan tertentu. Penting untuk dicatat bahwa properti aset hanya dapat ditautkan ke satu aliran data pada satu waktu.

Note

Properti aset tidak dapat dikaitkan dengan beberapa aliran data secara bersamaan.

AWS IoT SiteWise menggunakan `TimeSeries` sumber daya Amazon Resource Name (ARN) untuk menentukan biaya penyimpanan Anda. Untuk informasi selengkapnya, silakan lihat [Harga AWS IoT SiteWise](#).

Bagian berikut menunjukkan cara menggunakan AWS IoT SiteWise konsol atau API untuk mengelola aliran data.

Topik

- [Kelola aliran data](#)

Kelola aliran data

Untuk mulai mengelola aliran data, lengkapi yang berikut ini.

Note

Jika Anda baru mengenal AWS IoT SiteWise setelah 24 November 2021, Anda dapat melewati bagian ini. Pelanggan yang mulai menggunakan AWS IoT SiteWise sebelum tanggal ini perlu mengonfigurasi pengaturan layanan untuk memungkinkan AWS IoT SiteWise untuk menelan data tanpa model aset dan aset.

- Pastikan bahwa peran IAM Anda memiliki izin yang ditunjukkan dalam contoh berikut.

Example Kebijakan pengguna IAM

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PutAssetPropertyValuesAssetPropertyOnly",
      "Effect": "Allow",
      "Action": "iotsitewise:BatchPutAssetPropertyValue",
      "Resource": "arn:aws:iotsitewise:*:*:asset/*"
    },
    {
      "Sid": "PutAssetPropertyValuesPropertyAliasAllowed",
      "Effect": "Allow",
      "Action": "iotsitewise:BatchPutAssetPropertyValue",
      "Resource": "arn:aws:iotsitewise:*:*:time-series/*"
    }
  ]
}
```

Important

Sebelum Anda menelan data ke aliran data, lakukan hal berikut.

- `time-series` Sumber daya harus diotorisasi jika Anda menggunakan alias properti untuk mengidentifikasi aliran data.

- asset Sumber daya harus diotorisasi jika Anda menggunakan ID aset untuk mengidentifikasi aset yang berisi properti aset terkait.

Untuk informasi selengkapnya tentang mengonfigurasi kebijakan IAM, lihat [Mengelola kebijakan IAM di Panduan Pengguna IAM](#).

- Konfigurasi setelah konsumsi data AWS IoT SiteWise untuk memungkinkan menerima aliran data yang tidak terkait dengan properti aset.

Topik

- [Mengkonfigurasi pengaturan konsumsi data](#)
- [Mengelola aliran data](#)

Mengkonfigurasi pengaturan konsumsi data

Console

Konfigurasi AWS IoT SiteWise untuk menerima aliran data yang tidak terkait dengan properti aset dengan menggunakan AWS IoT SiteWise konsol.

Untuk mengkonfigurasi pengaturan konsumsi data (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, di bawah Pengaturan, pilih Penyerapan data.
3. Pada halaman Data ingestion, pilih Edit.
4. Di bagian konsumsi data terputus, pilih Aktifkan konsumsi data untuk aliran data yang tidak terkait dengan properti aset.

Important

Setelah mengonfigurasi AWS IoT SiteWise untuk menerima aliran data yang tidak terkait dengan properti aset, Anda tidak dapat menonaktifkan setelah ini.

5. Pilih Simpan.

6. Di Aktifkan konsumsi data yang tidak terkait, pilih Perbarui. Status untuk konsumsi data yang tidak terkait menjadi Aktif. Proses ini bisa memakan waktu beberapa menit untuk menyelesaikannya.

AWS CLI

Konfigurasi AWS IoT SiteWise untuk menerima aliran data yang tidak terkait dengan properti aset dengan menggunakan operasi [PutStorageConfiguration](#) API. Bagian berikut menggunakan AWS CLI.

Untuk mengkonfigurasi pengaturan konsumsi data ()AWS CLI

1. Untuk mengonfigurasi AWS IoT SiteWise untuk menerima aliran data yang tidak terkait dengan properti aset, jalankan perintah berikut.

Important

Setelah mengonfigurasi AWS IoT SiteWise untuk menerima aliran data yang tidak terkait dengan properti aset, Anda tidak dapat menonaktifkan setelan ini.

```
aws iotsitewise put-storage-configuration \  
                -\-storage-type SITEWISE_DEFAULT_STORAGE \  
                -\-disassociated-data-storage ENABLED
```

Anda dapat mengkonfigurasi `storageType` ke `MULTI_LAYER_STORAGE`. Untuk informasi selengkapnya, lihat [Mengelola penyimpanan data](#).

Example response

```
{  
  "storageType": "SITEWISE_DEFAULT_STORAGE",  
  "disassociatedDataStorage": "ENABLED",  
  "configurationStatus": {  
    "state": "UPDATE_IN_PROGRESS"  
  }  
}
```

Proses ini bisa memakan waktu beberapa menit untuk menyelesaikannya.

2. Untuk mengambil informasi konfigurasi penyimpanan, jalankan perintah berikut.

```
aws iotsitewise describe-storage-configuration
```

Example response

```
{
  "storageType": "SITEWISE_DEFAULT_STORAGE",
  "disassociatedDataStorage": "ENABLED",
  "configurationStatus": {
    "state": "ACTIVE"
  },
  "lastUpdateDate": "2021-11-16T15:54:14-07:00"
}
```

Mengelola aliran data

Kelola aliran data Anda menggunakan Konsol AWS IoT SiteWise atau AWS CLI.

Console

Gunakan AWS IoT SiteWise konsol untuk mengelola aliran data Anda.

Untuk mengelola aliran data (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aliran data.
3. (Opsional) Untuk menambah atau memperbarui tag, pilih aliran data yang akan diedit, lalu pilih Kelola tag.

Pada halaman Edit tag, pilih Tambahkan tag. Di bidang Key, ketikkan nama tag yang akan digunakan.

Pilih Simpan.

4. (Opsional) Dalam tabel aliran data, Anda dapat memfilter aliran data dengan cara berikut.
 - Di menu tarik-turun pertama, pilih awalan Alias atau ID Aset.
 - Awalan alias — Awalan alias dari aliran data. Anda dapat memilih opsi ini jika aliran data target Anda memiliki awalan alias.

- ID Aset — ID aset tempat properti aset dibuat. Anda dapat memilih opsi ini jika aliran data target Anda dikaitkan dengan properti aset.
 - Di menu tarik-turun kedua, pilih Semua aliran data, Aliran data terkait, atau Aliran data yang tidak terkait.
 - Semua aliran data — Aliran data yang terkait dengan atau tidak terkait dengan properti aset.
 - Aliran data terkait — Aliran data yang terkait dengan properti aset.
 - Aliran data terputus — Aliran data yang tidak terkait dengan properti aset.
5. Pilih aliran data yang Anda kelola. AWS IoT SiteWise menampilkan aliran data yang Anda pilih dalam grafik di bagian bawah halaman. Jika Anda memilih lebih dari 10, grafik hanya akan menampilkan 10 pertama.
6. (Opsional) Konfigurasi grafik dengan cara berikut.
- a. Untuk fungsi Agregasi, pilih salah satu dari berikut ini.
- Jumlah titik data — Jumlah total titik data untuk variabel yang diberikan selama interval waktu saat ini.
 - Rata-rata — Rata-rata nilai variabel yang diberikan selama interval waktu saat ini.
 - Jumlah — Jumlah nilai variabel yang diberikan selama interval waktu saat ini.
 - Minimum — Minimum nilai variabel yang diberikan selama interval waktu saat ini.
 - Maksimum — Maksimum nilai variabel yang diberikan selama interval waktu saat ini.
- Untuk informasi selengkapnya, lihat [Menggunakan fungsi agregasi dalam ekspresi rumus](#).
- b. Untuk rentang waktu, pilih salah satu dari berikut ini.
- 1 jam terakhir — Grafik menampilkan data agregat selama satu jam terakhir.
 - 2 jam terakhir — Grafik menampilkan data agregat selama dua jam terakhir.
 - 3 jam terakhir — Grafik menampilkan data agregat selama tiga jam terakhir.
 - 4 jam terakhir — Grafik menampilkan data agregat selama empat jam terakhir.
- c. Untuk Interval waktu, pilih salah satu dari berikut ini.
- 1 menit — Mengumpulkan data setiap menit selama rentang waktu yang ditentukan.
 - 1 jam — Mengagregasi data setiap jam selama rentang waktu yang ditentukan.

7. Pilih Kelola aliran data.
8. Di bagian Perbarui asosiasi aliran data, di kolom Nama pengukuran, lakukan salah satu hal berikut.
 - Jika aliran data dikaitkan dengan pengukuran, hapus asosiasi dengan memilih ikon tutup.
 - Jika aliran data tidak terkait dengan pengukuran, pilih Pilih pengukuran.
9. Dalam tabel Pilih pengukuran, arahkan ke aset target, lalu pilih pengukuran yang Anda kaitkan.
10. (Opsional) Di bagian Perbarui alias properti aset, masukkan alias unik untuk setiap pengukuran.
11. Pilih Perbarui.

Kolom Status dapat menampilkan salah satu nilai berikut.

- Tertunda - Anda memperbarui asosiasi aliran data atau alias properti aset.
- Kirim — Perubahan Anda ke asosiasi atau alias properti aset disimpan.
- Kesalahan — AWS IoT SiteWise tidak dapat memproses permintaan Anda untuk memperbarui asosiasi aliran data atau alias untuk pengukuran.
- Sukses — Anda berhasil memperbarui asosiasi aliran data atau alias untuk pengukuran.

AWS CLI

Gunakan operasi API berikut untuk mengelola aliran data Anda. Contoh kode menggunakan AWS CLI.

- [AssociateTimeSeriesToAssetProperty](#)— Mengaitkan aliran data (time series) dengan properti aset.
- [DisassociateTimeSeriesFromAssetProperty](#)— Memutuskan aliran data dari properti aset.
- [DeleteTimeSeries](#)— Menghapus aliran data.
- [DescribeTimeSeries](#)— Mengambil informasi tentang aliran data.
- [ListTimeSeries](#)— Mengambil daftar aliran data paginasi.

AssociateTimeSeriesToAssetProperty

Untuk mengaitkan aliran data dengan properti aset, jalankan perintah berikut.

⚠ Important

Properti aset yang ditentukan saat ini tidak boleh dikaitkan dengan aliran data apa pun.

- Ganti *data-stream-alias* dengan alias aliran data yang Anda kaitkan.
- Ganti *Asset-ID* dengan ID aset tempat properti aset dibuat.
- Ganti *Property-ID* dengan ID properti aset.

```
aws iotsitewise associate-time-series-to-asset-property \  
    --alias data-stream-alias \  
    --assetId asset-ID \  
    --propertyId property-ID
```

DisassociateTimeSeriesFromAssetProperty

Untuk memisahkan aliran data dari properti aset, jalankan perintah berikut.

- Ganti *data-stream-alias* dengan alias aliran data yang Anda lepaskan.
- Ganti *Asset-ID* dengan ID aset tempat properti aset dibuat.
- Ganti *Property-ID* dengan ID properti aset.

```
aws iotsitewise disassociate-time-series-from-asset-property \  
    --alias data-stream-alias \  
    --assetId asset-ID \  
    --propertyId property-ID
```

DeleteTimeSeries

Untuk menghapus aliran data, jalankan perintah berikut.

Ganti *data-stream-alias* dengan alias aliran data yang Anda hapus.

```
aws iotsitewise delete-time-series --alias data-stream-alias
```

Untuk mengidentifikasi aliran data, lakukan salah satu hal berikut:

- Jika aliran data tidak terkait dengan properti aset, tentukan alias aliran data.

- Jika aliran data dikaitkan dengan properti aset, tentukan salah satu dari berikut ini:
 - `alias` Dari aliran data.
 - `propertyId` atau `assetId` dan yang mengidentifikasi properti aset.

DescribeTimeSeries

Gunakan operasi `DescribeTimeSeries` API untuk memverifikasi apakah Anda berhasil mengaitkan atau memisahkan aliran data.

Untuk mengambil informasi tentang aliran data, jalankan perintah berikut.

```
aws iotsitewise describe-time-series --alias data-stream-alias
```

Untuk mengidentifikasi aliran data, lakukan salah satu hal berikut:

- Jika aliran data tidak terkait dengan properti aset, tentukan `alias` aliran data.
- Jika aliran data dikaitkan dengan properti aset, tentukan salah satu dari berikut ini:
 - `alias` Dari aliran data.
 - `propertyId` atau `assetId` dan yang mengidentifikasi properti aset.

ListTimeSeries

Gunakan operasi `ListTimeSeries` API untuk memverifikasi apakah Anda berhasil menghapus aliran data.

Untuk mengambil daftar aliran data paginasi, jalankan perintah berikut.

```
aws iotsitewise list-time-series
```

Menelan data menggunakan aturan AWS IoT Core

Kirim data ke AWS IoT SiteWise dari AWS IoT hal-hal dan AWS layanan lainnya dengan menggunakan aturan di AWS IoT Core. Aturan mengubah pesan MQTT dan melakukan tindakan untuk berinteraksi dengan layanan. AWS Tindakan AWS IoT SiteWise aturan meneruskan data pesan ke [BatchPutAssetPropertyValue](#) operasi dari AWS IoT SiteWise API. Untuk informasi selengkapnya, lihat [Aturan](#) dan [AWS IoT SiteWise tindakan](#) di Panduan AWS IoT Pengembang.

Untuk mengikuti tutorial yang berjalan melalui langkah-langkah yang diperlukan untuk mengatur aturan yang menyerap data melalui bayangan perangkat, lihat [Menelan data dari AWS IoT berbagai hal](#).

Anda juga dapat mengirim data dari AWS IoT SiteWise ke AWS layanan lain. Untuk informasi selengkapnya, lihat [Berinteraksi dengan layanan lain AWS](#).

Topik

- [Memberikan akses AWS IoT yang diperlukan](#)
- [Mengkonfigurasi tindakan AWS IoT SiteWise aturan](#)
- [Mengurangi biaya dengan konsumsi dasar](#)

Memberikan akses AWS IoT yang diperlukan

Anda menggunakan peran IAM untuk mengontrol AWS sumber daya yang dapat diakses oleh setiap aturan. Sebelum membuat aturan, Anda harus membuat peran IAM dengan kebijakan yang memungkinkan aturan melakukan tindakan pada AWS sumber daya yang diperlukan. AWS IoT mengasumsikan peran ini saat menjalankan aturan.

Jika Anda membuat tindakan aturan di AWS IoT konsol, Anda dapat memilih aset root untuk membuat peran yang memiliki akses ke hierarki aset yang dipilih. Untuk informasi selengkapnya tentang cara menentukan peran aturan secara manual, lihat [Memberikan akses AWS IoT yang diperlukan](#) dan [izin peran Lulus di Panduan AWS IoT Pengembang](#).

Untuk tindakan AWS IoT SiteWise aturan, Anda harus menentukan peran yang memungkinkan `iotsitewise:BatchPutAssetPropertyValue` akses ke properti aset tempat aturan mengirimkan data. Untuk meningkatkan keamanan, Anda dapat menentukan jalur hierarki AWS IoT SiteWise aset di `Condition` properti.

Contoh kebijakan kepercayaan berikut memungkinkan akses ke aset tertentu dan anak-anaknya.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iotsitewise:BatchPutAssetPropertyValue",
      "Resource": "*",
      "Condition": {
```

```

    "StringLike": {
      "iotsitewise:assetHierarchyPath": [
        "/root node asset ID",
        "/root node asset ID/*"
      ]
    }
  }
]
}

```

Hapus Condition dari kebijakan untuk mengizinkan akses ke semua aset Anda. Contoh kebijakan kepercayaan berikut memungkinkan akses ke semua aset Anda di Wilayah saat ini.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iotsitewise:BatchPutAssetPropertyValue",
      "Resource": "*"
    }
  ]
}

```

Mengkonfigurasi tindakan AWS IoT SiteWise aturan

Tindakan AWS IoT SiteWise aturan mengirimkan data dari pesan MQTT yang memulai aturan ke properti aset di. AWS IoT SiteWise Anda dapat mengunggah beberapa entri data ke properti aset yang berbeda secara bersamaan, untuk mengirim pembaruan untuk semua sensor perangkat dalam satu pesan. Anda juga dapat mengunggah beberapa titik data sekaligus untuk setiap entri data.

Note

Saat Anda mengirim data AWS IoT SiteWise dengan tindakan aturan, data Anda harus memenuhi semua persyaratan BatchPutAssetPropertyValue operasi. Misalnya, data Anda tidak dapat memiliki stempel waktu lebih awal dari 7 hari dari waktu epoch Unix saat ini. Untuk informasi selengkapnya, lihat [Menyerap data dengan AWS IoT SiteWise API](#).

Untuk setiap entri data dalam tindakan aturan, Anda mengidentifikasi properti aset dan menentukan stempel waktu, kualitas, dan nilai setiap titik data untuk properti aset tersebut. Tindakan aturan mengharapkan string untuk semua parameter.

Untuk mengidentifikasi properti aset dalam entri, tentukan salah satu dari berikut ini:

- ID Aset (`assetId`) dan ID Properti (`propertyId`) dari properti aset yang Anda kirim data. Anda dapat menemukan ID Aset dan ID Properti menggunakan file Konsol AWS IoT SiteWise. Jika Anda mengetahui ID Aset, Anda dapat menggunakan AWS CLI to call [DescribeAsset](#) untuk menemukan ID Properti.
- Alias properti (`propertyAlias`), yang merupakan alias aliran data (misalnya, `/company/windfarm/3/turbine/7/temperature`). Untuk menggunakan opsi ini, Anda harus terlebih dahulu menetapkan alias properti aset Anda. Untuk mempelajari cara menyetel alias properti, lihat [Memetakan aliran data industri ke properti aset](#).

Untuk stempel waktu di setiap entri, gunakan stempel waktu yang dilaporkan oleh peralatan Anda atau stempel waktu yang disediakan oleh AWS IoT Core Stempel waktu memiliki dua parameter:

- Waktu dalam detik (`timeInSeconds`) — Waktu epoch Unix, dalam detik, di mana sensor atau peralatan melaporkan data.
- Offset dalam nanos (`offsetInNanos`) — (Opsional) Nanodetik offset dari waktu dalam detik.

Important

Jika stempel waktu Anda adalah string, memiliki bagian desimal, atau tidak dalam hitungan detik, AWS IoT SiteWise tolak permintaan tersebut. Anda harus mengubah stempel waktu menjadi detik dan offset nanodetik. Gunakan fitur mesin AWS IoT aturan untuk mengonversi stempel waktu. Untuk informasi selengkapnya, lihat hal berikut:

- [Mendapatkan stempel waktu untuk perangkat yang tidak melaporkan waktu yang akurat](#)
- [Mengonversi stempel waktu yang dalam format string](#)

Anda dapat menggunakan templat substitusi untuk beberapa parameter dalam tindakan untuk melakukan perhitungan, memanggil fungsi, dan menarik nilai dari muatan pesan. Untuk informasi selengkapnya, lihat [Templat substitusi](#) di Panduan AWS IoT Pengembang.

Note

Karena ekspresi dalam templat substitusi dievaluasi secara terpisah dari SELECT pernyataan, Anda tidak dapat menggunakan templat substitusi untuk mereferensikan alias yang dibuat menggunakan klausa. AS Anda hanya dapat mereferensikan informasi yang ada di muatan asli, selain fungsi dan operator yang didukung.

Topik

- [Mendapatkan stempel waktu untuk perangkat yang tidak melaporkan waktu yang akurat](#)
- [Mengonversi stempel waktu yang dalam format string](#)
- [Mengonversi string stempel waktu presisi nanodetik](#)
- [Contoh konfigurasi aturan](#)
- [Memecahkan masalah tindakan aturan](#)

Mendapatkan stempel waktu untuk perangkat yang tidak melaporkan waktu yang akurat

Jika sensor atau peralatan Anda tidak melaporkan data waktu yang akurat, dapatkan waktu epoch Unix saat ini dari mesin AWS IoT aturan dengan [stempel](#) waktu (). Fungsi ini menghasilkan waktu dalam milidetik, jadi Anda harus mengonversi nilainya menjadi waktu dalam hitungan detik dan mengimbangi dalam nanodetik. Untuk melakukannya, gunakan konversi berikut:

- Untuk Time in seconds (`timeInSeconds`), gunakan $\{\text{floor}(\text{timestamp}() / 1\text{E3})\}$ untuk mengubah waktu dari milidetik ke detik.
- Untuk Offset dalam nanos (`offsetInNanos`), gunakan $\{(\text{timestamp}() \% 1\text{E3}) * 1\text{E6}\}$ untuk menghitung offset nanodetik dari stempel waktu.

Mengonversi stempel waktu yang dalam format string

Jika sensor atau peralatan Anda melaporkan data waktu dalam format string (misalnya, `2020-03-03T14:57:14.699Z`), gunakan [time_to_epoch](#) (String, String). Fungsi ini memasukkan stempel waktu dan pola format sebagai parameter dan waktu keluaran dalam milidetik. Kemudian, Anda harus mengubah waktu ke waktu dalam hitungan detik dan mengimbangi dalam nanodetik. Untuk melakukannya, gunakan konversi berikut:

- Untuk Time in seconds (`timeInSeconds`), gunakan $\{\text{floor}(\text{time_to_epoch}("2020-03-03T14:57:14.699Z", "yyyy-MM-dd'T'HH:mm:ss'Z'") / 1E3)\}$ untuk mengonversi string timestamp menjadi milidetik, lalu ke detik.
- Untuk Offset dalam nanos (`offsetInNanos`), gunakan $\{(\text{time_to_epoch}("2020-03-03T14:57:14.699Z", "yyyy-MM-dd'T'HH:mm:ss'Z'") \% 1E3) * 1E6\}$ untuk menghitung offset nanodetik dari string stempel waktu.

Note

`time_to_epoch` Fungsi ini mendukung string stempel waktu presisi hingga milidetik. Untuk mengonversi string dengan presisi mikrodetik atau nanodetik, konfigurasi AWS Lambda fungsi yang dipanggil aturan Anda untuk mengonversi stempel waktu menjadi nilai numerik. Untuk informasi selengkapnya, lihat [Mengonversi string stempel waktu presisi nanodetik](#).

Mengonversi string stempel waktu presisi nanodetik

Jika perangkat Anda mengirimkan informasi stempel waktu dalam format string dengan presisi nanodetik (misalnya, `2020-03-03T14:57:14.699728491Z`), gunakan prosedur berikut untuk mengonfigurasi tindakan aturan Anda. Anda dapat membuat AWS Lambda fungsi yang mengubah stempel waktu dari string menjadi Time in seconds (`timeInSeconds`) dan Offset in nanos (`offsetInNanos`). Kemudian, gunakan [aws_lambda \(FunctionARN, InputJson\) dalam parameter tindakan aturan Anda untuk menjalankan fungsi Lambda itu dan menggunakan output](#) dalam aturan Anda.

Note

Bagian ini berisi petunjuk lanjutan yang mengasumsikan bahwa Anda terbiasa dengan cara membuat sumber daya berikut:

- Fungsi Lambda. Untuk informasi selengkapnya, lihat [Membuat fungsi Lambda dengan konsol](#) atau Menggunakan [Lambda dengan Panduan AWS CLI Pengembang](#).AWS Lambda
- AWS IoT aturan dengan tindakan AWS IoT SiteWise aturan. Untuk informasi selengkapnya, lihat [Menelan data menggunakan aturan AWS IoT Core](#).

Untuk membuat tindakan AWS IoT SiteWise aturan yang mem-parsing string stempel waktu

1. Buat fungsi Lambda dengan properti berikut:

- Nama fungsi — Gunakan nama fungsi deskriptif (misalnya, **ConvertNanosecondTimestampFromString**).
- Runtime — Gunakan runtime Python 3, seperti Python 3.11 (). python3.11
- Izin — Buat peran dengan izin Lambda dasar (). AWS LambdaBasicExecutionRole
- Layers — Tambahkan layer AWS SDKPandas-Python311 untuk fungsi Lambda yang akan digunakan. numpy
- Kode fungsi — Gunakan kode fungsi berikut, yang mengkonsumsi argumen string bernama `timestamp` dan output `timeInSeconds` dan `offsetInNanos` nilai untuk stempel waktu itu.

```
import json
import math
import numpy

# Converts a timestamp string into timeInSeconds and offsetInNanos in Unix epoch
time.
# The input timestamp string can have up to nanosecond precision.
def lambda_handler(event, context):
    timestamp_str = event['timestamp']
    # Parse the timestamp string as nanoseconds since Unix epoch.
    nanoseconds = numpy.datetime64(timestamp_str, 'ns').item()
    time_in_seconds = math.floor(nanoseconds / 1E9)
    # Slice to avoid precision issues.
    offset_in_nanos = int(str(nanoseconds)[-9:])
    return {
        'timeInSeconds': time_in_seconds,
        'offsetInNanos': offset_in_nanos
    }
```

[Fungsi Lambda ini memasukkan string stempel waktu dalam format ISO 8601 menggunakan `datetime64` from. NumPy](#)

Note

Jika string stempel waktu Anda tidak dalam format ISO 8601, Anda dapat menerapkan solusi yang mendefinisikan format stempel waktu. `pandas` Untuk informasi selengkapnya, lihat [pandas.to_datetime](#).

2. Saat Anda mengonfigurasi AWS IoT SiteWise tindakan untuk aturan Anda, gunakan templat substitusi berikut untuk Time in seconds (`timeInSeconds`) dan Offset in nanos (`offsetInNanos`). `offsetInNanos` Template substitusi ini mengasumsikan bahwa payload pesan Anda berisi string stempel waktu. `timestamp` `aws_lambdaFungsi` ini menggunakan struktur JSON untuk parameter kedua, sehingga Anda dapat memodifikasi template substitusi di bawah ini jika diperlukan.

- Untuk Time in seconds (`timeInSeconds`), gunakan template substitusi berikut.

```
${aws_lambda('arn:aws:lambda:region:account-id:function:ConvertNanosecondTimestampFromString', {'timestamp': timestamp}).timeInSeconds}
```

- Untuk Offset di nanos (`offsetInNanos`), gunakan template substitusi berikut.

```
${aws_lambda('arn:aws:lambda:region:account-id:function:ConvertNanosecondTimestampFromString', {'timestamp': timestamp}).offsetInNanos}
```

Untuk setiap parameter, ganti `region` dan `account-id` dengan Region dan ID AWS akun Anda. Jika Anda menggunakan nama yang berbeda untuk fungsi Lambda Anda, ubah juga.

3. Berikan AWS IoT izin untuk menjalankan fungsi Anda dengan izin. `lambda:InvokeFunction` Untuk informasi selengkapnya, lihat [aws_lambda \(FunctionARN, InputJson\)](#).
4. Uji aturan Anda (misalnya, gunakan klien pengujian AWS IoT MQTT) dan verifikasi bahwa AWS IoT SiteWise menerima data yang Anda kirim.

Jika aturan Anda tidak berfungsi seperti yang diharapkan, lihat [Memecahkan masalah tindakan aturan AWS IoT SiteWise](#).

Note

Solusi ini memanggil fungsi Lambda dua kali untuk setiap string stempel waktu. Anda dapat membuat aturan lain untuk mengurangi jumlah pemanggilan fungsi Lambda jika aturan Anda menangani beberapa titik data yang memiliki stempel waktu yang sama di setiap muatan. Untuk melakukannya, buat aturan dengan tindakan penerbitan ulang yang memanggil Lambda dan menerbitkan payload asli dengan string stempel waktu yang dikonversi ke `dan.timeInSeconds offsetInNanos`. Kemudian, buat aturan dengan tindakan AWS IoT SiteWise aturan untuk mengkonsumsi payload yang dikonversi. Dengan pendekatan ini, Anda mengurangi berapa kali aturan memanggil Lambda tetapi meningkatkan jumlah tindakan aturan AWS IoT yang dijalankan. Pertimbangkan harga setiap layanan jika Anda menerapkan solusi ini untuk kasus penggunaan Anda.

Contoh konfigurasi aturan

Bagian ini berisi contoh konfigurasi aturan untuk membuat aturan dengan AWS IoT SiteWise tindakan.

Example Contoh tindakan aturan yang menggunakan alias properti sebagai topik pesan

Contoh berikut membuat aturan dengan AWS IoT SiteWise tindakan yang menggunakan topik (melalui [topik \(\)](#)) sebagai alias properti untuk mengidentifikasi properti aset. Gunakan contoh ini untuk menentukan satu aturan untuk menelan data tipe ganda ke semua turbin angin di semua ladang angin. Contoh ini mengharuskan Anda mendefinisikan alias properti pada semua properti aset turbin. Anda perlu mendefinisikan aturan kedua yang mirip dengan menelan data tipe bilangan bulat.

```
aws iot create-topic-rule \  
  --rule-name SiteWiseWindFarmRule \  
  --topic-rule-payload file://sitewise-rule-payload.json
```

Contoh payload di `sitewise-rule-payload.json` berisi konten berikut.

```
{  
  "sql": "SELECT * FROM '/company/windfarm/+/turbine/+/+' WHERE type = 'double'",  
  "description": "Sends data to the wind turbine asset property with the same alias as  
the topic",  
  "ruleDisabled": false,  
  "awsIotSqlVersion": "2016-03-23",  
  "actions": [  

```

```

{
  "iotSiteWise": {
    "putAssetPropertyValueEntries": [
      {
        "propertyAlias": "${topic()}",
        "propertyValues": [
          {
            "timestamp": {
              "timeInSeconds": "${timeInSeconds}"
            },
            "value": {
              "doubleValue": "${value}"
            }
          }
        ]
      }
    ],
    "roleArn": "arn:aws:iam::account-id:role/MySiteWiseActionRole"
  }
}

```

Dengan tindakan aturan ini, kirim pesan berikut ke alias properti turbin angin (misalnya, /company/windfarm/3/turbine/7/temperature) sebagai topik untuk menelan data.

```

{
  "type": "double",
  "value": "38.3",
  "timeInSeconds": "1581368533"
}

```

Example Contoh aturan tindakan yang menggunakan timestamp () untuk menentukan waktu

Contoh berikut membuat aturan dengan AWS IoT SiteWise tindakan yang mengidentifikasi properti aset berdasarkan ID dan menggunakan [timestamp \(\)](#) untuk menentukan waktu saat ini.

```

aws iot create-topic-rule \
  --rule-name SiteWiseAssetPropertyRule \
  --topic-rule-payload file://sitewise-rule-payload.json

```

Contoh payload di sitewise-rule-payload.json berisi konten berikut.

```
{
  "sql": "SELECT * FROM 'my/asset/property/topic'",
  "description": "Sends device data to an asset property",
  "ruleDisabled": false,
  "awsIotSqlVersion": "2016-03-23",
  "actions": [
    {
      "iotSiteWise": {
        "putAssetPropertyValueEntries": [
          {
            "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
            "propertyId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
            "propertyValues": [
              {
                "timestamp": {
                  "timeInSeconds": "${floor(timestamp() / 1E3)}",
                  "offsetInNanos": "${(timestamp() % 1E3) * 1E6}"
                },
                "value": {
                  "doubleValue": "${value}"
                }
              }
            ]
          }
        ],
        "roleArn": "arn:aws:iam::account-id:role/MySiteWiseActionRole"
      }
    }
  ]
}
```

Dengan tindakan aturan ini, kirim pesan berikut my/asset/property/topic ke data untuk menelan.

```
{
  "type": "double",
  "value": "38.3"
}
```

Memecahkan masalah tindakan aturan

Untuk memecahkan masalah tindakan AWS IoT SiteWise aturan Anda AWS IoT Core, konfigurasi CloudWatch Log atau konfigurasi tindakan kesalahan penerbitan ulang untuk aturan Anda. Untuk informasi selengkapnya, lihat [Memecahkan masalah tindakan aturan AWS IoT SiteWise](#).

Mengurangi biaya dengan konsumsi dasar

AWS IoT Core [menyediakan fitur yang disebut Basic Ingest yang dapat Anda gunakan untuk mengirim data AWS IoT Core tanpa menimbulkan biaya pengiriman pesan.](#) AWS IoT Basic Ingest mengoptimalkan aliran data untuk beban kerja konsumsi data volume tinggi dengan menghapus broker pesan terbitkan/berlangganan dari jalur konsumsi. Anda dapat menggunakan Basic Ingest jika Anda tahu aturan mana pesan Anda harus diarahkan.

Untuk menggunakan Basic Ingest, Anda mengirim pesan langsung ke aturan tertentu menggunakan topik khusus. `$aws/rules/rule-name` Misalnya, untuk mengirim pesan ke aturan bernama `SiteWiseWindFarmRule`, Anda mengirim pesan ke topik `$aws/rules/SiteWiseWindFarmRule`.

Jika tindakan aturan Anda menggunakan templat substitusi yang berisi [topik \(Desimal\)](#), Anda dapat meneruskan topik asli di akhir topik khusus Basic Ingest, seperti `$aws/rules/rule-name/original-topic` Misalnya, untuk menggunakan Basic Ingest dengan contoh alias properti ladang angin dari bagian sebelumnya, Anda dapat mengirim pesan ke topik berikut.

```
$aws/rules/SiteWiseWindFarmRule//company/windfarm/3/turbine/7/temperature
```

Note

Contoh di atas menyertakan garis miring kedua (//) karena AWS IoT menghapus awalan Basic Ingest (`$aws/rules/rule-name/`) dari topik yang terlihat oleh tindakan aturan. Dalam contoh ini, aturan menerima topik `/company/windfarm/3/turbine/7/temperature`.

Untuk informasi selengkapnya, lihat [Mengurangi biaya pesan dengan konsumsi dasar](#) di Panduan AWS IoT Pengembang.

Menelan data dari AWS IoT Events

Dengan AWS IoT Events, Anda dapat membangun aplikasi pemantauan acara yang kompleks untuk armada IoT Anda di Cloud. AWS Gunakan SiteWise tindakan IoT AWS IoT Events untuk mengirim data ke properti aset AWS IoT SiteWise ketika suatu peristiwa terjadi.

AWS IoT Events dirancang untuk merampingkan pengembangan aplikasi pemantauan acara untuk perangkat dan sistem IoT di dalam AWS Cloud. Menggunakan AWS IoT Events, Anda dapat:

- Mendeteksi dan merespons perubahan, anomali, atau kondisi spesifik di seluruh armada IoT Anda.
- Tingkatkan efisiensi operasional Anda dan aktifkan manajemen proaktif ekosistem IoT Anda.

Dengan mengintegrasikan dengan AWS IoT SiteWise melalui AWS IoT SiteWise tindakan, AWS IoT Events memperluas kemampuannya, memungkinkan Anda memperbarui properti aset secara otomatis sebagai respons terhadap peristiwa tertentu. AWS IoT SiteWise Interaksi ini dapat menyederhanakan konsumsi dan manajemen data. Ini juga dapat memberdayakan Anda dengan wawasan yang dapat ditindaklanjuti.

Untuk informasi selengkapnya, lihat topik berikut di Panduan AWS IoT Events Pengembang:

- [Apa itu AWS IoT Events?](#)
- [AWS IoT Events tindakan](#)
- [Tindakan IoT SiteWise](#)

Menggunakan pengelola AWS IoT Greengrass aliran

AWS IoT Greengrass Stream Manager adalah fitur integrasi yang memfasilitasi transfer aliran data dari sumber lokal ke AWS Cloud. Ini bertindak sebagai lapisan perantara yang mengelola aliran data, memungkinkan perangkat yang beroperasi di tepi untuk mengumpulkan dan menyimpan data sebelum dikirim ke AWS IoT SiteWise, untuk analisis dan pemrosesan lebih lanjut.

Tambahkan tujuan data dengan mengonfigurasi sumber lokal di AWS IoT SiteWise konsol. Anda juga dapat menggunakan pengelola aliran dalam AWS IoT Greengrass solusi kustom Anda untuk menyerap data. AWS IoT SiteWise

Note

Untuk menyerap data dari sumber OPC-UA, konfigurasi gateway AWS IoT SiteWise Edge yang berjalan. AWS IoT Greengrass Untuk informasi selengkapnya, lihat [Menggunakan SiteWise gateway Edge](#).

Untuk informasi selengkapnya tentang cara mengonfigurasi tujuan untuk data sumber lokal, lihat [Mengkonfigurasi sumber data](#).

Untuk informasi selengkapnya tentang cara menyerap data menggunakan pengelola aliran dalam AWS IoT Greengrass solusi khusus, lihat topik berikut di Panduan AWS IoT Greengrass Version 2 Pengembang:

- [Apa itu AWS IoT Greengrass?](#)
- [Mengelola aliran data pada inti AWS IoT Greengrass](#)
- [Mengekspor data ke properti AWS IoT SiteWise aset](#)

Menyerap data menggunakan API AWS IoT SiteWise

Gunakan AWS IoT SiteWise API untuk mengirim data industri dengan stempel waktu ke atribut dan properti pengukuran aset Anda. API menerima payload yang berisi struktur timestamp-quality-value (TQV).

Gunakan [BatchPutAssetPropertyValue](#) operasi untuk mengunggah data Anda. Dengan operasi ini, Anda dapat mengunggah beberapa entri data sekaligus untuk mengumpulkan data dari beberapa perangkat dan mengirimkan semuanya dalam satu permintaan.

Important

[BatchPutAssetPropertyValue](#) Operasi tunduk pada kuota berikut:

- Hingga 10 [entri](#) per permintaan.
- Hingga 10 [nilai properti](#) (titik data TQV) per entri.
- AWS IoT SiteWise menolak data apa pun dengan stempel waktu bertanggal lebih dari 7 hari di masa lalu atau lebih dari 10 menit di masa depan.

Untuk informasi selengkapnya tentang kuota ini, lihat [BatchPutAssetPropertyValue](#) di Referensi AWS IoT SiteWise API.

Untuk mengidentifikasi properti aset, tentukan salah satu dari berikut ini:

- Properti `propertyId` aset yang dikirimkan ke data. `assetId`
- `ThepropertyAlias`, yang merupakan alias aliran data (misalnya, `/company/windfarm/3/turbine/7/temperature`). Untuk menggunakan opsi ini, Anda harus terlebih dahulu menetapkan alias properti aset Anda. Untuk mengatur alias properti, lihat [Memetakan aliran data industri ke properti aset](#).

Contoh berikut menunjukkan cara mengirim suhu turbin angin dan rotasi per menit (RPM) pembacaan dari muatan yang disimpan dalam file JSON.

```
aws iotsitewise batch-put-asset-property-value --cli-input-json file://batch-put-payload.json
```

Contoh payload di `batch-put-payload.json` berisi konten berikut.

```
{
  "entries": [
    {
      "entryId": "unique entry ID",
      "propertyAlias": "/company/windfarm/3/turbine/7/temperature",
      "propertyValues": [
        {
          "value": {
            "integerValue": 38
          },
          "timestamp": {
            "timeInSeconds": 1575691200
          }
        }
      ]
    },
    {
      "entryId": "unique entry ID",
```



```
    "propertyAlias": "/company/windfarm/3/turbine/7/rpm",
    "propertyValues": [
      {
        "value": {
          "doubleValue": 15.09
        },
        "timestamp": {
          "timeInSeconds": 1575691200
        },
        "quality": "GOOD"
      }
    ]
  }
]
```

Setiap entri dalam payload berisi `entryId` yang dapat Anda definisikan sebagai string unik apa pun. Jika ada entri permintaan yang gagal, setiap kesalahan akan berisi permintaan `entryId` yang sesuai sehingga Anda tahu permintaan mana yang harus dicoba lagi.

Setiap struktur dalam daftar `propertyValues` adalah struktur timestamp-quality-value (TQV) yang berisi `value`, `timestamp`, dan opsional `quality`

- `value`— Struktur yang berisi salah satu bidang berikut, tergantung pada jenis properti yang ditetapkan:
 - `booleanValue`
 - `doubleValue`
 - `integerValue`
 - `stringValue`
- `timestamp`— Struktur yang berisi waktu epoch Unix saat ini dalam hitungan detik,. `timeInSeconds` Anda juga dapat mengatur `offsetInNanos` kunci dalam `timestamp` struktur jika Anda memiliki data yang tepat untuk sementara. AWS IoT SiteWise menolak setiap titik data dengan stempel waktu yang lebih tua dari 7 hari di masa lalu atau lebih baru dari 10 menit di masa depan.
- `quality`— (Opsional) Salah satu string kualitas berikut:
 - `GOOD`— (Default) Data tidak terpengaruh oleh masalah apa pun.
 - `BAD`— Data dipengaruhi oleh masalah seperti kegagalan sensor.
 - `UNCERTAIN`— Data dipengaruhi oleh masalah seperti ketidakakuratan sensor.

Untuk informasi selengkapnya tentang cara AWS IoT SiteWise menangani kualitas data dalam komputasi, lihat [Kualitas data dalam ekspresi rumus](#).

Menelan data menggunakan API CreateBulkImportJob

Gunakan `CreateBulkImportJob` API untuk mengimpor data dalam jumlah besar dari Amazon S3. Data Anda harus disimpan dalam format CSV di Amazon S3. File data dapat memiliki kolom berikut.

Note

Untuk mengidentifikasi properti aset, tentukan salah satu dari berikut ini.

- Properti `PROPERTY_ID` aset yang Anda kirim datanya. `ASSET_ID`
- `TheALIAS`, yang merupakan alias aliran data (misalnya, `/company/windfarm/3/turbine/7/temperature`). Untuk menggunakan opsi ini, Anda harus terlebih dahulu menetapkan alias properti aset Anda. Untuk mempelajari cara menyetel alias properti, lihat [the section called “Memetakan aliran data industri ke properti aset”](#).

- `ALIAS`— Alias yang mengidentifikasi properti, seperti jalur aliran data server OPC-UA (misalnya, `/company/windfarm/3/turbine/7/temperature`). Untuk informasi selengkapnya, lihat [Memetakan aliran data industri ke properti aset](#).
- `ASSET_ID`— ID aset.
- `PROPERTY_ID`— ID properti aset.
- `DATA_TYPE` Jenis data properti dapat menjadi salah satu dari berikut ini.
 - `STRING`— Sebuah string dengan hingga 1024 byte.
 - `INTEGER`— Bilangan bulat 32-bit yang ditandatangani dengan rentang `[-2,147.483.648, 2.147.483.647]`.
 - `DOUBLE`— Nomor floating point dengan jangkauan `[-10100, 10100]` dan presisi ganda IEEE 754.
 - `BOOLEAN`— `true` atau `false`.
- `TIMESTAMP_SECONDS`— Stempel waktu titik data, dalam waktu zaman Unix.
- `TIMESTAMP_NANO_OFFSET`— Offset nanodetik tertutup dari `TIMESTAMP_SECONDS`
- `QUALITY`— (Opsional) Kualitas nilai properti aset. Nilainya bisa menjadi salah satu dari berikut ini.

- GOOD— (Default) Data tidak terpengaruh oleh masalah apa pun.
- BAD— Data dipengaruhi oleh masalah seperti kegagalan sensor.
- UNCERTAIN— Data dipengaruhi oleh masalah seperti ketidakakuratan sensor.

Untuk informasi selengkapnya tentang cara AWS IoT SiteWise menangani kualitas data dalam komputasi, lihat [Kualitas data dalam ekspresi rumus](#).

- VALUE— Nilai properti aset.

Example file data dalam format.csv

```
asset_id,property_id,DOUBLE,1635201373,0,GOOD,1.0
asset_id,property_id,DOUBLE,1635201374,0,GOOD,2.0
asset_id,property_id,DOUBLE,1635201375,0,GOOD,3.0
```

```
unmodeled_alias1,DOUBLE,1635201373,0,GOOD,1.0
unmodeled_alias1,DOUBLE,1635201374,0,GOOD,2.0
unmodeled_alias1,DOUBLE,1635201375,0,GOOD,3.0
unmodeled_alias1,DOUBLE,1635201376,0,GOOD,4.0
unmodeled_alias1,DOUBLE,1635201377,0,GOOD,5.0
unmodeled_alias1,DOUBLE,1635201378,0,GOOD,6.0
unmodeled_alias1,DOUBLE,1635201379,0,GOOD,7.0
unmodeled_alias1,DOUBLE,1635201380,0,GOOD,8.0
unmodeled_alias1,DOUBLE,1635201381,0,GOOD,9.0
unmodeled_alias1,DOUBLE,1635201382,0,GOOD,10.0
```

AWS IoT SiteWise menyediakan operasi API berikut untuk membuat pekerjaan impor massal dan mendapatkan informasi tentang pekerjaan yang ada.

- [CreateBulkImportJob](#)— Membuat pekerjaan impor massal baru.
- [DescribeBulkImportJob](#)— Mengambil informasi tentang pekerjaan impor massal.
- [ListBulkImportJob](#)— Mengambil daftar ringkasan halaman dari semua pekerjaan impor massal.

Buat pekerjaan impor massal (AWS CLI)

Gunakan operasi [CreateBulkImportJob](#) API untuk mentransfer data dari Amazon S3 ke. AWS IoT SiteWise Gunakan [CreateBulkImportJob](#) API untuk menyerap data dalam batch kecil dengan cara yang hemat biaya. Contoh berikut menggunakan AWS CLI.

⚠ Important

Sebelum membuat pekerjaan impor massal, Anda harus mengaktifkan tingkat AWS IoT SiteWise hangat atau tingkat AWS IoT SiteWise dingin. Untuk informasi selengkapnya, lihat [Konfigurasi pengaturan penyimpanan](#).

Impor massal dirancang untuk menyimpan data historis ke AWS IoT SiteWise. Itu tidak memulai perhitungan atau pemberitahuan pada tingkat AWS IoT SiteWise hangat atau tingkat AWS IoT SiteWise dingin.

Jalankan perintah berikut. Ganti *nama file* dengan nama file yang berisi konfigurasi pekerjaan impor massal.

```
aws iotsitewise create-bulk-import-job --cli-input-json file://file-name.json
```

Example Konfigurasi pekerjaan impor massal

Berikut ini adalah contoh pengaturan konfigurasi:

- Ganti *adaptive-ingestion-flag* dengan `true` atau `false`.
 - Jika disetel ke `false`, pekerjaan impor massal akan memasukkan data historis ke dalam AWS IoT SiteWise.
 - Jika disetel ke `true`, pekerjaan impor massal melakukan hal berikut:
 - Menyerap data baru ke dalam AWS IoT SiteWise.
 - Menghitung metrik dan transformasi, dan mendukung pemberitahuan untuk data dengan cap waktu yang dalam tujuh hari.
- Ganti *delete-files-after-import-flag* dengan `true` untuk menghapus data dari bucket data S3 setelah masuk ke penyimpanan tingkat AWS IoT SiteWise hangat.
- Ganti *error-bucket* dengan nama bucket Amazon S3 tempat kesalahan yang terkait dengan pekerjaan impor massal ini dikirim.
- Ganti *error-bucket-prefix* dengan awalan bucket Amazon S3 tempat kesalahan yang terkait dengan pekerjaan impor massal ini dikirim.

Amazon S3 menggunakan awalan sebagai nama folder untuk mengatur data dalam bucket. Setiap objek Amazon S3 memiliki kunci yang merupakan pengenal uniknya di bucket. Setiap objek dalam bucket memiliki satu kunci. Prefiks harus diakhiri dengan garis miring ke depan (/).

Untuk informasi selengkapnya, lihat [Mengatur objek menggunakan awalan](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

- Ganti *data-bucket* dengan nama bucket Amazon S3 dari mana data diimpor.
- Ganti *data-bucket-key* dengan kunci objek Amazon S3 yang berisi data Anda. Setiap objek memiliki kunci yang berupa pengidentifikasi unik. Setiap objek memiliki tepat satu kunci.
- Ganti *data-bucket-version-id* dengan ID versi untuk mengidentifikasi versi tertentu dari objek Amazon S3 yang berisi data Anda. Parameter ini bersifat opsional.
- Ganti nama *kolom dengan nama* kolom yang ditentukan dalam file.csv.
- Ganti *job-name* dengan nama unik yang mengidentifikasi pekerjaan impor massal.
- Ganti *job-role-arn* dengan peran IAM yang memungkinkan AWS IoT SiteWise untuk membaca data Amazon S3.

Note

Pastikan peran Anda memiliki izin yang ditunjukkan dalam contoh berikut. Ganti *data-bucket* dengan nama bucket Amazon S3 yang berisi data Anda. Selain itu, ganti *error-bucket* dengan nama bucket Amazon S3 tempat kesalahan yang terkait dengan pekerjaan impor massal ini dikirim.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:GetBucketLocation"
      ],
      "Resource": [
        "arn:aws:s3:::data-bucket",
        "arn:aws:s3:::data-bucket/*",
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:GetBucketLocation"
      ],
```

```

    ],
    "Resource": [
      "arn:aws:s3:::error-bucket",
      "arn:aws:s3:::error-bucket/*"
    ],
    "Effect": "Allow"
  }
]
}

```

```

{
  "adaptiveIngestion": adaptive-ingestion-flag,
  "deleteFilesAfterImport": delete-files-after-import-flag,
  "errorReportLocation": {
    "bucket": "error-bucket",
    "prefix": "error-bucket-prefix"
  },
  "files": [
    {
      "bucket": "data-bucket",
      "key": "data-bucket-key",
      "versionId": "data-bucket-version-id"
    }
  ],
  "jobConfiguration": {
    "fileFormat": {
      "csv": {
        "columnNames": [ "column-name" ]
      }
    }
  },
  "jobName": "job-name",
  "jobRoleArn": "job-role-arn"
}

```

Example response

```

{
  "jobId": "f8c031d0-01d1-4b94-90b1-afe8bb93b7e5",
  "jobStatus": "PENDING",
  "jobName": "myBulkImportJob"
}

```

```
}
```

Jelaskan pekerjaan impor massal (AWS CLI)

Gunakan operasi [DescribeBulkImportJob](#) API untuk mengambil informasi tentang pekerjaan impor massal. Contoh berikut menggunakan AWS CLI.

Ganti *Job-ID* dengan ID pekerjaan impor massal yang ingin Anda ambil.

```
aws iotsitewise describe-bulk-import-job --job-id job-ID
```

Example response

```
{
  "files": [
    {
      "bucket": "test-bucket",
      "key": "100Tags12Hours.csv"
    },
    {
      "bucket": "test-bucket",
      "key": "BulkImportData1MB.csv"
    },
    {
      "bucket": "test-bucket",
      "key": "UnmodeledBulkImportData1MB.csv"
    }
  ],
  "errorReportLocation": {
    "prefix": "errors/",
    "bucket": "test-error-bucket"
  },
  "jobConfiguration": {
    "fileFormat": {
      "csv": {
        "columnNames": [
          "ALIAS",
          "DATA_TYPE",
          "TIMESTAMP_SECONDS",
          "TIMESTAMP_NANO_OFFSET",
          "QUALITY",
          "VALUE"
        ]
      }
    }
  }
}
```

```
    ]
  }
}
},
"jobCreationDate":1645745176.498,
"jobStatus":"COMPLETED",
"jobName":"myBulkImportJob",
"jobLastUpdateDate":1645745279.968,
"jobRoleArn":"arn:aws:iam::123456789012:role/DemoRole",
"jobId":"f8c031d0-01d1-4b94-90b1-afe8bb93b7e5"
}
```

Daftar pekerjaan impor massal (AWS CLI)

Gunakan operasi [ListBulkImportJobs](#) API untuk mengambil daftar ringkasan halaman dari semua pekerjaan impor massal. Contoh berikut menggunakan AWS CLI.

```
aws iotsitewise list-bulk-import-jobs --filter COMPLETED
```

Example response

```
{
  "jobSummaries":[
    {
      "id":"bdbbfa52-d775-4952-b816-13ba1c7cb9da",
      "name":"myBulkImportJob",
      "status":"COMPLETED"
    },
    {
      "id":"15ffc641-dbd8-40c6-9983-5cb3b0bc3e6b",
      "name":"myBulkImportJob2",
      "status":"RUNNING"
    }
  ]
}
```


Menggunakan SiteWise gateway Edge

Gerbang AWS IoT SiteWise Edge berfungsi sebagai perantara antara peralatan industri Anda dan AWS IoT SiteWise. Gateway SiteWise Edge berjalan di AWS IoT Greengrass V2 yang mendukung pengumpulan dan pemrosesan data di tempat. Anda dapat menggunakannya AWS OpsHub AWS IoT SiteWise untuk mengelola gateway SiteWise Edge Anda dan memantau operasi di tempat.

Anda dapat memantau data secara lokal di fasilitas Anda menggunakan portal SiteWise Monitor di perangkat lokal Anda. Untuk informasi selengkapnya, lihat [Mengaktifkan portal Anda di tepi](#).

Topik

- [SiteWise Persyaratan gateway tepi](#)
- [Membuat gateway SiteWise Edge](#)
- [Menginstal perangkat lunak gateway SiteWise Edge di perangkat lokal Anda](#)
- [Mengaktifkan pemrosesan data tepi](#)
- [Memproses data di tepi](#)
- [Mengkonfigurasi komponen AWS IoT SiteWise Publisher](#)
- [Mengkonfigurasi sumber data](#)
- [Menambahkan sumber data mitra ke gateway SiteWise Edge](#)
- [Menggunakan paket](#)
- [Mengelola SiteWise gateway Edge](#)
- [Running SiteWise Edge di Tepi Industri Siemens](#)
- [Memfilter aset di gateway SiteWise Edge](#)
- [Menggunakan AWS IoT SiteWise API di tepi](#)
- [Cadangkan dan pulihkan SiteWise gateway Edge](#)
- [Menyiapkan gateway SiteWise Edge \(AWS IoT Greengrass Version 1\)](#)

SiteWise Persyaratan gateway tepi

AWS IoT SiteWise Edge gateway berjalan AWS IoT Greengrass V2 sebagai satu set AWS IoT Greengrass komponen yang mendukung pengumpulan data, pemrosesan, dan penerbitan di tempat.

Untuk mengonfigurasi gateway SiteWise Edge yang berjalan AWS IoT Greengrass V2, Anda harus membuat gateway di AWS Cloud dan menjalankan perangkat lunak gateway SiteWise Edge untuk menyiapkan perangkat lokal Anda.

Persyaratan

Perangkat lokal harus memenuhi persyaratan berikut untuk menginstal dan menjalankan perangkat lunak gateway SiteWise Edge.

- Mendukung perangkat lunak AWS IoT Greengrass V2 Core versi [v2.3.0](#) atau yang lebih baru. Untuk informasi selengkapnya, lihat [Persyaratan](#) dalam Panduan AWS IoT Greengrass Version 2 Pengembang.
- Salah satu platform yang didukung berikut:
 - OS: Ubuntu 20.04 atau yang lebih baru
Arsitektur: x86_64 (AMD64) atau ARMv8 (Aarch64)
 - OS: Perusahaan Red Hat Linux (RHEL) 8
Arsitektur: x86_64 (AMD64) atau ARMv8 (Aarch64)
 - OS: Amazon Linux 2
Arsitektur: x86_64 (AMD64) atau ARMv8 (Aarch64)
 - OS: Debian 11
Arsitektur: x86_64 (AMD64) atau ARMv8 (Aarch64)
 - OS: Windows Server 2019 dan yang lebih baru
Arsitektur: x86_64 (AMD64)

Note

Platform ARM mendukung gateway SiteWise Edge hanya dengan Paket Pengumpulan Data. Paket Pemrosesan Data tidak didukung.

- Minimal 4 GB RAM.
- Ruang disk minimal 10 GB tersedia untuk perangkat lunak gateway SiteWise Edge.
- Jika Anda berencana untuk memproses data di edge AWS IoT SiteWise, perangkat lokal Anda juga harus memenuhi persyaratan berikut:

- Memiliki prosesor quad-core x86 64 bit.
- Memiliki setidaknya 16 GB RAM.
- Memiliki setidaknya 32 GB untuk RAM jika menggunakan Windows.
- Memiliki setidaknya 256 GB ruang disk kosong.
- Ruang disk minimum dan persyaratan kapasitas komputasi bergantung pada berbagai faktor yang unik untuk implementasi dan kasus penggunaan Anda.
- Ruang disk yang diperlukan untuk caching data untuk konektivitas internet intermiten tergantung pada faktor-faktor berikut:
 - Jumlah aliran data yang diunggah
 - Titik data per aliran data per detik
 - Ukuran setiap titik data
 - Kecepatan komunikasi
 - Waktu henti jaringan yang diharapkan
- Kapasitas komputasi yang diperlukan untuk melakukan polling dan mengunggah data tergantung pada faktor-faktor berikut:
 - Jumlah aliran data yang diunggah
 - Titik data per aliran data per detik
- Konfigurasi perangkat lokal Anda untuk memastikan bahwa port berikut dapat diakses:
 - Perangkat lokal harus mengizinkan lalu lintas masuk jaringan pada port 443.
 - Perangkat lokal harus memungkinkan lalu lintas keluar pada port 443 dan 8883.

Untuk daftar lengkap titik akhir layanan keluar yang diperlukan, lihat Titik akhir layanan yang [diperlukan untuk AWS IoT SiteWise gateway Edge](#).

- Port berikut dicadangkan untuk digunakan oleh AWS IoT SiteWise: 80, 443, 3001, 4569, 4572, 8000, 8081, 8082, 8084, 8085, 8445, 8086, 9000, 9500, 11080, dan 50010. Menggunakan port cadangan untuk lalu lintas dapat mengakibatkan koneksi terputus.
- Java Runtime Environment (JRE) versi 11 atau lebih tinggi. Java harus tersedia pada variabel PATH lingkungan pada perangkat. Untuk menggunakan Java untuk mengembangkan komponen kustom, Anda harus menginstal Java Development Kit (JDK). [Kami menyarankan Anda menggunakan Amazon Corretto atau OpenJDK](#).

Anda harus memiliki izin berikut untuk menggunakan gateway SiteWise Edge:

Note

Jika Anda menggunakan AWS IoT SiteWise konsol untuk membuat gateway SiteWise Edge Anda, izin ini ditambahkan untuk Anda.

- Peran IAM untuk gateway SiteWise Edge Anda harus memungkinkan Anda menggunakan gateway SiteWise Edge pada AWS IoT Greengrass V2 perangkat untuk memproses data model aset dan data aset.

Peran tersebut memungkinkan layanan berikut untuk mengambil `peran:credentials.iot.amazonaws.com`.

Detail izin

Peran harus memiliki izin berikut:

- `iotsitewise`— Memungkinkan prinsipal untuk mengambil data model aset dan data aset di tepi.
- `iot`— Memungkinkan AWS IoT Greengrass V2 perangkat Anda untuk berinteraksi dengan AWS IoT.
- `logs`— Memungkinkan AWS IoT Greengrass V2 perangkat Anda mengirim log ke Amazon CloudWatch Logs.
- `s3`— Memungkinkan AWS IoT Greengrass V2 perangkat Anda mengunduh artefak komponen khusus dari Amazon S3.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:BatchPutAssetPropertyValue",
        "iotsitewise:List*",
        "iotsitewise:Describe*",
        "iotsitewise:Get*"
      ],
      "Resource": "*"
    },
    {
```


```
    "Effect": "Allow",
    "Action": [
        "iot:DescribeCertificate",
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogStreams",
        "s3:GetBucketLocation",
        "s3:GetObject",
        "iot:Connect",
        "iot:Publish",
        "iot:Subscribe",
        "iot:Receive",
        "iot:DescribeEndpoint"
    ],
    "Resource": "*"
}
]
```

Membuat gateway SiteWise Edge

Anda dapat menggunakan AWS IoT SiteWise konsol untuk membuat gateway SiteWise Edge. Prosedur ini merinci cara membuat gateway SiteWise Edge yang dihosting sendiri yang akan Anda instal di perangkat keras Anda sendiri. Untuk informasi tentang membuat gateway SiteWise Edge yang berjalan di Siemens Industrial Edge, lihat [Running SiteWise Edge di Tepi Industri Siemens](#)


Buat gateway SiteWise Edge

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih gateway Edge.
3. Pilih Buat gateway.
4. Untuk jenis Deployment, pilih Gateway yang di-host sendiri.
5. Masukkan nama untuk gateway SiteWise Edge Anda atau gunakan nama yang dihasilkan oleh AWS IoT SiteWise.
6. Di bawah Greengrass device OS, pilih sistem operasi perangkat tempat Anda akan menginstal gateway Edge ini. SiteWise

 Note

Paket Pemrosesan Data hanya tersedia di platform x86.

7. (Opsional) Untuk memproses dan mengatur data di tepi, di bawah kemampuan Edge, pilih Paket Pemrosesan Data.

 Note

Untuk memberikan akses kepada grup pengguna di direktori perusahaan Anda ke gateway SiteWise Edge ini, lihat [Menyiapkan kemampuan tepi](#)

8. (Opsional) Di bawah konfigurasi lanjutan, lakukan hal berikut:
 - Untuk perangkat inti Greengrass, pilih salah satu opsi berikut:
 - Pengaturan default —: AWS secara otomatis menggunakan pengaturan default untuk membuat perangkat inti Greengrass di. AWS IoT Greengrass V2
 1. Masukkan nama untuk perangkat inti Greengrass atau gunakan nama yang dihasilkan oleh. AWS IoT SiteWise
 - Pengaturan lanjutan -: Pilih opsi ini jika Anda ingin menggunakan perangkat inti Greengrass yang ada atau membuatnya secara manual.
 1. Pilih perangkat inti Greengrass atau pilih Buat perangkat inti Greengrass untuk membuatnya di konsol. AWS IoT Greengrass V2 Untuk informasi selengkapnya, lihat [Menyiapkan perangkat AWS IoT Greengrass V2 inti](#) di Panduan AWS IoT Greengrass Version 2 Pengembang.
9. Pilih Buat gateway.
10. Dalam kotak dialog Generate SiteWise Edge gateway installer, pilih Hasilkan dan unduh. AWS IoT SiteWise secara otomatis menghasilkan penginstal yang dapat Anda gunakan untuk mengonfigurasi perangkat lokal Anda.

 Important

Pastikan Anda menyimpan file installer di lokasi yang aman. Anda akan menggunakan file nanti.

Sekarang setelah Anda membuat gateway SiteWise Edge, tambahkan [sumber data](#), konfigurasi [komponen penerbit](#), dan dapatkan gateway SiteWise Edge Anda menerima data dan mengirimkannya ke AWS Cloud.

Menginstal perangkat lunak gateway SiteWise Edge di perangkat lokal Anda

Setelah Anda membuat gateway SiteWise Edge, Anda perlu menginstal perangkat lunak gateway SiteWise Edge di perangkat lokal Anda. SiteWise Perangkat lunak gateway Edge dapat diinstal pada perangkat lokal yang memiliki sistem operasi server Linux atau Windows diinstal.

Important

Pastikan perangkat lokal Anda terhubung ke internet.

Linux

Prosedur berikut menggunakan SSH untuk terhubung ke perangkat lokal Anda. Atau, Anda dapat menggunakan USB flash drive atau alat lain untuk mentransfer file installer ke perangkat lokal Anda. Jika Anda tidak ingin menggunakan SSH, lewati ke Langkah 2: Instal perangkat lunak gateway SiteWise Edge di bawah ini.

Prasyarat SSH

Sebelum Anda terhubung ke perangkat Anda menggunakan SSH, lengkapi prasyarat berikut.

- Dapatkan alamat IP perangkat Anda.
- Dapatkan nama pengguna untuk terhubung ke perangkat Anda.
- Instal klien SSH di komputer lokal Anda sesuai kebutuhan.

Komputer lokal Anda kemungkinan telah menginstal klien SSH secara default. Anda dapat memverifikasi ini dengan mengetikkan ssh baris perintah. Jika komputer Anda tidak mengenali perintah tersebut, Anda dapat menginstal klien SSH.

- Linux dan macOS - Unduh dan instal OpenSSH. Untuk informasi selengkapnya, lihat <https://www.openssh.com>.

Langkah 1: Salin penginstal ke perangkat gateway SiteWise Edge Anda

Petunjuk berikut menjelaskan cara menghubungkan ke perangkat lokal Anda menggunakan klien SSH.

1. Untuk terhubung ke perangkat Anda, jalankan perintah berikut di jendela terminal di komputer Anda, ganti *nama pengguna* dan *IP dengan nama pengguna* yang memiliki priveleges dan alamat IP yang tinggi.

```
ssh username@IP
```

2. Untuk mentransfer file penginstal yang AWS IoT SiteWise dihasilkan ke perangkat gateway SiteWise Edge Anda, jalankan perintah berikut.

Note

- Ganti *path-to-saved-installer* dengan path di komputer Anda yang Anda gunakan untuk menyimpan file installer dan nama file installer.
- Ganti *alamat IP dengan alamat* IP perangkat lokal Anda.
- Ganti *directory-to-receive-installer* dengan jalur di perangkat lokal yang Anda gunakan untuk menerima file penginstal.

```
scp path-to-saved-installer.sh user-name@IP-address:directory-to-receive-installer
```

Langkah 2: Instal perangkat lunak gateway SiteWise Edge

Dalam prosedur berikut, jalankan perintah di jendela terminal pada perangkat gateway SiteWise Edge Anda.

1. Berikan file penginstal izin eksekusi.

```
chmod +x path-to-installer.sh
```

2. Jalankan pemasang.


```
sudo ./path-to-installer.sh
```

Windows server

Prasyarat

Anda harus memiliki prasyarat berikut untuk menginstal perangkat lunak gateway Edge: SiteWise

- Windows Server 2019 atau yang lebih baru diinstal
- Hak istimewa administrator
- PowerShell versi 5.1 atau yang lebih baru diinstal
- SiteWise Edge gateway installer diunduh ke Windows Server di mana ia akan disediakan

Langkah 1: Jalankan PowerShell sebagai administrator

1. Di server Windows tempat Anda ingin menginstal gateway SiteWise Edge, masuk sebagai administrator.
2. Masukkan PowerShell di bilah pencarian Windows.
3. Dalam hasil pencarian, buka menu konteks (klik kanan) pada PowerShell aplikasi Windows. Pilih Jalankan sebagai Administrator.

Langkah 2: Instal perangkat lunak gateway SiteWise Edge

Jalankan perintah berikut di jendela terminal pada perangkat SiteWise Edge Gateway Anda.

1. Buka blokir penginstal gateway SiteWise Edge.

```
unblock-file path-to-installer.ps1
```

2. Jalankan Installer.

```
./path-to-installer.ps1
```

Note

Jika eksekusi skrip dinonaktifkan pada sistem, ubah kebijakan eksekusi skrip menjadi `RemoteSigned`.

```
Set-ExecutionPolicy RemoteSigned
```

Mengaktifkan pemrosesan data tepi

Anda dapat menggunakan AWS IoT SiteWise Edge untuk mengumpulkan, memproses, dan memantau data peralatan secara lokal. Anda dapat menggunakan SiteWise Edge sehingga Anda dapat memodelkan data industri Anda dan SiteWise Monitor untuk membuat dasbor bagi staf operasional Anda untuk memvisualisasikan data secara lokal. Anda dapat memproses data secara lokal dan mengirimkannya ke AWS Cloud, atau memprosesnya secara lokal menggunakan API. AWS IoT SiteWise

Dengan AWS IoT SiteWise Edge, Anda dapat memproses data mentah secara lokal dan memilih untuk mengirim hanya data agregat ke AWS Cloud untuk mengoptimalkan penggunaan bandwidth dan biaya penyimpanan cloud Anda.

Note

- AWS IoT SiteWise mempertahankan data tepi Anda di gateway SiteWise Edge Anda hingga 30 hari. Periode penyimpanan data Anda tergantung pada ruang disk yang tersedia di perangkat Anda.
- Jika gateway SiteWise Edge Anda telah terputus dari AWS Cloud selama 30 hari, [Paket Pemrosesan Data](#) secara otomatis dinonaktifkan.

Menyiapkan kemampuan tepi

AWS IoT SiteWise menyediakan paket berikut yang dapat digunakan gateway SiteWise Edge Anda untuk menentukan cara mengumpulkan dan memproses data Anda. Pilih paket untuk mengaktifkan kemampuan tepi untuk gateway SiteWise Edge Anda.

- Paket Pengumpulan Data memungkinkan gateway SiteWise Edge Anda mengumpulkan data dari beberapa server OPC-UA, lalu mengeksport data dari edge ke server. AWS Cloud Ini menjadi aktif setelah Anda menambahkan sumber data ke gateway SiteWise Edge Anda.
- Paket Pemrosesan Data memungkinkan gateway SiteWise Edge Anda untuk memproses data peralatan Anda di tepi. Misalnya, Anda dapat menggunakan model aset untuk menghitung metrik dan transformasi. Untuk informasi selengkapnya tentang model aset dan aset, lihat [Pemodelan aset industri](#).

Note

Paket Pemrosesan Data hanya tersedia di platform x86.

Untuk mengkonfigurasi kemampuan tepi

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih gateway Edge.
3. Pilih gateway SiteWise Edge yang ingin Anda aktifkan kemampuan tepi.
4. Di bagian kemampuan Edge, pilih Edit
5. Di bagian kemampuan Edge, pilih Aktifkan paket pemrosesan data (dikenakan biaya tambahan).
6. (Opsional) Di bagian koneksi Edge LDAP, Anda dapat memberikan grup pengguna di direktori perusahaan Anda akses ke gateway SiteWise Edge ini. Grup pengguna dapat menggunakan kredensi Lightweight Directory Access Protocol (LDAP) untuk mengakses gateway Edge. SiteWise Kemudian mereka dapat menggunakan AWS IoT SiteWise aplikasi AWS OpsHub for, operasi AWS IoT SiteWise API, atau alat lain untuk mengelola gateway SiteWise Edge. Untuk informasi selengkapnya, lihat [Mengelola SiteWise gateway Edge](#).

Note

Anda juga dapat menggunakan kredensi Linux atau Windows untuk mengakses gateway SiteWise Edge. Untuk informasi selengkapnya, lihat [Mengakses gateway SiteWise Edge Anda menggunakan kredensial sistem operasi Linux](#).

- a. Pilih Diaktifkan.
- b. Untuk nama Penyedia, masukkan nama untuk penyedia LDAP Anda.

- c. Untuk nama host atau alamat IP, masukkan nama host atau alamat IP server LDAP Anda.
- d. Untuk Port, masukkan nomor port.
- e. Untuk Base Distinguished Name (DN), masukkan Distinguished Name (DN) untuk basis.

Jenis atribut berikut didukung: commonName (CN), localityName (L), Name (ST), OrganizationName (O), stateOrProvince (OU), countryName (C), streetAddress organizationalUnitName (STREET), domainComponent (DC), dan userid (UID).

- f. Untuk grup Admin DN, masukkan DN.
- g. Untuk grup Pengguna DN, masukkan DN.

7. Pilih Simpan.

Sekarang setelah Anda mengaktifkan kemampuan edge di gateway SiteWise Edge Anda, Anda perlu mengonfigurasi model aset Anda untuk edge. Konfigurasi tepi model aset Anda menentukan tempat properti aset Anda dihitung. Anda dapat menghitung semua properti di edge, atau Anda dapat mengonfigurasi properti model aset Anda secara terpisah. [Properti model aset meliputi metrik, transformasi, dan pengukuran.](#)

Untuk informasi selengkapnya tentang properti aset, lihat [the section called “Mendefinisikan properti data”](#).

Setelah Anda membuat model aset Anda, Anda kemudian dapat mengonfigurasinya untuk edge. Untuk informasi selengkapnya tentang mengonfigurasi model aset Anda untuk edge, lihat [the section called “Membuat model aset \(konsol\)”](#).

Note

Model aset dan dasbor secara otomatis disinkronkan antara gateway SiteWise Edge AWS Cloud dan Edge Anda setiap 10 menit. Anda juga dapat menyinkronkan secara manual dari aplikasi gateway SiteWise Edge lokal.

Memproses data di tepi

Anda harus mengonfigurasi model aset Anda untuk edge sebelum Anda dapat memproses data gateway SiteWise Edge Anda di tepi. Konfigurasi tepi model aset Anda menentukan tempat properti aset Anda dihitung. Anda dapat memilih untuk menghitung semua properti di tepi dan mengirim

hasilnya ke AWS Cloud, atau menyesuaikan tempat menghitung setiap properti aset secara terpisah. Untuk informasi selengkapnya, lihat [Mengaktifkan pemrosesan data tepi](#).

Properti aset meliputi metrik, transformasi, dan pengukuran:

- Metrik adalah data agregat aset selama periode waktu tertentu. Anda dapat menghitung metrik baru dengan menggunakan data metrik yang ada. AWS IoT SiteWise selalu mengirimkan metrik Anda ke AWS Cloud untuk penyimpanan jangka panjang. AWS IoT SiteWise menghitung metrik di AWS Cloud secara default. Anda dapat mengonfigurasi model aset Anda untuk menghitung metrik Anda di tepi. AWS IoT SiteWise mengirimkan hasil yang diproses ke AWS Cloud.
- Transformasi adalah ekspresi matematika yang memetakan titik data properti aset dari satu bentuk ke bentuk lainnya. Transformasi dapat menggunakan metrik sebagai data input dan harus dihitung dan disimpan di lokasi yang sama dengan inputnya. Jika Anda mengonfigurasi input metrik untuk menghitung di tepi, hitung AWS IoT SiteWise juga transformasi terkaitnya di tepi.
- Pengukuran diformat sebagai data mentah yang dikumpulkan dan dikirim perangkat Anda ke AWS Cloud secara default. Anda dapat mengonfigurasi model aset untuk menyimpan data ini di perangkat lokal Anda.

Untuk informasi selengkapnya tentang properti aset, lihat [the section called “Mendefinisikan properti data”](#).

Setelah Anda membuat model aset Anda, Anda kemudian dapat mengonfigurasinya untuk edge. Untuk informasi selengkapnya tentang mengonfigurasi model aset Anda untuk edge, lihat [the section called “Membuat model aset \(konsol\)”](#).

Note

Model aset dan dasbor secara otomatis disinkronkan antara AWS Cloud dan gateway SiteWise Edge Anda setiap 10 menit. Anda juga dapat menyinkronkan secara manual dari file [Mengelola SiteWise gateway Edge](#).

Anda dapat menggunakan AWS IoT SiteWise REST API dan AWS Command Line Interface (AWS CLI) untuk menanyakan gateway SiteWise Edge Anda untuk data di edge. Sebelum Anda menanyakan gateway SiteWise Edge Anda untuk data di tepi, Anda harus memenuhi prasyarat berikut:

- Kredensial Anda harus disetel untuk REST API. Untuk informasi selengkapnya tentang menyetel kredensial, lihat [the section called “Mengelola SiteWise gateway Edge”](#)
- Titik akhir SDK harus mengarah ke alamat IP gateway SiteWise Edge Anda. Anda dapat menemukan informasi selengkapnya di dokumentasi SDK Anda. Misalnya, lihat [Menentukan Titik Akhir Kustom di Panduan AWS SDK for Java 2.x](#) Pengembang.
- Sertifikat gateway SiteWise Edge Anda harus terdaftar. Anda dapat menemukan informasi selengkapnya tentang mendaftarkan sertifikat gateway SiteWise Edge Anda di dokumentasi untuk SDK Anda. Misalnya, lihat [Mendaftarkan Bundel Sertifikat di Node.js](#) di Panduan AWS SDK for Java 2.x Pengembang.

Untuk informasi selengkapnya tentang kueri data dengan AWS IoT SiteWise, lihat [Data kueri dari AWS IoT SiteWise](#).

Mengkonfigurasi komponen AWS IoT SiteWise Publisher

Setelah Anda membuat gateway AWS IoT SiteWise Edge dan menginstal perangkat lunak, siapkan komponen Publisher sehingga gateway SiteWise Edge Anda dapat mengeksport data ke AWS Cloud. Untuk informasi selengkapnya, lihat [AWS IoT SiteWise Penerbit](#) di Panduan AWS IoT Greengrass Version 2 Pengembang.

Untuk mengkonfigurasi penerbit (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih gateway Edge.
3. Pilih gateway SiteWise Edge yang ingin Anda konfigurasi penerbit.
4. Di bagian konfigurasi Publisher, pilih Edit
5. Untuk pesanan Penerbitan, pilih salah satu dari berikut ini:
 - Publikasikan data tertua terlebih dahulu — Gateway SiteWise Edge menerbitkan data paling awal ke cloud terlebih dahulu secara default.
 - Publikasikan data terbaru terlebih dahulu — Gateway SiteWise Edge menerbitkan data terbaru ke cloud terlebih dahulu.
6. (Opsional) Jika Anda tidak ingin gateway SiteWise Edge mengompres data Anda, batalkan pilihan Aktifkan kompresi saat mengunggah data.
7. (Opsional) Jika Anda tidak ingin mempublikasikan data lama, pilih Kecualikan data kedaluwarsa dan lakukan hal berikut:

- Untuk periode Cutoff, masukkan nomor dan pilih unit. Periode batas harus antara lima menit dan tujuh hari. Misalnya, jika batas waktu adalah tiga hari, data yang lebih awal dari tiga hari tidak dipublikasikan ke cloud.
8. (Opsional) Untuk mengatur pengaturan kustom tentang cara data ditangani pada perangkat lokal Anda, pilih Pengaturan penyimpanan lokal dan lakukan hal berikut:
- a. Untuk periode Retensi, masukkan nomor dan pilih unit. Periode retensi harus antara satu menit dan 30 hari, dan lebih besar dari atau sama dengan periode rotasi. Misalnya, jika periode retensi adalah 14 hari, gateway SiteWise Edge menghapus data apa pun di tepi yang lebih awal dari periode batas yang ditentukan setelah disimpan selama 14 hari.
 - b. Untuk periode Rotasi, masukkan nomor dan pilih unit. Periode rotasi harus lebih besar dari satu menit, dan sama dengan, atau kurang dari, periode retensi. Misalnya, jika periode rotasi adalah dua hari, gateway SiteWise Edge mengumpulkan dan menyimpan data yang lebih awal dari periode cutoff ke satu file. Gateway SiteWise Edge juga mentransfer sekumpulan data ke direktori lokal berikut setiap dua hari sekali: `/greengrass/v2/work/aws.iot.SiteWiseEdgePublisher/exports`.
 - c. Untuk kapasitas penyimpanan, masukkan angka yang lebih besar dari atau sama dengan 1. Jika kapasitas penyimpanan 2 GB, gateway SiteWise Edge mulai menghapus data ketika lebih dari 2 GB data disimpan secara lokal.
9. Pilih Simpan.

Mengkonfigurasi sumber data

Setelah menyiapkan gateway AWS IoT SiteWise Edge, Anda dapat mengonfigurasi sumber data sehingga gateway SiteWise Edge Anda dapat menyerap data dari peralatan industri lokal ke AWS IoT SiteWise. Setiap sumber mewakili server lokal, seperti server OPC-UA, yang menghubungkan gateway SiteWise Edge Anda dan mengambil aliran data industri. Untuk informasi selengkapnya tentang menyiapkan gateway SiteWise Edge, lihat [Mengkonfigurasi gateway AWS IoT Greengrass V1 SiteWise Edge](#).

Note

AWS IoT SiteWise restart gateway SiteWise Edge Anda setiap kali Anda menambahkan atau mengedit sumber. Gateway SiteWise Edge Anda tidak akan menelan data saat dimulai ulang. Waktu untuk memulai ulang gateway SiteWise Edge Anda tergantung pada jumlah tag

pada sumber gateway SiteWise Edge Anda. Waktu restart dapat berkisar dari beberapa detik (untuk gateway SiteWise Edge dengan beberapa tag) hingga beberapa menit (untuk gateway SiteWise Edge dengan banyak tag).

Setelah membuat sumber, Anda dapat mengaitkan aliran data Anda dengan properti aset. Untuk informasi selengkapnya tentang cara membuat dan menggunakan aset, lihat [Pemodelan aset industri](#) dan [Memetakan aliran data industri ke properti aset](#).

Anda dapat melihat CloudWatch metrik untuk memverifikasi bahwa sumber data terhubung AWS IoT SiteWise. Untuk informasi selengkapnya, lihat [AWS IoT Greengrass Version 2 metrik gateway](#).

Saat ini, AWS IoT SiteWise mendukung protokol sumber data berikut:

- [OPC-UA](#) — Protokol komunikasi machine-to-machine (M2M) untuk otomasi industri.

Note

SiteWise Edge gateway yang berjalan AWS IoT Greengrass V2 saat ini tidak mendukung sumber Modbus TCP dan Ethernet IP.

Topik

- [Konfigurasi sumber OPC-UA](#)
- [Mengkonfigurasi otentikasi sumber data](#)
- [Memilih tujuan untuk data server sumber Anda](#)

Konfigurasi sumber OPC-UA

Anda dapat menggunakan AWS IoT SiteWise konsol atau kemampuan gateway SiteWise Edge untuk menentukan dan menambahkan sumber OPC-UA ke gateway SiteWise Edge Anda untuk mewakili server OPC-UA lokal.

Topik

- [Konfigurasi sumber OPC-UA \(konsol\)](#)
- [Konfigurasi sumber OPC-UA \(CLI\)](#)

- [Mengaktifkan server sumber OPC-UA Anda untuk mempercayai gateway Edge SiteWise](#)
- [Filter rentang konsumsi data dengan OPC-UA](#)
- [Menggunakan filter node OPC-UA](#)

Konfigurasi sumber OPC-UA (konsol)

Untuk mengonfigurasi sumber OPC-UA menggunakan konsol AWS IoT SiteWise

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Gateway.
3. Pilih gateway SiteWise Edge untuk menambahkan sumber OPC-UA.
4. Pilih Tambahkan sumber data.
5. Masukkan nama untuk sumbernya.
6. (Opsional) Masukkan awalan aliran Data. Gateway SiteWise Edge menambahkan awalan ini ke semua aliran data dari sumber ini. Gunakan awalan aliran data untuk membedakan antara aliran data yang memiliki nama yang sama dari sumber yang berbeda. Setiap aliran data harus memiliki nama unik dalam akun Anda.
7. Masukkan titik akhir lokal dari server sumber data. Titik akhir dapat berupa alamat IP atau nama host. Anda juga dapat menambahkan nomor port ke titik akhir lokal. Misalnya, titik akhir lokal Anda mungkin terlihat seperti ini: **opc.tcp://203.0.113.0:49320**

Note

Jika gateway SiteWise Edge Anda memiliki Deployment type perangkat Siemens Industrial Edge - baru dan Anda ingin menyerap data dari aplikasi Edge OPC UA Server yang berjalan pada Perangkat Tepi Industri Siemens yang sama dengan aplikasi Edge, masukkan. AWS IoT SiteWise **opc.tcp://ie-opcua:48010**

8. (Opsional) Untuk ID Node untuk pemilihan, tambahkan filter node untuk membatasi aliran data mana yang dicerna ke file. AWS Cloud Secara default, gateway SiteWise Edge menggunakan simpul akar server untuk menelan semua aliran data. Untuk menentukan filter node, Anda dapat menggunakan ID node dan karakter * dan ** wildcard.
9. Untuk Destinasi, pilih tujuan untuk sumber data:
 - AWS IoT SiteWise real-time — Pilih ini untuk mengirim data langsung ke AWS IoT SiteWise penyimpanan. Menelan dan memantau data secara real-time, dan memproses data di tepi.

- AWS IoT SiteWise Buffer menggunakan Amazon S3 - Kirim data dalam format parquet ke Amazon S3 dan kemudian impor ke penyimpanan. AWS IoT SiteWise Pilih opsi ini untuk menelan data dalam batch, dan menyimpan data historis dengan cara yang hemat biaya. Anda dapat mengonfigurasi lokasi bucket Amazon S3 pilihan Anda, dan frekuensi di mana Anda ingin data diunggah ke Amazon S3. Anda juga dapat memilih apa yang harus dilakukan dengan data setelah tertelan. AWS IoT SiteWise Anda dapat memilih untuk memiliki data yang tersedia di keduanya SiteWise dan Amazon S3 atau Anda dapat memilih untuk menghapusnya secara otomatis dari Amazon S3.
- Bucket Amazon S3 adalah mekanisme pementasan dan buffering dan mendukung file dalam format parquet.
- Jika Anda memilih kotak centang Impor data ke AWS IoT SiteWise penyimpanan, data diunggah ke Amazon S3 terlebih dahulu, lalu AWS IoT SiteWise ke penyimpanan.
 - Jika Anda memilih kotak centang Hapus data dari Amazon S3, data dihapus dari Amazon S3, setelah diimpor ke penyimpanan. SiteWise
 - Jika Anda menghapus kotak centang Hapus data dari Amazon S3, data disimpan baik di Amazon S3, dan di penyimpanan. SiteWise
- Jika Anda menghapus kotak centang Impor data ke AWS IoT SiteWise penyimpanan, data hanya disimpan di Amazon S3. Itu tidak diimpor ke SiteWise penyimpanan.

Kunjungi [Mengelola penyimpanan data](#) untuk detail tentang berbagai opsi penyimpanan yang AWS IoT SiteWise disediakan. Untuk mempelajari lebih lanjut tentang opsi harga, lihat [AWS IoT SiteWise harga](#).

- AWS IoT Greengrass pengelola aliran — Gunakan pengelola AWS IoT Greengrass aliran untuk mengirim data ke AWS Cloud tujuan berikut: saluran masuk AWS IoT Analytics, streaming di Amazon Kinesis Data Streams, properti aset di, atau objek AWS IoT SiteWise di Amazon Simple Storage Service (Amazon S3). Untuk informasi selengkapnya, lihat [Mengelola aliran data di Panduan AWS IoT Greengrass Version 2 Pengembang AWS IoT Greengrass Inti](#).

Masukkan nama untuk AWS IoT Greengrass streaming.

10. Saat mengkonfigurasi sumber data, ID Node untuk seleksi digunakan untuk menentukan tujuan aliran data.
 - Jika data yang sama dipublikasikan ke AWS IoT SiteWise real-time dan AWS IoT SiteWise Buffered menggunakan Amazon S3, Anda harus menambahkan dua sumber data yang dipublikasikan ke kedua tujuan.

- Untuk membagi data sehingga sebagian dipublikasikan ke AWS IoT SiteWise real-time, dan bagian lainnya ke AWS IoT SiteWise Buffered menggunakan Amazon S3, Anda harus memfilter alias data berikut:

```
/Alias01/Data1  
/Alias02/Data1  
/Alias03/Data1  
/Alias03/Data2
```

Misalnya, Anda dapat menambahkan sumber data yang menunjuk ke filter `/**/Data1` node, ke AWS IoT SiteWise real-time, dan sumber data lain yang menunjuk ke `/**/Data2` AWS IoT SiteWise buffer menggunakan Amazon S3

11. Di panel konfigurasi lanjutan, lakukan hal berikut:

- a. Pilih mode keamanan Pesan untuk koneksi dan data dalam perjalanan antara server sumber dan gateway SiteWise Edge Anda. Bidang ini adalah kombinasi dari kebijakan keamanan OPC-UA dan mode keamanan pesan. Pilih kebijakan keamanan dan mode keamanan pesan yang sama yang Anda tentukan untuk server OPC-UA Anda.
- b. Jika sumber Anda memerlukan otentikasi, pilih AWS Secrets Manager rahasia dari daftar konfigurasi Otentikasi. Gateway SiteWise Edge menggunakan kredensial otentikasi dalam rahasia ini ketika terhubung ke sumber data ini. Anda harus melampirkan rahasia ke AWS IoT Greengrass komponen gateway SiteWise Edge Anda untuk menggunakannya untuk otentikasi sumber data. Untuk informasi selengkapnya, lihat [the section called “Mengkonfigurasi otentikasi sumber data”](#).

 Tip

Server data Anda mungkin memiliki opsi bernama Izinkan login anonim. Jika opsi ini adalah Ya, maka sumber Anda tidak memerlukan otentikasi.

- c. Untuk grup Properti, pilih Tambahkan grup baru.
- d. Masukkan Nama untuk grup properti.
- e. Untuk Properti:
 1. (Opsional) Untuk jalur Node, tambahkan filter node OPC-UA untuk membatasi jalur OPC-UA mana yang diunggah. AWS IoT SiteWise Anda dapat menggunakan filter node untuk mengurangi waktu startup dan penggunaan CPU gateway SiteWise Edge Anda dengan

hanya menyertakan jalur ke data yang Anda modelkan AWS IoT SiteWise. Secara default, gateway SiteWise Edge mengunggah semua jalur OPC-UA kecuali yang dimulai dengan `/Server/`. Untuk menentukan filter node OPC-UA, Anda dapat menggunakan jalur node `*` dan `**` karakter wildcard. Untuk informasi selengkapnya, lihat [Menggunakan filter node OPC-UA](#).

f. Untuk pengaturan Grup, lakukan hal berikut:

1. (Opsional) Untuk pengaturan kualitas Data, pilih jenis kualitas data yang ingin Anda konsumsi oleh AWS IoT SiteWise Kolektor.
2. (Opsional) Untuk pengaturan mode Pindai, konfigurasi properti langganan standar berikut:
 - Untuk mode Pindai, pilih mode yang AWS IoT SiteWise ingin Anda gunakan untuk mengumpulkan data Anda. Untuk informasi selengkapnya tentang mode pemindaian, lihat [the section called “Memfilter rentang konsumsi data dengan OPC-UA”](#).
 - [Inisiasi perubahan data](#) —: Anda dapat menentukan kondisi yang memulai peringatan perubahan data.
 - [Ukuran antrian langganan](#) —: Kedalaman antrian pada server OPC-UA untuk metrik tertentu di mana pemberitahuan untuk item yang dipantau diantrian.
 - [Interval penerbitan langganan](#) —: Interval (dalam milidetik) siklus penerbitan yang ditentukan saat langganan dibuat.
 - Interval snapshot —: Anda dapat mengonfigurasi pengaturan batas waktu frekuensi snapshot untuk memastikan bahwa AWS IoT SiteWise Edge menyerap aliran data yang stabil.
 - Untuk tingkat Pemindaian, perbarui tarif yang Anda inginkan gateway SiteWise Edge untuk membaca register Anda. AWS IoT SiteWise secara otomatis menghitung tingkat pemindaian minimum yang diijinkan untuk gateway SiteWise Edge Anda.
3. (Opsional) Konfigurasi jenis Deadband untuk sumber Anda. Ini mengontrol data apa yang dikirim sumber Anda ke Anda AWS IoT SiteWise, dan data apa yang dibuang. Untuk informasi lebih lanjut tentang pengaturan deadband, lihat [the section called “Memfilter rentang konsumsi data dengan OPC-UA”](#).

g. Pilih Tambahkan.

12. Pilih Berikutnya.

Konfigurasi sumber OPC-UA (CLI)

Anda dapat menentukan sumber data OPC-UA untuk gateway SiteWise Edge menggunakan file. AWS CLI Untuk melakukan ini, buat file JSON konfigurasi kemampuan OPC-UA dan gunakan [update-gateway-capability-configuration](#) perintah untuk memperbarui konfigurasi gateway SiteWise Edge. Anda harus menentukan semua sumber OPC-UA Anda dalam satu konfigurasi kemampuan.

Untuk informasi lebih lanjut tentang mendefinisikan sumber dengan AWS Command Line Interface, lihat [the section called "Mengkonfigurasi sumber data \(\)AWS CLI"](#).

Kemampuan ini memiliki versi berikut.

Versi	Namespace
1	ioticsitewise:opcuacollector:1

Sintaksis Permintaan

```
{
  "sources": [
    {
      "name": "string",
      "endpoint": {
        "certificateTrust": {
          "type": "string"
          "certificateBody": "string"
          "certificateChain": "string"
        },
        "endpointUri": "string",
        "securityPolicy": "string",
        "messageSecurityMode": "string",
        "identityProvider": {
          "type": "string",
          "usernameSecretArn": "string"
        },
        "nodeFilterRules": [
          {
            "action": "string",
            "definition": {
              "type": "string",
              "rootPath": "string"
            }
          }
        ]
      }
    }
  ]
}
```

```
    }
  }
]
},
"measurementDataStreamPrefix": "string"
"propertyGroups": [
{
  "name": "string",
  "deadband": {
    "type": "string",
    "value": string,
    "eguMin": string,
    "eguMax": string,
    "timeoutMilliseconds": string
  },
  "scanMode": {
    "type": "string",
    "rate": string
  },
  "nodeFilterRuleDefinitions": [
    {
      "type": "string",
      "rootPath": "string"
    }
  ]
}
}
]
}
```

Isi permintaan

sumber

Daftar struktur definisi sumber OPC-UA yang masing-masing berisi informasi berikut:

name

Nama yang unik dan ramah untuk sumbernya.

titik akhir

Struktur endpoint yang berisi informasi berikut:

CertificateTrust

Struktur kebijakan kepercayaan sertifikat yang berisi informasi berikut:

tipe

Mode kepercayaan sertifikat untuk sumbernya. Pilih salah satu cara berikut:

- **TrustAny**— Gateway SiteWise Edge mempercayai sertifikat apa pun saat terhubung ke sumber OPC-UA.
- **X509**— Gateway SiteWise Edge mempercayai sertifikat X.509 saat terhubung ke sumber OPC-UA. Jika Anda memilih opsi ini, Anda harus menentukan `certificateBodycertificateTrust`. Anda juga dapat menentukan `certificateChain` dalam `certificateTrust`.

CertificateBody

(Opsional) Badan sertifikat X.509.

Bidang ini diperlukan jika Anda memilih X509 untuk type masuk `certificateTrust`.

CertificateChain

(Opsional) Rantai kepercayaan untuk sertifikat X.509.

Bidang ini hanya digunakan jika Anda memilih X509 untuk type masuk `certificateTrust`.

EndpointTuri


Titik akhir lokal dari sumber OPC-UA. Misalnya, titik akhir lokal Anda mungkin terlihat seperti `opc.tcp://203.0.113.0:49320`.

Kebijakan Keamanan

Kebijakan keamanan yang digunakan sehingga Anda dapat mengamankan pesan yang dibaca dari sumber OPC-UA. Pilih salah satu cara berikut:

- **NONE**— Gateway SiteWise Edge tidak mengamankan pesan dari sumber OPC-UA. Kami menyarankan Anda memilih kebijakan keamanan yang berbeda. Jika Anda memilih opsi ini, Anda juga harus memilih **NONE** untuk `messageSecurityMode`.
- **BASIC256_SHA256**- Kebijakan Basic256Sha256 keamanan.
- **AES128_SHA256_RSA0AEP**- Kebijakan Aes128_Sha256_Rsa0aep keamanan.
- **AES256_SHA256_RSAPSS**- Kebijakan Aes256_Sha256_RsaPss keamanan.

- BASIC128_RSA15— (Usang) Kebijakan Basic128Rsa15 keamanan tidak digunakan lagi dalam spesifikasi OPC-UA karena tidak lagi dianggap aman. Kami menyarankan Anda memilih kebijakan keamanan yang berbeda. Untuk informasi lebih lanjut, lihat [Basic128Rsa15](#).
- BASIC256— (Usang) Kebijakan Basic256 keamanan tidak digunakan lagi dalam spesifikasi OPC-UA karena tidak lagi dianggap aman. Kami menyarankan Anda memilih kebijakan keamanan yang berbeda. Untuk informasi lebih lanjut, lihat [Basic256](#).


 Important

Jika Anda memilih kebijakan keamanan selain NONE, Anda harus memilih SIGN atau SIGN_AND_ENCRYPT untuk `messageSecurityMode`. Anda juga harus mengonfigurasi server sumber Anda untuk mempercayai gateway SiteWise Edge. Untuk informasi selengkapnya, lihat [Mengaktifkan server sumber OPC-UA Anda untuk mempercayai gateway Edge SiteWise](#) .

`messageSecurityMode`

Mode keamanan pesan yang digunakan untuk mengamankan koneksi ke sumber OPC-UA. Pilih salah satu cara berikut:

- NONE— Gateway SiteWise Edge tidak mengamankan koneksi ke sumber OPC-UA. Kami menyarankan Anda memilih mode keamanan pesan yang berbeda. Jika Anda memilih opsi ini, Anda juga harus memilih NONE untuk `securityPolicy`.
- SIGN— Data dalam perjalanan antara gateway SiteWise Edge dan sumber OPC-UA ditandatangani tetapi tidak dienkripsi.
- SIGN_AND_ENCRYPT— Data dalam perjalanan antara gateway dan sumber OPC-UA ditandatangani dan dienkripsi.

 Important

Jika Anda memilih mode keamanan pesan selain NONE, Anda harus memilih `securityPolicy` selain NONE. Anda juga harus mengonfigurasi server sumber Anda untuk mempercayai gateway SiteWise Edge. Untuk informasi selengkapnya, lihat [Mengaktifkan server sumber OPC-UA Anda untuk mempercayai gateway Edge SiteWise](#) .

Penyedia Identitas

Struktur penyedia identitas yang berisi informasi berikut:

tipe

Jenis kredensial otentikasi yang diperlukan oleh sumber. Pilih salah satu cara berikut:

- **Anonymous**— Sumber tidak memerlukan otentikasi untuk terhubung.
- **Username**— Sumber membutuhkan nama pengguna dan kata sandi untuk terhubung. Jika Anda memilih opsi ini, Anda harus menentukan `usernameSecretArnidentityProvider`.

usernameSecretArn

(Opsional) ARN rahasia. AWS Secrets Manager Gateway SiteWise Edge menggunakan kredensial otentikasi dalam rahasia ini ketika terhubung ke sumber ini. Anda harus melampirkan rahasia ke SiteWise konektor IoT gateway SiteWise Edge Anda untuk menggunakannya untuk otentikasi sumber. Untuk informasi selengkapnya, lihat [Mengkonfigurasi otentikasi sumber data](#).

Bidang ini diperlukan jika Anda memilih Username untuk type `masukidentityProvider`.

nodeFilterRules

Daftar struktur aturan filter node yang menentukan jalur aliran data OPC-UA untuk dikirim ke Cloud. AWS Anda dapat menggunakan filter node untuk mengurangi waktu startup dan penggunaan CPU gateway SiteWise Edge Anda dengan hanya menyertakan jalur ke data yang Anda modelkan AWS IoT SiteWise. Secara default, gateway SiteWise Edge mengunggah semua jalur OPC-UA kecuali yang dimulai dengan `/Server/` Untuk menentukan filter node OPC-UA, Anda dapat menggunakan jalur node `*` dan `**` karakter wildcard. Untuk informasi selengkapnya, lihat [Menggunakan filter node OPC-UA](#).

Setiap struktur dalam daftar harus berisi informasi berikut:

aksi

Tindakan untuk aturan filter node ini. Anda dapat memilih opsi berikut:

- **INCLUDE**— Gateway SiteWise Edge hanya mencakup aliran data yang cocok dengan aturan ini.

definisi

Struktur aturan filter node yang berisi informasi berikut:

tipe

Jenis jalur filter node untuk aturan ini. Anda dapat memilih opsi berikut:

- `OpcUaRootPath`— Gateway SiteWise Edge mengevaluasi jalur filter node ini terhadap root hierarki jalur OPC-UA.

RootPath

Jalur filter node untuk mengevaluasi terhadap akar hierarki jalur OPC-UA. Jalan ini harus dimulai dengan `/`.

measurementDataStreamAwalan

Sebuah string untuk ditambahkan ke semua aliran data dari sumber. Gateway SiteWise Edge menambahkan awalan ini ke semua aliran data dari sumber ini. Gunakan awalan aliran data untuk membedakan antara aliran data yang memiliki nama yang sama dari sumber yang berbeda. Setiap aliran data harus memiliki nama unik dalam akun Anda.

PropertyGroups

(Opsional) Daftar grup properti yang menentukan deadband dan scanMode diminta oleh protokol.

name

Nama grup properti. Ini harus menjadi pengidentifikasi unik.

deadband

deadbandStruktur yang berisi informasi berikut:

tipe

Jenis deadband yang didukung. Nilai yang diterima adalah ABSOLUTE dan PERCENT.

value

Nilai deadband. `typeKapanABSOLUTE`, nilai ini adalah ganda tanpa unit.

`typeKapanPERCENT`, nilai ini adalah ganda antara 1 dan 100.

eGUMin

(Opsional) Unit teknik minimum saat menggunakan PERCENT deadband. Anda mengatur ini jika server OPC-UA tidak memiliki unit teknik yang dikonfigurasi.

eGUMAX

(Opsional) Unit teknik maksimum saat menggunakan PERCENT deadband. Anda mengatur ini jika server OPC-UA tidak memiliki unit teknik yang dikonfigurasi.

TimeoutMilliseconds

Durasi dalam milidetik sebelum batas waktu. Minimal adalah 100.

ScanMode

scanModeStruktur yang berisi informasi berikut:

tipe

Jenis yang didukung dari scanMode. Nilai yang diterima adalah POLL dan EXCEPTION.

tingkat

Interval pengambilan sampel untuk mode pemindaian.

nodeFilterRuleDefinisi

(Opsional) Daftar jalur simpul untuk disertakan dalam grup properti. Grup properti tidak dapat tumpang tindih. Jika Anda tidak menentukan nilai untuk bidang ini, grup berisi semua jalur di bawah root, dan Anda tidak dapat membuat grup properti tambahan.

nodeFilterRuleDefinitionsStruktur berisi informasi berikut:

tipe

OpcUaRootPath adalah satu-satunya tipe yang didukung. Ini menentukan bahwa nilai rootPath adalah jalur relatif terhadap root ruang penjelajahan OPC-UA.

RootPath

Daftar yang dibatasi koma yang menentukan jalur (relatif terhadap root) untuk disertakan dalam grup properti.

Contoh konfigurasi kemampuan

Contoh berikut mendefinisikan konfigurasi kemampuan gateway OPC-UA SiteWise Edge dari payload yang disimpan dalam file JSON.

```
aws iotsitewise update-gateway-capability-configuration \  
--capability-namespace "iotsitewise:opcuacollector:1" \  
--capability-configuration file://opc-ua-configuration.json
```

Example : Konfigurasi sumber OPC-UA

`opc-ua-configuration.json`File berikut mendefinisikan konfigurasi sumber OPC-UA dasar dan tidak aman.

```
{
  "sources": [
    {
      "name": "Wind Farm #1",
      "endpoint": {
        "certificateTrust": {
          "type": "TrustAny"
        },
        "endpointUri": "opc.tcp://203.0.113.0:49320",
        "securityPolicy": "NONE",
        "messageSecurityMode": "NONE",
        "identityProvider": {
          "type": "Anonymous"
        },
        "nodeFilterRules": []
      },
      "measurementDataStreamPrefix": ""
    }
  ]
}
```

Example : Konfigurasi sumber OPC-UA dengan grup properti yang ditentukan

`opc-ua-configuration.json`File berikut mendefinisikan konfigurasi sumber OPC-UA dasar dan tidak aman dengan grup properti yang ditentukan.

```
{
  "sources": [
    {
      "name": "source1",
      "endpoint": {
        "certificateTrust": {
          "type": "TrustAny"
        },
        "endpointUri": "opc.tcp://10.0.0.9:49320",
        "securityPolicy": "NONE",
        "messageSecurityMode": "NONE",
        "identityProvider": {
```

```

        "type": "Anonymous"
    },
    "nodeFilterRules": [
        {
            "action": "INCLUDE",
            "definition": {
                "type": "OpcUaRootPath",
                "rootPath": "/Utilities/Tank"
            }
        }
    ]
},
"measurementDataStreamPrefix": "propertyGroups",
"propertyGroups": [
    {
        "name": "Deadband_Abs_5",
        "nodeFilterRuleDefinitions": [
            {
                "type": "OpcUaRootPath",
                "rootPath": "/Utilities/Tank/Temperature/TT-001"
            },
            {
                "type": "OpcUaRootPath",
                "rootPath": "/Utilities/Tank/Temperature/TT-002"
            }
        ],
        "deadband": {
            "type": "ABSOLUTE",
            "value": 5.0,
            "timeoutMilliseconds": 120000
        }
    },
    {
        "name": "Polling_10s",
        "nodeFilterRuleDefinitions": [
            {
                "type": "OpcUaRootPath",
                "rootPath": "/Utilities/Tank/Pressure/PT-001"
            }
        ],
        "scanMode": {
            "type": "POLL",
            "rate": 10000
        }
    }
]

```

```

    }
  },
  {
    "name": "Percent_Deadband_Timeout_90s",
    "nodeFilterRuleDefinitions": [
      {
        "type": "OpcUaRootPath",
        "rootPath": "/Utilities/Tank/Flow/FT-*"
      }
    ],
    "deadband": {
      "type": "PERCENT",
      "value": 5.0,
      "eguMin": -100,
      "eguMax": 100,
      "timeoutMilliseconds": 90000
    }
  }
]
}
}

```

Example : Konfigurasi sumber OPC-UA dengan properti

Contoh JSON berikut untuk `opc-ua-configuration.json` mendefinisikan konfigurasi sumber OPC-UA dengan properti berikut:

- Mempercayai sertifikat apa pun.
- Menggunakan kebijakan BASIC256 keamanan untuk mengamankan pesan.
- Menggunakan SIGN_AND_ENCRYPT mode untuk mengamankan koneksi.
- Menggunakan kredensial otentikasi yang disimpan dalam rahasia Secrets Manager.
- Memfilter aliran data kecuali yang jalurnya dimulai dengan `/WindFarm/2/WindTurbine/`.
- Menambahkan `/Washington` ke awal setiap jalur aliran data untuk membedakan antara "Wind Farm #2" dan "Wind Farm #2" di area lain.

```

{
  "sources": [
    {
      "name": "Wind Farm #2",

```

```

"endpoint": {
  "certificateTrust": {
    "type": "TrustAny"
  },
  "endpointUri": "opc.tcp://203.0.113.1:49320",
  "securityPolicy": "BASIC256",
  "messageSecurityMode": "SIGN_AND_ENCRYPT",
  "identityProvider": {
    "type": "Username",
    "usernameSecretArn":
"arn:aws:secretsmanager:region:123456789012:secret:greenrass-windfarm2-auth-1ABCDE"
  },
  "nodeFilterRules": [
    {
      "action": "INCLUDE",
      "definition": {
        "type": "OpcUaRootPath",
        "rootPath": "/WindFarm/2/WindTurbine/"
      }
    }
  ]
},
"measurementDataStreamPrefix": "/Washington"
}
]
}

```

Example

Contoh JSON berikut untuk `opc-ua-configuration.json` mendefinisikan konfigurasi sumber OPC-UA dengan properti berikut:

- Mempercayai sertifikat X.509 yang diberikan.
- Menggunakan kebijakan BASIC256 keamanan untuk mengamankan pesan.
- Menggunakan SIGN_AND_ENCRYPT mode untuk mengamankan koneksi.

```

{
  "sources": [
    {
      "name": "Wind Farm #3",
      "endpoint": {

```

```

    "certificateTrust": {
      "type": "X509",
      "certificateBody": "-----BEGIN CERTIFICATE-----
MIICiTCCAfICCD6m7oRw0uX0jANBgkqhkiG9w
0BAQUFADCBiDELMAKGA1UEBhMVCVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZ
WF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAstC01BTSBDb25zb2x1MRIw
EAYDVQQDEw1UZXR0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5
jb20wHhcNMTEwNDI1MjA0NTIxWhcNMTEwNDI1MjA0NTIxWjCBiDELMAKGA1UEBh
MVCVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBb
WF6b24xFDASBgNVBAstC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxh
ZAdBgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wgZ8wDQYJKoZIhvcNAQE
BBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ21uUSfwfEvySwTc2XADZ4nB+BLyGVI
k60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/MbQ
ITx0USQv7c7ugFFDzQGBzZswY6786m86gpEibb30hjZnzcVQAaRHhd1QWIMm2nr
AgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4nUhVVxYUntneD9+h8Mg9q6q+auN
KyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJI1J00zbhNYS5f6Guo
EDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStbNYiytVbZPQUQ5Yaxu2jXnimvw
3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
      "certificateChain": "-----BEGIN CERTIFICATE-----
MIICiTCCAfICCD6m7oRw0uX0jANBgkqhkiG9w
0BAQUFADCBiDELMAKGA1UEBhMVCVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZ
WF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAstC01BTSBDb25zb2x1MRIw
EAYDVQQDEw1UZXR0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5
jb20wHhcNMTEwNDI1MjA0NTIxWhcNMTEwNDI1MjA0NTIxWjCBiDELMAKGA1UEBh
MVCVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBb
WF6b24xFDASBgNVBAstC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxh
ZAdBgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wgZ8wDQYJKoZIhvcNAQE
BBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ21uUSfwfEvySwTc2XADZ4nB+BLyGVI
k60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/MbQ
ITx0USQv7c7ugFFDzQGBzZswY6786m86gpEibb30hjZnzcVQAaRHhd1QWIMm2nr
AgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4nUhVVxYUntneD9+h8Mg9q6q+auN
KyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJI1J00zbhNYS5f6Guo
EDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStbNYiytVbZPQUQ5Yaxu2jXnimvw
3rrszlaEXAMPLE=
-----END CERTIFICATE-----"
    },
    "endpointUri": "opc.tcp://203.0.113.2:49320",
    "securityPolicy": "BASIC256",
    "messageSecurityMode": "SIGN_AND_ENCRYPT",
    "identityProvider": {
      "type": "Anonymous"
    },
    "nodeFilterRules": []
  }

```



```
    },  
    "measurementDataStreamPrefix": ""  
  }  
]  
}
```

Mengaktifkan server sumber OPC-UA Anda untuk mempercayai gateway Edge SiteWise

Jika Anda memilih `messageSecurityMode` selain `None` saat mengonfigurasi sumber OPC-UA Anda, Anda harus mengaktifkan server sumber Anda untuk mempercayai gateway Edge. AWS IoT SiteWise Gateway SiteWise Edge menghasilkan sertifikat yang mungkin diperlukan oleh server sumber Anda. Prosesnya bervariasi tergantung pada server sumber Anda. Untuk informasi selengkapnya, lihat dokumentasi untuk server Anda.

Prosedur berikut menguraikan langkah-langkah dasar.

Untuk mengaktifkan server OPC-UA untuk mempercayai gateway SiteWise Edge

1. Buka antarmuka untuk mengonfigurasi server OPC-UA Anda.
2. Masukkan nama pengguna dan kata sandi untuk administrator server OPC-UA.
3. Temukan Klien Tepercaya di antarmuka, lalu pilih AWS IoT SiteWise Gateway Client.
4. Pilih Kepercayaan.

Mengekspor sertifikat klien OPC-UA

Beberapa server OPC-UA memerlukan akses ke file sertifikat klien OPC-UA untuk mempercayai gateway Edge. SiteWise Jika ini berlaku untuk server OPC-UA Anda, Anda dapat menggunakan prosedur berikut untuk mengekspor sertifikat klien OPC-UA dari gateway Edge. SiteWise Kemudian, Anda dapat mengimpor sertifikat di server OPC-UA Anda.

Untuk mengekspor file sertifikat klien OPC-UA untuk sumber

1. Jalankan perintah berikut untuk mengubah ke direktori yang berisi file sertifikat. *Ganti `sitewise-work` dengan jalur penyimpanan lokal untuk `aws.iot.SiteWiseEdgeCollectorOpcua` Greengrass folder kerja dan `ganti nama sumber dengan nama` sumber data.*

Secara default, folder kerja Greengrass adalah `/greengrass/v2/work/aws.iot.SiteWiseEdgeCollectorOpcua` di Linux dan `C: /greengrass/v2/work/aws.iot.SiteWiseEdgeCollectorOpcua` di Windows.

```
cd /sitewise-work/source-name/opcua-certificate-store
```

2. Sertifikat klien OPC-UA gateway SiteWise Edge untuk sumber ini ada di file `aws-iot-opcua-client.pfx`

Jalankan perintah berikut untuk mengeksport sertifikat ke `.pem` file bernama `aws-iot-opcua-client-certificate.pem`.

```
keytool -exportcert -v -alias aws-iot-opcua-client -keystore aws-iot-opcua-client.pfx -storepass amazon -storetype PKCS12 -rfc > aws-iot-opcua-client-certificate.pem
```

3. Transfer file sertifikat, `aws-iot-opcua-client-certificate.pem`, dari gateway SiteWise Edge ke server OPC-UA.

Untuk melakukannya, Anda dapat menggunakan perangkat lunak umum seperti `scp` program untuk mentransfer file menggunakan protokol SSH. Untuk informasi selengkapnya, lihat [Salinan aman](#) di Wikipedia.

Note

Jika gateway SiteWise Edge Anda berjalan di Amazon Elastic Compute Cloud (Amazon EC2) dan Anda menghubungkannya untuk pertama kalinya, Anda harus mengonfigurasi prasyarat untuk terhubung. Untuk informasi lebih lanjut, lihat [Connect ke Instans Linux Anda](#) dalam Panduan Pengguna Amazon EC2 untuk Instans Linux.

4. Impor file sertifikat, `aws-iot-opcua-client-certificate.pem`, di server OPC-UA untuk mempercayai gateway SiteWise Edge. Langkah-langkah dapat bervariasi tergantung pada server sumber yang Anda gunakan. Konsultasikan dokumentasi untuk server.

Filter rentang konsumsi data dengan OPC-UA

Anda dapat mengontrol cara Anda menelan data dengan sumber OPC-UA dengan menggunakan mode pemindaian dan rentang deadband. Fitur-fitur ini memungkinkan Anda mengontrol jenis data

apa yang akan dikonsumsi, dan bagaimana serta kapan server dan gateway SiteWise Edge Anda bertukar informasi ini.

Kontrol frekuensi pengumpulan data dengan mode Pindai

Anda dapat mengonfigurasi mode pemindaian OPC-UA untuk mengontrol cara Anda mengumpulkan data dari sumber OPC-UA Anda. Anda dapat memilih mode berlangganan atau polling.

- Mode berlangganan — Sumber OPC-UA mengumpulkan data untuk dikirim ke gateway SiteWise Edge Anda pada frekuensi yang ditentukan oleh tingkat pemindaian Anda. Server hanya mengirim data ketika nilai telah berubah, jadi ini adalah frekuensi maksimum gateway SiteWise Edge Anda menerima data.
- Mode polling - Gateway SiteWise Edge Anda akan melakukan polling sumber OPC-UA pada frekuensi yang ditetapkan yang ditentukan oleh tingkat pemindaian Anda. Server mengirimkan data terlepas dari apakah nilainya telah berubah, sehingga gateway SiteWise Edge Anda selalu menerima data pada interval ini.

Note

Opsi mode polling mengesampingkan pengaturan deadband Anda untuk sumber ini.

Filter konsumsi data OPC-UA dengan rentang deadband

Anda dapat menerapkan deadband ke grup properti sumber OPC-UA Anda untuk memfilter dan membuang data tertentu alih-alih mengirimkannya ke Cloud. AWS Deadband menentukan jendela fluktuasi yang diharapkan dalam nilai data yang masuk dari sumber OPC-UA Anda. Jika nilai termasuk dalam jendela ini, server OPC-UA Anda tidak akan mengirimkannya ke Cloud. AWS Anda dapat menggunakan pemfilteran deadband untuk mengurangi jumlah data yang Anda proses dan kirim ke Cloud. AWS Untuk mempelajari cara menyiapkan sumber OPC-UA untuk gateway SiteWise Edge Anda, lihat [the section called “Mengkonfigurasi sumber data”](#)

Note

Server Anda menghapus semua data yang ada di dalam jendela yang ditentukan oleh deadband Anda. Anda tidak dapat memulihkan data yang dibuang ini.

Jenis-jenis deadband

Anda dapat menentukan dua jenis deadband untuk grup properti server OPC-UA Anda. Ini memungkinkan Anda memilih berapa banyak data yang dikirim ke AWS Cloud, dan berapa banyak yang dibuang.

- **Persentase** - Anda menentukan jendela menggunakan persentase fluktuasi yang diharapkan dalam nilai pengukuran. Server menghitung jendela yang tepat dari persentase ini, dan mengirimkan data ke AWS Cloud yang melebihi jatuh di luar jendela. Misalnya, menentukan nilai deadband 2% pada sensor dengan kisaran dari -100 derajat Fahrenheit hingga +100 derajat Fahrenheit memberi tahu server untuk mengirim data ke Cloud ketika nilainya berubah 4 derajat Fahrenheit atau lebih. AWS

Note

Anda dapat secara opsional menentukan nilai deadband minimum dan maksimum untuk jendela ini jika server sumber Anda tidak menentukan unit teknik. Jika rentang unit teknik tidak disediakan, server OPC-UA default ke rentang penuh tipe data pengukuran.

- **Absolute** — Anda menentukan jendela menggunakan unit yang tepat. Misalnya, menentukan nilai deadband 2 pada sensor memberitahu server untuk mengirim data ke AWS Cloud ketika nilainya berubah setidaknya 2 unit. Anda dapat menggunakan deadbanding absolut untuk lingkungan dinamis di mana fluktuasi diharapkan secara teratur selama operasi normal.

Batas waktu deadband

Anda dapat secara opsional mengonfigurasi pengaturan batas waktu deadband. Setelah batas waktu ini, server OPC-UA mengirimkan nilai pengukuran saat ini meskipun berada dalam fluktuasi deadband yang diharapkan. Anda dapat menggunakan pengaturan batas waktu untuk memastikan bahwa AWS IoT SiteWise menelan aliran data yang stabil setiap saat, bahkan ketika nilai tidak melebihi jendela deadband yang ditentukan.

Menggunakan filter node OPC-UA

Saat Anda menentukan sumber data OPC-UA untuk gateway SiteWise Edge, Anda dapat menentukan filter node. Filter node memungkinkan Anda membatasi jalur aliran data mana yang dikirim gateway SiteWise Edge ke cloud. Anda dapat menggunakan filter node untuk mengurangi waktu startup dan penggunaan CPU gateway SiteWise Edge Anda dengan hanya menyertakan jalur ke data yang Anda modelkan AWS IoT SiteWise. Secara default, gateway

SiteWise Edge mengunggah semua jalur OPC-UA kecuali yang dimulai dengan `/Server/`. Anda dapat menggunakan karakter `*` dan `**` wildcard di filter node Anda untuk menyertakan beberapa jalur aliran data dengan satu filter. Untuk mempelajari cara menyiapkan sumber OPC-UA untuk gateway SiteWise Edge Anda, lihat [Mengkonfigurasi sumber data](#)

Note

AWS IoT SiteWise restart gateway SiteWise Edge Anda setiap kali Anda menambahkan atau mengedit sumber. Gateway SiteWise Edge Anda tidak akan menelan data saat dimulai ulang. Waktu untuk memulai ulang gateway SiteWise Edge Anda tergantung pada jumlah tag pada sumber gateway SiteWise Edge Anda. Waktu restart dapat berkisar dari beberapa detik (untuk gateway SiteWise Edge dengan beberapa tag) hingga beberapa menit (untuk gateway SiteWise Edge dengan banyak tag).

Tabel berikut mencantumkan wildcard yang dapat Anda gunakan untuk memfilter sumber data OPC-UA.

Kartu pengganti filter node OPC-UA

Wildcard	Deskripsi
<code>*</code>	Cocokkan satu level dalam jalur aliran data.
<code>**</code>	Cocokkan beberapa level di jalur aliran data.

Note

Jika Anda mengonfigurasi sumber dengan filter luas dan kemudian mengubah sumber untuk menggunakan filter yang lebih ketat, AWS IoT SiteWise berhenti menyimpan data yang tidak cocok dengan filter baru.

Example Contoh skenario menggunakan filter node

Pertimbangkan aliran data hipotetis berikut:

- `/WA/Factory 1/Line 1/PLC1`
- `/WA/Factory 1/Line 1/PLC2`

- /WA/Factory 1/Line 2/Counter1
- /WA/Factory 1/Line 2/PLC1
- /OR/Factory 1/Line 1/PLC1
- /OR/Factory 1/Line 2/Counter2

Menggunakan aliran data sebelumnya, Anda dapat menentukan filter node untuk membatasi data apa yang akan disertakan dari sumber OPC-UA Anda.

- Untuk memilih semua node dalam contoh ini, gunakan / atau /**/. Anda dapat menyertakan beberapa direktori atau folder dengan ** karakter wildcard.
- Untuk memilih semua aliran PLC data, gunakan /*/*/*/PLC* atau/***/PLC*.
- Untuk memilih semua penghitung dalam contoh ini, gunakan /**/Counter* atau/*/*/*/Counter*.
- Untuk memilih semua penghitung dari Line 2, gunakan/***/Line 2/Counter*.

Mengkonfigurasi otentikasi sumber data

Jika server OPC-UA Anda memerlukan kredensi otentikasi untuk terhubung, Anda dapat menggunakannya AWS Secrets Manager untuk membuat dan menyebarkan rahasia ke gateway Edge Anda. SiteWise AWS Secrets Manager mengenkripsi rahasia pada perangkat untuk menjaga nama pengguna dan kata sandi Anda tetap aman sampai Anda perlu menggunakannya. Untuk informasi selengkapnya, lihat [Manajer rahasia](#) di Panduan AWS IoT Greengrass Version 2 Pengembang.

Langkah 1: Buat rahasia otentikasi sumber

Anda dapat menggunakan AWS Secrets Manager untuk membuat rahasia otentikasi untuk sumber data Anda. Secara rahasia, tentukan **username** dan pasangan **password** nilai kunci yang berisi detail otentikasi untuk sumber data Anda.

Untuk membuat rahasia (konsol)

1. Navigasikan ke [konsol AWS Secrets Manager](#) tersebut.
2. Pilih Simpan rahasia baru.
3. Di bawah Tipe rahasia, pilih Jenis rahasia lainnya.
4. Di bawah pasangan kunci/nilai, lakukan hal berikut:

1. Di kotak input pertama, masukkan **username** dan di kotak input kedua masukkan nama pengguna.
2. Pilih Tambahkan baris.
3. Di kotak input pertama, masukkan **password** dan di kotak input kedua masukkan kata sandi.
5. Untuk kunci Enkripsi, pilih aws/secretsmanager, lalu pilih Berikutnya.
6. Di Store halaman rahasia baru, masukkan nama Rahasia.
7. (Opsional) Masukkan Deskripsi yang membantu Anda mengidentifikasi rahasia ini, lalu pilih Berikutnya.
8. (Opsional) Di Menyimpan halaman rahasia baru, nyalakan Rotasi otomatis. Untuk informasi selengkapnya, lihat [Memutar rahasia](#) di Panduan AWS Secrets Manager Pengguna.
9. Tentukan jadwal rotasi.
10. Pilih fungsi Lambda yang dapat memutar rahasia ini, lalu pilih Berikutnya.
11. Tinjau konfigurasi rahasia Anda, lalu pilih Store.

Untuk mengotorisasi gateway SiteWise Edge Anda untuk berinteraksi AWS Secrets Manager, peran IAM untuk gateway SiteWise Edge Anda harus mengizinkan tindakan tersebut `secretsmanager:GetSecretValue`. Anda dapat menggunakan perangkat inti Greengrass untuk mencari kebijakan IAM. Untuk informasi selengkapnya tentang memperbarui kebijakan IAM, lihat [Mengedit kebijakan IAM](#) di AWS Identity and Access Management Panduan Pengguna.

Example kebijakan

Ganti *secret-arn* dengan Amazon Resource Name (ARN) dari rahasia yang Anda buat di langkah sebelumnya. Untuk informasi selengkapnya tentang cara mendapatkan ARN rahasia, lihat [Mengambil rahasia Anda dari Panduan AWS Secrets Manager](#) Pengguna. AWS Secrets Manager

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "secretsmanager:GetSecretValue"
      ],
      "Effect": "Allow",
      "Resource": [
```

```
        "secret-arn"  
    ]  
}  
]  
}
```

Langkah 2: Terapkan rahasia ke perangkat gateway SiteWise Edge Anda

Anda dapat menggunakan AWS IoT SiteWise konsol untuk menyebarkan rahasia ke gateway SiteWise Edge Anda.

Untuk menyebarkan rahasia (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Gateway.
3. Dari daftar Gateways, pilih gateway SiteWise Edge target.
4. Di bagian konfigurasi Gateway, pilih tautan perangkat inti Greengrass untuk membuka inti yang terkait dengan gateway AWS IoT Greengrass Edge. SiteWise
5. Di panel navigasi, pilih Deployment.
6. Pilih penyebaran target, lalu pilih Revisi.
7. Pada halaman Tentukan target, pilih Selanjutnya.
8. Pada halaman Pilih komponen, di bagian Komponen publik, matikan Tampilkan hanya komponen yang dipilih.
9. Cari dan pilih `aws.greengrass.SecretManager` komponen, dan kemudian pilih Berikutnya.
10. Dari daftar Komponen yang dipilih, pilih `aws.greengrass.SecretManager` komponen, dan kemudian pilih Configure component.
11. Di bidang Configuration to merge, tambahkan objek JSON berikut.

Note

Ganti `rahasia-arn` dengan ARN rahasia yang Anda buat pada langkah sebelumnya. Untuk informasi selengkapnya tentang cara mendapatkan ARN rahasia, lihat [Mengambil rahasia Anda dari Panduan AWS Secrets Manager](#) Pengguna.AWS Secrets Manager

```
{
```



```
"cloudSecrets":[
  {
    "arn":"secret-arn"
  }
]
```

12. Pilih Konfirmasi.
13. Pilih Berikutnya.
14. Pada halaman Konfigurasi pengaturan lanjutan, pilih Berikutnya.
15. Tinjau konfigurasi penerapan Anda, lalu pilih Deploy.

Langkah 3: Tambahkan konfigurasi otentikasi

Anda dapat menggunakan AWS IoT SiteWise konsol untuk menambahkan konfigurasi otentikasi ke gateway SiteWise Edge Anda.

Untuk menambahkan konfigurasi otentikasi (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Dari daftar Gateways, pilih gateway SiteWise Edge target.
3. Dari daftar Sumber data, pilih sumber data target, lalu pilih Edit.
4. Pada halaman Tambahkan sumber data, pilih Konfigurasi lanjutan.
5. Untuk konfigurasi Otentikasi, pilih rahasia yang Anda gunakan di langkah sebelumnya.
6. Pilih Simpan.

Mengkonfigurasi sumber data ()AWS CLI

Anda dapat menggunakan AWS IoT SiteWise API dan AWS Command Line Interface menambahkan sumber ke gateway AWS IoT SiteWise Edge Anda. Anda menentukan sumber dalam kemampuan gateway SiteWise Edge. Kemampuan gateway SiteWise Edge mewakili fitur perangkat lunak yang berjalan di gateway SiteWise Edge, seperti kemampuan untuk mengumpulkan data industri dari sumber OPC-UA.

SiteWise Kemampuan gateway tepi memiliki komponen-komponen berikut:


- Konfigurasi — Dokumen JSON yang mendefinisikan semua sumber data untuk suatu kemampuan.

- **Namespace** — String unik yang mengidentifikasi jenis dan versi kemampuan. Misalnya, namespace kemampuan sumber OPC-UA adalah `hiotsitewise:opcuacollector:version`, di mana *versi adalah versi* kemampuan OPC-UA. Semua sumber OPC-UA didefinisikan dalam satu kemampuan dengan namespace ini.
- **Status sinkronisasi** — Status yang menunjukkan apakah suatu kemampuan disinkronkan antara AWS Cloud dan gateway SiteWise Edge. Status sinkronisasi dapat berupa salah satu dari yang berikut:
 - **IN_SYNC**— Gateway SiteWise Edge menjalankan konfigurasi kemampuan.
 - **OUT_OF_SYNC**— Gateway SiteWise Edge belum menerima konfigurasi kemampuan.
 - **SYNC_FAILED**— Gateway SiteWise Edge menolak konfigurasi kemampuan.

Setelah Anda memperbarui konfigurasi kemampuan, status sinkronisasi `OUT_OF_SYNC` sampai gateway SiteWise Edge menerima dan menerapkan atau menolak konfigurasi yang diperbarui.

Gunakan operasi berikut untuk menanyakan dan memperbarui sumber gateway SiteWise Edge dan konfigurasi kemampuan Anda:

- [DescribeGateway](#)— Mengambil informasi tentang gateway SiteWise Edge tertentu. Responsnya mencakup daftar ringkasan kemampuan, termasuk ruang nama kemampuan.
- [DescribeGatewayCapabilityConfiguration](#)— Mengambil konfigurasi kemampuan tertentu. Gunakan operasi ini untuk mengambil konfigurasi kemampuan untuk memperbarui.
- [ListGateways](#)— Daftar informasi tentang semua gateway SiteWise Edge. Respons tersebut mencakup daftar ringkasan kemampuan untuk setiap gateway SiteWise Edge, termasuk ruang nama kemampuan.
- [UpdateGatewayCapabilityConfiguration](#)— Memperbarui konfigurasi kemampuan gateway SiteWise Edge atau mendefinisikan konfigurasi kemampuan baru. Operasi ini mengidentifikasi kemampuan dengan namespace kemampuan. Jika Anda menyediakan namespace yang sudah ada, operasi ini memperbarui kemampuan untuk namespace tersebut. Jika tidak, operasi ini menciptakan kemampuan baru.

 Warning

[UpdateGatewayCapabilityConfiguration](#) Operasi menimpa konfigurasi kemampuan yang ada dengan konfigurasi yang Anda berikan di payload. Untuk menghindari penghapusan

konfigurasi kemampuan Anda, Anda harus menambahkan ke konfigurasi yang ada saat Anda memperbarui kemampuan.

SiteWise Kemampuan gateway tepi

- [???](#)
- [???](#)
- [???](#)

Memilih tujuan untuk data server sumber Anda

Data diekspor dari edge ke AWS IoT SiteWise real time, atau dalam batch menggunakan Amazon S3. Anda juga dapat mengirim aliran ke komponen lain menggunakan AWS IoT Greengrass aliran.

- AWS IoT SiteWise real-time — Pilih ini untuk mengirim data langsung ke AWS IoT SiteWise penyimpanan. Menelan dan memantau data secara real-time, dan memproses data di tepi.
- AWS IoT SiteWise Buffer menggunakan Amazon S3 - Kirim data dalam format parquet ke Amazon S3 dan kemudian impor ke penyimpanan. AWS IoT SiteWise Pilih opsi ini untuk menelan data dalam batch, dan menyimpan data historis dengan cara yang hemat biaya. Anda dapat mengonfigurasi lokasi bucket Amazon S3 pilihan Anda, dan frekuensi di mana Anda ingin data diunggah ke Amazon S3. Anda juga dapat memilih apa yang harus dilakukan dengan data setelah tertelan. AWS IoT SiteWise Anda dapat memilih untuk memiliki data yang tersedia di keduanya SiteWise dan Amazon S3 atau Anda dapat memilih untuk menghapusnya secara otomatis dari Amazon S3.
 - Bucket Amazon S3 adalah mekanisme pementasan dan buffering dan mendukung file dalam format parquet.
 - Jika Anda memilih kotak centang Impor data ke AWS IoT SiteWise penyimpanan, data diunggah ke Amazon S3 terlebih dahulu, lalu AWS IoT SiteWise ke penyimpanan.
 - Jika Anda memilih kotak centang Hapus data dari Amazon S3, data dihapus dari Amazon S3, setelah diimpor ke penyimpanan. SiteWise
 - Jika Anda menghapus kotak centang Hapus data dari Amazon S3, data disimpan baik di Amazon S3, dan di penyimpanan. SiteWise
 - Jika Anda menghapus kotak centang Impor data ke AWS IoT SiteWise penyimpanan, data hanya disimpan di Amazon S3. Itu tidak diimpor ke SiteWise penyimpanan.

Kunjungi [Mengelola penyimpanan data](#) untuk detail tentang berbagai opsi penyimpanan yang AWS IoT SiteWise disediakan. Untuk mempelajari lebih lanjut tentang opsi harga, lihat [AWS IoT SiteWise harga](#).

- AWS IoT Greengrass pengelola aliran — Gunakan pengelola AWS IoT Greengrass aliran untuk mengirim data ke AWS Cloud tujuan berikut: saluran masuk AWS IoT Analytics, streaming di Amazon Kinesis Data Streams, properti aset di, atau objek AWS IoT SiteWise di Amazon Simple Storage Service (Amazon S3). Untuk informasi selengkapnya, lihat [Mengelola aliran data di Panduan AWS IoT Greengrass Version 2 Pengembang AWS IoT Greengrass Inti](#).

Contoh berikut menunjukkan struktur pesan aliran data yang diperlukan. Semua bidang diperlukan.

```
{
  "assetId": "string",
  "propertyAlias": "string",
  "propertyId": "string",
  "propertyValues": [
    {
      "quality": "string",
      "timestamp": {
        "offsetInNanos": number,
        "timeInSeconds": number
      },
      "value": {
        "booleanValue": boolean,
        "doubleValue": number,
        "integerValue": number,
        "stringValue": "string"
      }
    }
  ]
}
```

Note

Pesan aliran data harus menyertakan (`assetId` dan `propertyId`) atau `propertyAlias` dalam strukturnya.

assetId

(Opsional) ID aset yang akan diperbarui.

propertyAlias

(Opsional) Alias yang mengidentifikasi properti, seperti jalur aliran data server OPC-UA. Sebagai contoh:

```
/company/windfarm/3/turbine/7/temperature
```

Untuk informasi selengkapnya, lihat [Memetakan aliran data industri ke properti aset](#) di AWS IoT SiteWise Panduan Pengguna.

propertyId

(Opsional) ID properti aset untuk entri ini.

propertyValues

(Wajib) Daftar nilai properti yang akan diunggah. Anda dapat menentukan hingga 10 elemen `propertyValues` array.

quality

(Opsional) Kualitas nilai properti aset.

timestamp

(Wajib) Stempel waktu dari nilai properti aset.

offsetInNanos

(Opsional) Offset nanodetik dari `timeInSeconds`

timeInSeconds

(Wajib) Tanggal stempel waktu, dalam hitungan detik, dalam format epoch Unix. Data nanodetik pecahan disediakan oleh `offsetInNanos`

value

(Wajib) Nilai properti aset.

Note

Hanya satu dari nilai berikut yang dapat ada di `value` lapangan.

booleanValue

(Opsional) Data properti aset tipe Boolean (true atau false).

doubleValue

(Opsional) Data properti aset tipe ganda (nomor floating point).

integerValue

(Opsional) Data properti aset tipe integer (bilangan bulat).

stringValue

(Opsional) Data properti aset tipe string (urutan karakter).

Menambahkan sumber data mitra ke gateway SiteWise Edge

Saat menggunakan gateway AWS IoT SiteWise Edge, Anda dapat menghubungkan sumber data mitra ke gateway SiteWise Edge Anda dan menerima data dari mitra di gateway SiteWise Edge dan AWS cloud Anda. Sumber data mitra ini adalah AWS IoT Greengrass komponen yang dikembangkan dalam kemitraan antara AWS dan mitra. Saat Anda menambahkan sumber data mitra, AWS IoT SiteWise akan membuat komponen ini dan menerapkannya di gateway SiteWise Edge Anda.

Untuk menambahkan sumber data mitra, lakukan hal berikut:

- [Menambahkan sumber data mitra](#)
- Buka portal web mitra dan konfigurasi sumber data mitra sehingga terhubung ke gateway SiteWise Edge.

Topik

- [Keamanan](#)
- [Menambahkan sumber data mitra](#)
- [Siapkan docker di gateway Edge Anda SiteWise](#)
- [SiteWise Sumber data mitra gateway tepi](#)

Keamanan

Sebagai bagian dari [Model Tanggung Jawab Bersama](#) antara AWS, pelanggan kami, dan mitra kami, berikut ini menjelaskan siapa yang bertanggung jawab atas berbagai aspek keamanan:

Tanggung jawab pelanggan

- Memeriksa pemasangan.
- Mengkonfigurasi akses jaringan yang diberikan kepada mitra.

AWSTanggung jawab

- Mengisolasi mitra dari sumber daya AWS cloud pelanggan kecuali yang dibutuhkan oleh mitra. Dalam hal ini, AWS IoT SiteWise konsumsi.
- Membatasi solusi mitra untuk penggunaan yang wajar dari sumber daya mesin gateway SiteWise Edge (CPU, memori, sistem file).

Tanggung jawab mitra

- Menggunakan default yang aman.
- Menjaga solusi tetap aman dari waktu ke waktu melalui tambalan dan pembaruan lain yang sesuai.
- Menjaga kerahasiaan data pelanggan.

Menambahkan sumber data mitra

Untuk menghubungkan sumber data mitra ke gateway SiteWise Edge Anda, tambahkan sebagai sumber data. Saat Anda menemukannya sebagai sumber data, AWS IoT SiteWise akan menyebarkan AWS IoT Greengrass komponen pribadi ke gateway SiteWise Edge Anda.

Prasyarat

Untuk menambahkan sumber data mitra, Anda harus melakukan hal berikut:

- Buat akun dengan mitra.
- Ikat akun.

Untuk membuat gateway SiteWise Edge dengan sumber data mitra

Jika Anda ingin membuat gateway SiteWise Edge baru, selesaikan langkah-langkahnya [Membuat gateway SiteWise Edge](#). Setelah Anda membuat gateway SiteWise Edge, ikuti langkah-langkah [Untuk menambahkan sumber data mitra ke gateway SiteWise Edge yang ada](#) untuk menambahkan sumber data mitra.

Untuk menambahkan sumber data mitra ke gateway SiteWise Edge yang ada

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Gateway.
3. Pilih gateway SiteWise Edge yang ingin Anda sambungkan dengan sumber data mitra.
4. Di bawah Sumber data, pilih Tambahkan sumber data.
5. Untuk jenis Sumber, pilih mitra yang ingin Anda sambungkan gateway SiteWise Edge Anda.

Note

Saat ini, EasyEdge adalah satu-satunya sumber data mitra yang tersedia. Pertama kali Anda menambahkan sumber EasyEdge data, Anda harus membuat [EasyEdge akun](#).

6. Masukkan nama untuk sumbernya.
7. Untuk memberikan akses mitra ke sumber data, pilih Otorisasi.
8. Untuk mengizinkan AWS IoT SiteWise memperbarui komponen AWS IoT SiteWise penerbit Anda dan, jika paket pemrosesan data diaktifkan, komponen AWS IoT SiteWise prosesor, pilih Perbarui komponen.
9. Pilih Simpan.

Siapkan docker di gateway Edge Anda SiteWise

Untuk menambahkan sumber data mitra, [Docker Engine](#) 1.9.1 atau yang lebih baru harus diinstal pada perangkat lokal Anda.

Note

Versi 20.10 adalah versi terbaru yang diverifikasi untuk bekerja dengan perangkat lunak gateway SiteWise Edge.

Untuk memverifikasi Docker diinstal

Untuk memverifikasi Docker diinstal, jalankan perintah berikut dari terminal yang terhubung ke gateway SiteWise Edge Anda:


```
docker info
```

Jika perintah mengembalikan `docker is not recognized` hasil, atau versi Docker yang lebih lama diinstal, [Instal Docker Engine sebelum melanjutkan](#).

Untuk mengatur Docker

Pengguna sistem yang menjalankan komponen kontainer Docker harus memiliki izin root atau administrator, atau Anda harus mengonfigurasi Docker untuk menjalankannya sebagai pengguna non-root atau non-administrator.

Pada perangkat Linux, Anda harus menambahkan `ggc_user` pengguna ke `docker` grup untuk memanggil perintah Docker tanpa `sudo`.

Untuk menambah `ggc_user`, atau pengguna non-root yang Anda gunakan untuk menjalankan komponen kontainer Docker, ke `docker` grup, jalankan perintah berikut:

```
sudo usermod -aG docker ggc_user
```

Untuk informasi selengkapnya, lihat [langkah-langkah pasca-instalasi Linux untuk Docker Engine](#).

SiteWise Sumber data mitra gateway tepi

Gunakan informasi di bawah ini untuk mengonfigurasi sumber data mitra.

EasyEdge

Portal:

<https://studio.easyedge.io/>

EasyEdge dokumentasi:

[EasyEdge untuk AWS](#)

Menggunakan paket

AWS IoT SiteWiseEdge gateway menggunakan paket yang berbeda untuk menentukan cara mengumpulkan dan memproses data Anda.

Saat ini, paket berikut tersedia:

- Paket pengumpulan data — Gunakan paket ini untuk mengumpulkan data industri Anda dan mengarahkannya ke tujuan AWS Cloud. Secara default, paket ini diaktifkan secara otomatis untuk gateway SiteWise Edge Anda.
- Paket pemrosesan data — Gunakan paket ini untuk mengaktifkan komunikasi gateway SiteWise Edge dengan model dan aset aset yang dikonfigurasi tepi. Anda dapat menggunakan konfigurasi tepi untuk mengontrol data aset apa yang akan dihitung dan diproses di tempat. Anda kemudian dapat mengirim data Anda ke AWS IoT SiteWise atau AWS layanan lain. Untuk informasi selengkapnya tentang paket pemrosesan data, lihat [the section called “Mengaktifkan pemrosesan data tepi”](#).

Paket upgrade

Important

Memutakhirkan versi paket pemrosesan data dari sebelumnya (dan termasuk) 2.0.x ke versi 2.1.x akan mengakibatkan hilangnya data pengukuran yang disimpan secara lokal.

SiteWise Edge gateway menggunakan paket yang berbeda untuk menentukan cara mengumpulkan dan memproses data Anda. Anda dapat menggunakan AWS IoT SiteWise konsol untuk meningkatkan paket.


Untuk meng-upgrade paket (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Gateway.
3. Dalam daftar Gateways, pilih gateway SiteWise Edge dengan paket yang ingin Anda tingkatkan.
4. Di halaman ringkasan gateway SiteWise Edge, pilih Pembaruan.

Note

Anda hanya dapat memutakhirkan paket yang diaktifkan. Untuk menemukan daftar paket yang diaktifkan untuk gateway SiteWise Edge ini, pilih Ikhtisar, lalu lihat bagian Kemampuan tepi.

5. Di bagian Pembaruan paket, lakukan salah satu hal berikut.
 - Untuk kolektor OPC-UA, pilih versi, lalu pilih Deploy.
 - Untuk Publisher, pilih versi, lalu pilih Deploy.
 - Untuk paket pemrosesan data, pilih versi, lalu pilih Deploy.
6. Untuk mengonfirmasi penerapan, pilih Deploy. Di kolom Deployment dalam daftar Gateways, Anda akan melihat Selesai. Jika Anda tidak melihat pembaruan pada status penerapan, segarkan halaman.

 Note

- Gunakan hanya satu paket pada satu waktu. Jika Anda menerapkan beberapa paket secara bersamaan, hanya paket terakhir yang Anda pilih yang akan di-deploy.

Jika Anda mengalami masalah dalam meningkatkan paket, lihat [Tidak dapat menerapkan paket ke gateway SiteWise Edge](#).

Mengelola SiteWise gateway Edge

Anda dapat menggunakan operasi AWS IoT SiteWise konsol dan API untuk mengelola gateway AWS IoT SiteWise Edge. Anda juga dapat menggunakan aplikasi [AWS OpsHub for AWS IoT SiteWise for Windows](#) untuk mengelola beberapa aspek gateway SiteWise Edge Anda dari perangkat lokal Anda.

Kami sangat menyarankan Anda menggunakan AWS IoT SiteWise aplikasi AWS OpsHub for untuk memantau penggunaan disk pada perangkat lokal Anda. Anda juga dapat memantau CloudWatch metrik `Gateway.AvailableDiskSpace` dan `Gateway.UsedPercentageDiskSpace` Amazon dan membuat alarm untuk mendapatkan pemberitahuan saat ruang disk semakin rendah. Untuk informasi selengkapnya tentang CloudWatch alarm Amazon, lihat [Membuat CloudWatch alarm berdasarkan ambang batas statis](#).

Pastikan perangkat Anda memiliki cukup ruang untuk data yang akan datang. Ketika Anda akan kehabisan ruang di perangkat lokal Anda, layanan secara otomatis menghapus sejumlah kecil data dengan stempel waktu tertua untuk memberi ruang bagi data yang akan datang.

Untuk memeriksa apakah layanan menghapus data Anda, lakukan hal berikut:

1. Masuk ke AWS IoT SiteWise aplikasi AWS OpsHub untuk.
2. Pilih Pengaturan.
3. Untuk Log, tentukan rentang waktu, lalu pilih Unduh.
4. Buka zip file log.
5. Jika file log berisi pesan berikut, layanan menghapus data Anda: *jumlah* byte data telah dihapus untuk mencegah penyimpanan gateway SiteWise Edge kehabisan ruang.

Mengelola gateway SiteWise Edge Anda dengan AWS IoT SiteWise konsol

Anda dapat menggunakan AWS IoT SiteWise konsol untuk mengonfigurasi, memperbarui, dan memantau semua gateway SiteWise Edge di akun Anda AWS .

[Anda dapat melihat gateway SiteWise Edge Anda dengan menavigasi ke halaman Edge Gateways di konsol.AWS IoT SiteWise](#) Untuk mengakses halaman detail gateway Edge untuk gateway tertentu, pilih nama gateway Edge.

Dari tab Ikhtisar halaman detail gateway Edge, Anda dapat melakukan hal berikut:

- Di bagian Sumber data, perbarui konfigurasi sumber data dan konfigurasi sumber data tambahan
- Pilih Buka CloudWatch metrik untuk melihat jumlah titik data yang dicerna per sumber data di konsol metrik CloudWatch
- Di bagian kemampuan Edge, tambahkan paket data ke gateway SiteWise Edge Anda dengan mengklik Edit
- Di bagian konfigurasi Gateway, lihat status konektivitas gateway SiteWise Edge Anda
- Di bagian konfigurasi Publisher, lihat status sinkronisasi gateway SiteWise Edge dan konfigurasi komponen AWS IoT SiteWise penerbit

Dari tab Pembaruan pada halaman detail gateway Edge, Anda dapat melihat komponen saat ini dan versi paket yang digunakan ke gateway Edge. Ini juga tempat Anda menerapkan versi baru, saat tersedia.

Mengelola gateway SiteWise Edge menggunakan for AWS OpsHubAWS IoT SiteWise

Anda menggunakan AWS IoT SiteWise aplikasi AWS OpsHub for untuk mengelola dan memantau gateway SiteWise Edge Anda. Aplikasi ini menyediakan opsi pemantauan dan manajemen berikut:

- Di bawah Ikhtisar, Anda dapat melakukan hal berikut:
 - Lihat detail gateway SiteWise Edge yang membantu Anda mendapatkan wawasan tentang data perangkat gateway SiteWise Edge Anda, mengidentifikasi masalah, dan meningkatkan kinerja gateway SiteWise Edge.
 - Lihat portal SiteWise Monitor yang memantau data dari server lokal dan peralatan di tepi. Untuk informasi selengkapnya, lihat [Apa yang ada AWS IoT SiteWise Monitor](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.
- Di bawah Health, ada dasbor yang menampilkan data dari gateway SiteWise Edge Anda. Pakar domain, seperti insinyur proses, dapat menggunakan dasbor untuk melihat ikhtisar perilaku gateway SiteWise Edge.
- Di bawah Aset, lihat aset yang digunakan ke perangkat lokal dan nilai terakhir yang dikumpulkan atau dihitung untuk properti aset.
- Di bawah Pengaturan, Anda dapat melakukan hal berikut:
 - Jika Paket Pemrosesan Data diinstal, lihat informasi konfigurasi gateway SiteWise Edge dan sinkronkan sumber daya dengan AWS Cloud.
 - Unduh file otentikasi yang dapat Anda gunakan untuk mengakses gateway SiteWise Edge dengan menggunakan alat lain.
 - Unduh log yang dapat Anda gunakan untuk memecahkan masalah gateway SiteWise Edge.
 - Lihat AWS IoT SiteWise komponen yang digunakan ke gateway SiteWise Edge.

Important

Berikut ini diperlukan untuk digunakan AWS OpsHub untuk AWS IoT SiteWise:

- Perangkat lokal Anda dan AWS IoT SiteWise aplikasi AWS OpsHub for harus terhubung ke jaringan yang sama.
- Paket pemrosesan data harus diaktifkan.

Untuk mengelola gateway SiteWise Edge menggunakan AWS OpsHub

1. Unduh dan instal aplikasi [AWS OpsHubAWS IoT SiteWise untuk Windows](#).
2. Buka aplikasi .
3. Jika Anda tidak memiliki kredensi lokal yang disiapkan untuk gateway Anda, ikuti langkah-langkah di bawah ini [Mengakses gateway SiteWise Edge Anda menggunakan kredensyal sistem operasi lokal](#) untuk mengaturnya.
4. Anda dapat masuk ke gateway SiteWise Edge Anda dengan kredensi Linux atau Lightweight Directory Access Protocol (LDAP) Anda. Untuk masuk ke gateway SiteWise Edge Anda, lakukan salah satu hal berikut:

Linux

1. Untuk nama host atau alamat IP, masukkan nama host atau alamat IP perangkat lokal Anda.
2. Untuk Otentikasi, pilih Linux.
3. Untuk nama pengguna, masukkan nama pengguna sistem operasi Linux Anda.
4. Untuk Kata Sandi, masukkan kata sandi sistem operasi Linux Anda.
5. Pilih Masuk.

LDAP

1. Untuk nama host atau alamat IP, masukkan nama host atau alamat IP perangkat lokal Anda.
2. Untuk Otentikasi, pilih LDAP.
3. Untuk nama Pengguna, masukkan nama pengguna LDAP Anda.
4. Untuk Kata Sandi, masukkan kata sandi LDAP Anda.
5. Pilih Masuk.

Mengakses gateway SiteWise Edge Anda menggunakan kredensyal sistem operasi lokal

Selain Lightweight Directory Access Protocol (LDAP), Anda dapat menggunakan kredensi Linux atau Windows untuk mengakses gateway Edge Anda SiteWise .

⚠ Important

Untuk mengakses gateway SiteWise Edge Anda dengan kredensi Linux, Anda harus mengaktifkan paket pemrosesan data untuk gateway SiteWise Edge Anda.

Mengakses gateway SiteWise Edge Anda menggunakan kredensial sistem operasi Linux

Langkah-langkah berikut mengasumsikan bahwa Anda menggunakan perangkat dengan Ubuntu. Jika Anda menggunakan distribusi Linux yang berbeda, lihat dokumentasi yang relevan untuk perangkat Anda.

Untuk membuat kumpulan pengguna Linux

1. Untuk membuat grup admin, jalankan perintah berikut.

```
sudo groupadd --system SWE_ADMIN_GROUP
```

Pengguna dalam SWE_ADMIN_GROUP grup dapat mengizinkan akses admin untuk gateway SiteWise Edge.

2. Untuk membuat grup pengguna, jalankan perintah berikut.

```
sudo groupadd --system SWE_USER_GROUP
```

Pengguna dalam SWE_USER_GROUP grup dapat mengizinkan akses hanya-baca untuk gateway SiteWise Edge.

3. Untuk menambahkan pengguna ke grup admin, jalankan perintah berikut. Ganti *nama pengguna* dan *kata sandi* dengan nama pengguna dan kata sandi yang ingin Anda tambahkan.

```
sudo useradd -p $(openssl passwd -1 password) user-name
```

4. Untuk menambahkan pengguna ke salah satu SWE_ADMIN_GROUP atau SWE_USER_GROUP, ganti *nama pengguna dengan nama* pengguna yang Anda tambahkan pada langkah sebelumnya.

```
sudo usermod -a -G SWE_ADMIN_GROUP user-name
```

Anda sekarang dapat menggunakan nama pengguna dan kata sandi untuk masuk ke gateway SiteWise Edge pada AWS IoT SiteWise aplikasi AWS OpsHub for.

Mengakses gateway SiteWise Edge Anda menggunakan kredensial Windows

Langkah-langkah berikut mengasumsikan bahwa Anda menggunakan perangkat dengan Windows.

 Important

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Buat kebijakan kata sandi yang kuat dengan setidaknya 12 karakter dan kombinasi huruf besar, huruf kecil, angka, dan simbol. Selain itu, tetapkan aturan Windows Firewall untuk memungkinkan lalu lintas masuk pada port 443 dan untuk memblokir lalu lintas masuk di semua port lainnya.

Untuk membuat kumpulan pengguna Windows Server

1. Jalankan PowerShell sebagai administrator.
 - a. Di server Windows tempat Anda ingin menginstal SiteWise Edge Gateway, masuk sebagai administrator.
 - b. Masukkan PowerShell di bilah pencarian Windows.
 - c. Di hasil pencarian, klik kanan pada PowerShell aplikasi Windows. Pilih Jalankan sebagai Administrator.
2. Untuk membuat grup admin, jalankan perintah berikut.

```
net localgroup SWE_ADMIN_GROUP /add
```

Anda harus menjadi pengguna dalam SWE_ADMIN_GROUP grup untuk mengizinkan akses admin untuk gateway SiteWise Edge.

3. Untuk membuat grup pengguna, jalankan perintah berikut.

```
net localgroup SWE_USER_GROUP /add
```

Anda harus menjadi pengguna dalam SWE_USER_GROUP grup untuk mengizinkan akses siapa-saja untuk gateway SiteWise Edge.

4. Untuk menambahkan pengguna, jalankan perintah berikut. Ganti *nama pengguna* dan *kata sandi* dengan nama pengguna dan kata sandi yang ingin Anda buat.


```
net user user-name password /add
```

5. Untuk menambahkan pengguna ke grup admin, jalankan perintah berikut. Ganti *nama pengguna* dengan nama pengguna yang ingin Anda tambahkan.

```
net localgroup SWE_ADMIN_GROUP user-name /add
```

Anda sekarang dapat menggunakan nama pengguna dan kata sandi untuk masuk ke gateway SiteWise Edge pada AWS IoT SiteWise aplikasi AWS OpsHub for.

Mengelola sertifikat gateway SiteWise Edge

Anda dapat menggunakan SiteWise Monitor dan aplikasi pihak ketiga, seperti Grafana, di perangkat gateway SiteWise Edge Anda. Aplikasi ini memerlukan koneksi TLS ke layanan. SiteWise Edge gateway saat ini menggunakan sertifikat yang ditandatangani sendiri. Jika Anda menggunakan browser untuk membuka aplikasi, seperti portal SiteWise Monitor, Anda mungkin menerima peringatan untuk sertifikat yang tidak dipercaya.

Berikut ini menunjukkan cara mengunduh sertifikat tepercaya dari AWS IoT SiteWise aplikasi AWS OpsHub for.

1. Masuk ke aplikasi.
2. Pilih Pengaturan.
3. Untuk Otentikasi, pilih Unduh sertifikat.

Berikut ini mengasumsikan bahwa Anda menggunakan Google Chrome atau FireFox. Jika Anda menggunakan browser lain, lihat dokumentasi yang relevan untuk browser Anda. Untuk menambahkan sertifikat yang Anda unduh pada langkah sebelumnya ke browser, lakukan salah satu hal berikut:

- Jika Anda menggunakan Google Chrome, ikuti [Menyiapkan sertifikat](#) di dokumentasi Bantuan Google Chrome Enterprise.
- Jika Anda menggunakan Firefox, ikuti To [Load the Certificate ke Mozilla atau Firefox Browser](#) dalam dokumentasi Oracle.

Mengubah versi paket komponen gateway SiteWise Edge

Anda dapat menggunakan AWS IoT SiteWise konsol untuk mengubah versi paket komponen di gateway SiteWise Edge Anda.

Untuk mengubah versi paket komponen gateway SiteWise Edge

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Gateway.
3. Pilih gateway SiteWise Edge yang ingin Anda ubah versi paketnya.
4. Di bawah konfigurasi Gateway, pilih Lihat versi perangkat lunak.
5. Pada halaman Edit versi perangkat lunak, untuk paket yang ingin Anda perbarui versi, pilih versi yang ingin Anda gunakan dan pilih Deploy.
6. Pilih Selesai.

Running SiteWise Edge di Tepi Industri Siemens

Anda dapat menelan data dari perangkat Siemens Industrial Edge ke perangkat Anda Akun AWS dengan menjalankan gateway SiteWise Edge di perangkat. Untuk melakukan ini, Anda membuat sumber daya gateway SiteWise Edge dengan target penyebaran perangkat Siemens Industrial Edge - baru, unduh file konfigurasi, dan unggah ke aplikasi Siemens Anda melalui portal Siemens Industrial Edge Management (IEM). Untuk informasi selengkapnya tentang menjalankan AWS IoT SiteWise Edge on Siemens Industrial Edge, termasuk cara menyiapkan sumber daya Siemens yang diperlukan, lihat [Apa itu Industrial Edge?](#) dalam dokumentasi Siemens.

Note

Siemens bukan vendor atau pemasok untuk AWS IoT SiteWise Edge. Siemens Industrial Edge Marketplace adalah pasar independen.

Topik

- [Prasyarat](#)
- [Keamanan](#)
- [Membuat file konfigurasi](#)
- [Pemecahan Masalah](#)

- [Hubungi kami](#)

Prasyarat

Untuk menjalankan AWS IoT SiteWise Edge di Siemens Industrial Edge, Anda memerlukan yang berikut ini:

- Akun Platform [Pertukaran Digital Siemens](#)
- Akun Siemens Industrial Edge Hub (iehub)
- Contoh Manajemen Tepi Industri Siemens (IEM)
- Baik Perangkat Tepi Industri Siemens (IED) atau Perangkat virtual Siemens Industrial Edge (iEVD)
- Akses ke target penyebaran perangkat Siemens Industrial Edge. Untuk mendapatkan akses, buka [AWS IoT SiteWise konsol](#) dan pilih Minta akses.

Keamanan

Sebagai bagian dari [Model Tanggung Jawab Bersama](#) antara AWS, pelanggan kami, dan mitra kami, berikut ini menjelaskan siapa yang bertanggung jawab atas berbagai aspek keamanan:

Tanggung jawab pelanggan

- Memeriksa pemasangan.
- Mengkonfigurasi akses jaringan yang diberikan kepada mitra.
- Secara fisik mengamankan perangkat yang menjalankan AWS IoT SiteWise Edge.

AWS tanggung jawab

- Mengisolasi mitra dari sumber daya AWS cloud pelanggan.

Tanggung jawab mitra

- Menggunakan default yang aman.
- Menjaga solusi tetap aman dari waktu ke waktu melalui tambalan dan pembaruan lain yang sesuai.
- Menjaga kerahasiaan data pelanggan.
- Memeriksa aplikasi lain yang tersedia di pasar mitra.

Selama tahap pratinjau fitur ini, data pelanggan yang AWS IoT SiteWise di-cache pada perangkat mitra dapat diakses oleh mitra dan aplikasi lain yang diinstal melalui pasar mitra.

Membuat file konfigurasi

Setelah Anda memiliki akun Siemens dan instans IEM yang tepat, Anda dapat membuat gateway SiteWise Edge dari perangkat jenis penyebaran Siemens Industrial Edge.

Untuk membuat file konfigurasi

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih gateway Edge.
3. Pilih Buat gateway.
4. Untuk jenis Deployment, pilih perangkat Siemens Industrial Edge - baru.
5. Masukkan nama untuk gateway SiteWise Edge Anda atau gunakan nama yang dihasilkan oleh AWS IoT SiteWise.
6. (Opsional) Di bawah konfigurasi lanjutan, lakukan hal berikut:
 - Masukkan nama untuk AWS IoT Core Benda Anda atau gunakan nama yang dihasilkan oleh AWS IoT SiteWise.
7. Pilih Buat gateway.
8. Di kotak dialog Generate SiteWise Edge gateway configuration file, pilih Generate and download. AWS IoT SiteWise secara otomatis menghasilkan file konfigurasi yang akan Anda gunakan untuk mengkonfigurasi aplikasi AWS IoT SiteWise Edge.

Important

Pastikan Anda menyimpan file konfigurasi di lokasi yang aman. Anda akan menggunakan file nanti.

Sekarang setelah Anda membuat gateway SiteWise Edge, lakukan hal berikut untuk menyelesaikan pengaturan gateway SiteWise Edge Anda:

1. [Tambahkan sumber data](#)
2. [Konfigurasi komponen Publisher](#)

Setelah Anda memiliki file konfigurasi dan gateway SiteWise Edge dikonfigurasi, unduh aplikasi AWS IoT SiteWise Edge dari Siemens Industrial Edge Marketplace dan instal menggunakan portal

Siemens Industrial Edge Management (IEM). Kemudian, akses perangkat Siemens Industrial Edge Anda melalui portal Siemens Industrial Edge Management (IEM) dan unggah file konfigurasi pada perangkat tempat Anda ingin menginstal gateway Edge. SiteWise

Pemecahan Masalah

Untuk memecahkan masalah gateway SiteWise Edge pada perangkat Siemens Industrial Edge Anda, Anda dapat mengakses log untuk aplikasi melalui portal Siemens Industrial Edge Management (IEM) atau Siemens Industrial Edge Device (IED). Untuk informasi selengkapnya, lihat [Mengunduh Log](#) di dokumentasi Siemens.

Saya melihat 'SESSION_TAKEN_OVER' atau 'com.aws.greengrass.mqttclient. MqttClient: Gagal mempublikasikan pesan melalui Spooler dan akan mencoba lagi. ' di log

Jika Anda melihat peringatan yang menyertakan SESSION_TAKEN_OVER atau kesalahan yang disertakan `com.aws.greengrass.mqttclient.MqttClient: Failed to publish the message via Spooler and will retry.` dalam log Anda di `/greengrass/v2/logs/greengrass.log`, Anda mungkin mencoba menggunakan file konfigurasi yang sama untuk beberapa gateway SiteWise Edge di beberapa perangkat. Setiap gateway SiteWise Edge membutuhkan file konfigurasi unik untuk terhubung ke Anda Akun AWS.

Saya melihat 'com.aws.greengrass.deployment. IotJobsHelper: Tidak ada pekerjaan penerapan yang ditemukan. ' atau 'Hasil penerapan sudah dilaporkan. ' di log

Jika Anda melihat `com.aws.greengrass.deployment.IotJobsHelper: No deployment job found.` atau `Deployment result already reported.` dalam log Anda di `/greengrass/v2/logs/greengrass.log`, Anda mungkin mencoba untuk menggunakan kembali file konfigurasi yang sama.

Ada beberapa solusi:

- Jika Anda ingin menggunakan kembali file konfigurasi, lakukan hal berikut:
 1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
 2. Di panel navigasi, pilih Gateway.
 3. Pilih gateway SiteWise Edge yang ingin Anda gunakan kembali.
 4. Pilih tab Pembaruan.

5. Pilih versi Publisher yang berbeda dan pilih Deploy.
- Ikuti langkah-langkah [Membuat file konfigurasi](#) untuk membuat file konfigurasi baru.

Saya melihat 'File konfigurasi hilang AWS_REGION' di log.

Jika Anda melihat Config file missing AWS_REGION di log Siemens, JSON dari file konfigurasi telah rusak. Anda harus membuat file konfigurasi baru. Ikuti langkah-langkah [Membuat file konfigurasi](#) untuk membuat file konfigurasi baru.

Hubungi kami

- Jika Anda ingin meminta akses ke aplikasi, buka [AWS IoT SiteWise konsol](#) dan pilih Minta akses.
- Jika Anda ingin membantu memecahkan masalah aplikasi, buka [AWS IoT SiteWise konsol](#), navigasikan ke halaman detail gateway SiteWise Edge, dan pilih Dapatkan dukungan.

Memfilter aset di gateway SiteWise Edge

Anda dapat menggunakan pemfilteran tepi untuk mengelola aset Anda secara lebih efisien dengan mengirimkan hanya sebagian aset ke gateway SiteWise Edge tertentu untuk digunakan dalam pemrosesan data. Jika aset Anda diatur dalam struktur pohon, atau orang tua-anak, Anda dapat menyiapkan kebijakan IAM yang dilampirkan ke peran IAM gateway SiteWise Edge yang hanya mengizinkan akar pohon, atau induk, dan anak-anaknya dikirim ke gateway Edge tertentu. SiteWise

Note

Jika Anda mengatur aset yang ada ke dalam struktur pohon, setelah Anda membuat struktur, masuk ke setiap aset yang ada yang Anda tambahkan ke struktur dan pilih Edit lalu pilih Simpan untuk memastikan AWS IoT SiteWise mengenali struktur baru.

Menyiapkan penyaringan tepi

Siapkan pemfilteran tepi di gateway SiteWise Edge Anda dengan menambahkan kebijakan IAM berikut ke peran IAM gateway SiteWise Edge, ganti `< root-asset-id >` dengan ID aset root yang ingin Anda kirim ke gateway Edge. SiteWise

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Deny",
    "Action": [
      "iotsitewise:DescribeAsset",
      "iotsitewise>ListAssociatedAssets"
    ],
    "Resource": "arn:aws:iotsitewise:*:*:asset/*",
    "Condition": {
      "StringNotLike": {
        "iotsitewise:assetHierarchyPath": "/<root-asset-id>*"
      }
    }
  }
]
```

Jika ada aset saat ini di gateway SiteWise Edge Anda yang ingin Anda hapus, masuk ke gateway SiteWise Edge Anda dan jalankan perintah berikut untuk memaksa gateway SiteWise Edge disinkronkan AWS IoT SiteWise dengan menghapus cache.

```
sudo rm /greengrass/v2/work/aws.iot.SiteWiseEdgeProcessor/sync-app/
sync_resource_bundles/edge.json
```

Menggunakan AWS IoT SiteWise API di tepi

Anda dapat menggunakan subset API yang tersedia bersama dengan AWS IoT SiteWise API khusus tepi untuk berinteraksi dengan model aset dan asetnya di edge. Model aset harus dikonfigurasi untuk berjalan di tepi. Untuk informasi selengkapnya, lihat [Memproses data di tepi](#).

Gunakan API ini untuk mengumpulkan data tentang model dan aset aset Anda, memantau portal dan metrik dashboard yang diterapkan, dan mengumpulkan data aset di edge. Ini menyediakan host pusat di jaringan Anda untuk interaksi dengan AWS IoT SiteWise tanpa memerlukan panggilan API web.

Topik

- [Semua API yang tersedia untuk digunakan dengan perangkat AWS IoT SiteWise edge](#)
- [API khusus tepi untuk digunakan dengan AWS IoT SiteWise perangkat tepi](#)
- [Tutorial: Mendapatkan daftar model aset di gateway SiteWise Edge](#)

Semua API yang tersedia untuk digunakan dengan perangkat AWS IoT SiteWise edge

Saat bekerja dengan perangkat di tepi, Anda dapat menggunakan berbagai API untuk berinteraksi AWS IoT SiteWise dan menyelesaikan tugas secara lokal di perangkat.

AWS IoT SiteWiseAPI yang tersedia

AWS IoT SiteWiseAPI berikut tersedia di perangkat edge:

- [ListAssetModels](#)
- [DescribeAssetModel](#)
- [ListAssets](#)
- [DescribeAsset](#)
- [DescribeAssetProperty](#)
- [ListAssociatedAssets](#)
- [GetAssetPropertyAggregates](#)
- [GetAssetPropertyValue](#)
- [GetAssetPropertyValueHistory](#)
- [ListDashboards](#)
- [ListPortals](#)
- [ListProjectAssets](#)
- [ListProjects](#)
- [DescribeDashboard](#)
- [DescribePortal](#)
- [DescribeProject](#)

API khusus tepi yang tersedia

API berikut digunakan secara lokal pada perangkat di edge:

- [Mengautentikasi](#)— Gunakan API ini untuk mendapatkan kredensial sementara SigV4 yang akan Anda gunakan untuk melakukan panggilan API.

API khusus tepi untuk digunakan dengan AWS IoT SiteWise perangkat tepi

Selain AWS IoT SiteWise API yang tersedia di edge, ada yang khusus tepi. API edge-specific tersebut dijelaskan di bawah ini.

Mengautentikasi

Mendapat kredensialnya dari gateway SiteWise Edge. Anda harus menambahkan pengguna lokal atau terhubung ke sistem Anda menggunakan LDAP atau kumpulan pengguna Linux. Untuk informasi selengkapnya tentang menambahkan pengguna, lihat [kumpulan pengguna LDAP atau Linux](#).

Sintaksis Permintaan

```
POST /authenticate HTTP/1.1
Content-type: application/json
{
  "username": "string",
  "password": "string",
  "authMechanism": "string"
}
```

Parameter permintaan URI

Permintaan tidak menggunakan parameter URI apa pun.

Isi permintaan

Permintaan menerima data berikut dalam format JSON.

nama pengguna

Nama pengguna yang digunakan untuk memvalidasi panggilan permintaan.

Tipe: String

Diperlukan: Ya

password

Kata sandi pengguna yang meminta kredensi.

Tipe: String

Diperlukan: Ya

AuthMechanism

Metode otentikasi untuk memvalidasi pengguna ini di host.

Jenis: String

Nilai valid: ldap, linux, winnt

Diperlukan: Ya

Sintaksis Respons

```
HTTP/1.1 200
Content-type: application/json
{
  "accessKeyId": "string",
  "secretAccessKey": "string",
  "sessionToken": "string",
  "region": "edge"
}
```

Elemen jawaban

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam format JSON.

accessKeyId

ID kunci akses yang mengidentifikasi kredensi keamanan sementara.

Kendala Panjang: Panjang minimum 16. Panjang maksimum 128.

Pola: `[\w]*`

secretAccessKey

Kunci akses rahasia yang dapat digunakan untuk menandatangani permintaan.

Jenis: String

sessionToken

Token yang harus diteruskan pengguna ke API layanan untuk menggunakan kredensial sementara.

Jenis: String

region

Wilayah yang Anda targetkan untuk panggilan API.

Jenis: KONSTAN - edge

Kesalahan

IllegalArgumentException

Permintaan ditolak karena dokumen badan yang disediakan cacat. Pesan kesalahan menjelaskan kesalahan tertentu.

Kode Status HTTP: 400

AccessDeniedException

Pengguna tidak memiliki kredensial yang valid berdasarkan Penyedia Identitas saat ini. Pesan kesalahan menjelaskan Mekanisme otentikasi.

Kode Status HTTP: 403

TooManyRequestsException

Permintaan telah mencapai batas upaya otentikasi. Pesan kesalahan berisi jumlah waktu untuk menunggu sampai upaya otentikasi baru dilakukan.

Kode Status HTTP: 429

Tutorial: Mendapatkan daftar model aset di gateway SiteWise Edge

Anda dapat menggunakan subset API yang tersedia bersama dengan AWS IoT SiteWise API khusus tepi untuk berinteraksi dengan model aset dan asetnya di edge. Tutorial ini akan memandu Anda mendapatkan kredensial sementara ke gateway AWS IoT SiteWise Edge dan mendapatkan daftar model aset di gateway SiteWise Edge.

Prasyarat

Dalam langkah-langkah tutorial ini Anda dapat menggunakan berbagai alat. Untuk menggunakan alat ini, pastikan Anda telah menginstal prasyarat yang sesuai.

Untuk menyelesaikan tutorial ini, Anda memerlukan hal berikut:

- Sebuah komputer yang diarahkan dan berjalan [SiteWise Persyaratan gateway tepi](#)
- Akses ke gateway SiteWise Edge Anda di jaringan yang sama melalui port 443.
- [OpenSSL diinstal](#)
- (AWS OpsHub untuk AWS IoT SiteWise) [AWS IoT SiteWise Aplikasi AWS OpsHub untuk](#)
- (curl) [curl](#) terpasang
- [\(Python\) urllib3 diinstal](#)
- [\(Python\) Python3 diinstal](#)
- [\(Python\) Boto3 diinstal](#)
- (Python) diinstal [BotoCore](#)

Langkah 1: Dapatkan sertifikat yang ditandatangani layanan gateway SiteWise Edge

Untuk membuat koneksi TLS ke API yang tersedia di gateway SiteWise Edge, Anda memerlukan sertifikat tepercaya. Anda dapat membuat sertifikat ini menggunakan AWS OpsHub OpenSSL atau for. AWS IoT SiteWise

OpenSSL

Note

Anda perlu [OpenSSL](#) diinstal untuk menjalankan perintah ini.

Buka terminal dan jalankan perintah berikut untuk mendapatkan sertifikat yang ditandatangani dari gateway SiteWise Edge. Ganti `<sitewise_gateway_ip>` dengan IP gateway SiteWise Edge.

```
openssl s_client -connect <sitewise_gateway_ip>:443 </dev/null 2>/dev/null | openssl x509 -outform PEM > GatewayCert.pem
```

AWS OpsHub for AWS IoT SiteWise

Anda dapat menggunakan AWS OpsHub untuk AWS IoT SiteWise. Untuk informasi selengkapnya, lihat [Mengelola SiteWise gateway Edge](#).

Jalur absolut ke sertifikat gateway SiteWise Edge yang diunduh digunakan dalam tutorial ini. Jalankan perintah berikut untuk mengekspor jalur lengkap sertifikat Anda, ganti `<absolute_path_to_certificate>` dengan jalur ke sertifikat:

```
export PATH_TO_CERTIFICATE='<absolute_path_to_certificate>'
```

Langkah 2: Dapatkan nama host gateway SiteWise Edge Anda

Note

Anda perlu [OpenSSL](#) diinstal untuk menjalankan perintah ini.

Untuk menyelesaikan tutorial, Anda memerlukan nama host gateway SiteWise Edge Anda. Untuk mendapatkan nama host gateway SiteWise Edge Anda, jalankan yang berikut ini, ganti `<sitewise_gateway_ip>` dengan IP gateway SiteWise Edge:

```
openssl s_client -connect <sitewise_gateway_ip>:443 </dev/null 2>/dev/null | grep -Po  
'CN = \K.*'| head -1
```

Jalankan perintah berikut untuk mengekspor nama host untuk digunakan nanti, ganti `<your_edge_gateway_hostname>` dengan nama host gateway SiteWise Edge Anda:

```
export GATEWAY_HOSTNAME='<your_edge_gateway_hostname>'
```

Langkah 3: Dapatkan kredensial sementara untuk gateway Edge Anda SiteWise

Sekarang setelah Anda memiliki sertifikat yang ditandatangani dan nama host gateway SiteWise Edge Anda, Anda perlu mendapatkan kredensial sementara sehingga Anda dapat menjalankan API di gateway. Anda bisa mendapatkan kredensial ini melalui AWS OpsHub for AWS IoT SiteWise atau langsung dari gateway SiteWise Edge menggunakan API.

⚠ Important

Kredensial kedaluwarsa setiap 4 jam, jadi Anda harus mendapatkan kredensialnya sebelum menggunakan API di gateway Edge Anda. SiteWise Jangan menyimpan kredensial selama lebih dari 4 jam.

Dapatkan kredensial sementara menggunakan for AWS OpsHubAWS IoT SiteWise

📘 Note

Anda memerlukan [AWS IoT SiteWiseaplikasi AWS OpsHub for](#) diinstal.

Untuk menggunakan AWS IoT SiteWise aplikasi AWS OpsHub untuk mendapatkan kredensial sementara Anda lakukan hal berikut:

1. Masuk ke aplikasi.
2. Pilih Pengaturan.
3. Untuk Otentikasi, pilih Salin kredensial.
4. Perluas opsi yang sesuai dengan lingkungan Anda dan pilih Salin.
5. Simpan kredensialnya untuk digunakan nanti.

Dapatkan kredensial sementara menggunakan SiteWise Edge gateway API

Untuk menggunakan SiteWise Edge gateway API untuk mendapatkan kredensial sementara, Anda dapat menggunakan skrip Python atau curl, pertama-tama Anda harus memiliki nama pengguna dan kata sandi untuk gateway Edge Anda. SiteWise Gateway SiteWise Edge menggunakan otentikasi dan otorisasi SiGv4. Untuk informasi selengkapnya tentang menambahkan pengguna, lihat [kumpulan pengguna LDAP atau Linux](#). Kredensial ini akan digunakan dalam langkah-langkah berikut untuk mendapatkan kredensial lokal di gateway SiteWise Edge Anda yang diperlukan untuk menggunakan API. AWS IoT SiteWise

Python

Note

[Anda perlu menginstal urllib3 dan Python3.](#)

Untuk mendapatkan kredensial menggunakan Python

1. Buat file bernama `get_credentials.py` dan salin kode berikut ke dalamnya.

```
'''
The following demonstrates how to get the credentials from the SiteWise Edge
gateway. You will need to add local users or connect your system to LDAP/AD
https://docs.aws.amazon.com/iot-sitewise/latest/userguide/manage-gateways-
ggv2.html#create-user-pool

Example usage:
    python3 get_credentials.py -e https://<gateway_hostname> -c
    <path_to_certificate> -u '<gateway_username>' -p '<gateway_password>' -m
    '<method>'
'''
import urllib3
import json
import urllib.parse
import sys
import os
import getopt

'''
This function retrieves the AWS IoT SiteWise Edge gateway credentials.
'''
def get_credentials(endpoint, certificatePath, user, password, method):
    http = urllib3.PoolManager(cert_reqs='CERT_REQUIRED', ca_certs=
certificatePath)
    encoded_body = json.dumps({
        "username": user,
        "password": password,
        "authMechanism": method,
    })

    url = urllib.parse.urljoin(endpoint, "/authenticate")
```

```
response = http.request('POST', url,
                        headers={'Content-Type': 'application/json'},
                        body=encoded_body)

if response.status != 200:
    raise Exception(f'Failed to authenticate! Response status
{response.status}')

auth_data = json.loads(response.data.decode('utf-8'))

accessKeyId = auth_data["accessKeyId"]
secretAccessKey = auth_data["secretAccessKey"]
sessionToken = auth_data["sessionToken"]
region = "edge"

return accessKeyId, secretAccessKey, sessionToken, region

def print_help():
    print('Usage:')
    print(f'{os.path.basename(__file__)} -e <endpoint> -c <path/to/certificate>
-u <user> -p <password> -m <method> -a <alias>')
    print('')
    print('-e, --endpoint    edge gateway endpoint. Usually the Edge gateway
hostname.')
    print('-c, --cert_path path to downloaded gateway certificate')
    print('-u, --user        Edge user')
    print('-p, --password   Edge password')
    print('-m, --method    (Optional) Authentication method (linux, winnt,
ldap), default is linux')
    sys.exit()

def parse_args(argv):
    endpoint = ""
    certificatePath = None
    user = None
    password = None
    method = "linux"

    try:
        opts, args = getopt.getopt(argv, "he:c:u:p:m:",
["endpoint=", "cert_path=", "user=", "password=", "method="])
    except getopt.GetoptError:
        print_help()
```



```
for opt, arg in opts:
    if opt == '-h':
        print_help()
    elif opt in ("-e", "--endpoint"):
        endpoint = arg
    elif opt in ("-u", "--user"):
        user = arg
    elif opt in ("-p", "--password"):
        password = arg
    elif opt in ("-m", "--method"):
        method = arg.lower()
    elif opt in ("-c", "--cert_path"):
        certificatePath = arg

if method not in ['ldap', 'linux', 'winnt']:
    print("not valid method parameter, required are ldap, linux, winnt")
    print_help()

if (user == None or password == None):
    print("To authenticate against edge user, password have to be passed
together, and the region has to be set to 'edge'")
    print_help()

if(endpoint == ""):
    print("You must provide a valid and reachable gateway hostname")
    print_help()

return endpoint,certificatePath, user, password, method

def main(argv):
    # get the command line args
    endpoint, certificatePath, user, password, method = parse_args(argv)

    accessKeyId, secretAccessKey, sessionToken, region=get_credentials(endpoint,
certificatePath, user, password, method)

    print("Copy and paste the following credentials into the shell, they are
valid for 4 hours:")
    print(f"export AWS_ACCESS_KEY_ID={accessKeyId}")
    print(f"export AWS_SECRET_ACCESS_KEY={secretAccessKey}")
    print(f"export AWS_SESSION_TOKEN={sessionToken}")
    print(f"export AWS_REGION={region}")
```

```
print()
```

```
if __name__ == "__main__":
    main(sys.argv[1:])
```

2. Jalankan `get_credentials.py` dari terminal menggantikan `<gateway_username>` dan `<gateway_password>` dengan kredensial yang Anda buat.

```
python3 get_credentials.py -e https://$GATEWAY_HOSTNAME -c $PATH_TO_CERTIFICATE
-u '<gateway_username>' -p '<gateway_password>' -m 'linux'
```

curl

Note

Anda perlu menginstal [curl](#).

Untuk mendapatkan kredensial menggunakan curl

1. Jalankan perintah berikut dari penggantian terminal `<gateway_username>` dan `<gateway_password>` dengan kredensial yang Anda buat.

```
curl --cacert $PATH_TO_CERTIFICATE --location \
-X POST https://$GATEWAY_HOSTNAME:443/authenticate \
--header 'Content-Type: application/json' \
--data-raw '{
  "username": "<gateway_username>",
  "password": "<gateway_password>",
  "authMechanism": "linux"
}'
```

Responsnya akan terlihat seperti berikut:

```
{
  "username": "sweuser",
  "accessKeyId": "<accessKeyId>",
```

```
"secretAccessKey": "<secretAccessKey>",
"sessionToken": "<sessionToken>",
"sessionExpiryTime": "2022-11-17T04:51:40.927095Z",
"authMechanism": "linux",
"role": "edge-user"
}
```

2. Jalankan perintah berikut dari terminal Anda.

```
export AWS_ACCESS_KEY_ID=<accessKeyId>
export AWS_SECRET_ACCESS_KEY=<secretAccessKey>
export AWS_SESSION_TOKEN=<sessionToken>
export AWS_REGION=edge
```

Langkah 4: Dapatkan daftar model aset di gateway SiteWise Edge

Sekarang setelah Anda memiliki sertifikat yang ditandatangani, nama host gateway SiteWise Edge Anda, dan kredensial sementara untuk gateway SiteWise Edge Anda, Anda dapat menggunakan `ListAssetModels` API untuk mendapatkan daftar model aset di gateway Edge Anda SiteWise .

Python

Note

Anda perlu [Python3](#), [Boto3](#), dan diinstal. [BotoCore](#)

Untuk mendapatkan daftar model aset menggunakan Python

1. Buat file bernama `list_asset_model.py` dan salin kode berikut ke dalamnya.

```
import json
import boto3
import botocore
import os

# create the client using the credentials
client = boto3.client("iotsitewise",
    endpoint_url= "https://" + os.getenv("GATEWAY_HOSTNAME"),
    region_name=os.getenv("AWS_REGION"),
    aws_access_key_id=os.getenv("AWS_ACCESS_KEY_ID"),
```

```
aws_secret_access_key=os.getenv("AWS_SECRET_ACCESS_KEY"),
aws_session_token=os.getenv("AWS_SESSION_TOKEN"),
verify=os.getenv("PATH_TO_CERTIFICATE"),
config=botocore.config.Config(inject_host_prefix=False))

# call the api using local credentials
response = client.list_asset_models()
print(response)
```

2. Jalankan `list_asset_model.py` dari terminal.

```
python3 list_asset_model.py
```

curl

Note

Anda perlu menginstal [curl](#).

Untuk mendapatkan daftar model aset menggunakan curl

Jalankan perintah berikut dari terminal.

```
curl \
  --request GET https://$GATEWAY_HOSTNAME:443/asset-models \
  --cacert $PATH_TO_CERTIFICATE \
  --aws-sigv4 "aws:amz:edge:iotsitewise" \
  --user "$AWS_ACCESS_KEY_ID:$AWS_SECRET_ACCESS_KEY" \
  -H "x-amz-security-token:$AWS_SESSION_TOKEN"
```

Responsnya akan terlihat seperti berikut:

```
{
  "assetModelSummaries": [
    {
      "arn": "arn:aws:iotsitewise:{region}:{account-id}:asset-model/{asset-
model-id}",
      "creationDate": 1.669245291E9,
      "description": "This is a small example asset model",
```

```
    "id": "{asset-model-id}",
    "lastUpdateDate": 1.669249038E9,
    "name": "Some Metrics Model",
    "status": {
      "error": null,
      "state": "ACTIVE"
    }
  },
  .
  .
  .
],
"nextToken": null
}
```

Cadangkan dan pulihkan SiteWise gateway Edge

Topik ini mencakup cara memulihkan gateway SiteWise Edge dan mencadangkan data metrik Anda. Jika Anda mengalami masalah dengan gateway SiteWise Edge yang rusak pada mesin yang sama dan perlu memecahkan masalah, baca AWS IoT SiteWise dokumentasi [SiteWise Pemecahan Masalah gateway Edge](#).

Note

Panduan yang dibahas dalam topik ini adalah untuk gateway SiteWise Edge yang diinstal pada AWS IoT Greengrass V2 versi 2.1.0 atau lebih tinggi.

Backup harian data metrik

Membuat cadangan adalah penting, jika Anda ingin mentransfer atau mengembalikan data pada mesin baru. Mencadangkan data Anda sangat mengurangi risiko hilangnya data operasi selama proses transfer atau pemulihan.

Jalur folder influxdb adalah sebagai berikut:

Linux

```
/greengrass/v2/work/aws.iot.SiteWiseEdgeProcessor/influxdb
```

Windows

```
C:\greengrass\v2\work\aws.iot.SiteWiseEdgeProcessor\influxdb
```

Kami menyarankan Anda membuat cadangan seluruh folder dengan semua yang ada di bawahnya.

Kami menyarankan Anda secara berkala mencadangkan data metrik Anda dari 1.0 SiteWise Edge ke hard drive eksternal atau ke AWS cloud.

Kembalikan gateway SiteWise Edge

Gunakan prosedur berikut untuk memulihkan gateway SiteWise Edge:

1. Gunakan skrip instalasi yang diunduh saat Anda membuat gateway SiteWise Edge untuk memulihkan gateway SiteWise Edge pada mesin baru. Baca [Menginstal perangkat lunak gateway SiteWise Edge pada prosedur perangkat lokal Anda](#) untuk mengatur gateway SiteWise Edge.

Jika Anda kehilangan atau tidak dapat menemukan skrip instalasi, silakan hubungi [AWS Customer Support](#).

2. Setelah gateway SiteWise Edge diinstal, masuk ke [AWS IoT Greengrass konsol](#).
3. Untuk menerapkan ulang komponen, navigasikan ke Kelola lalu di bawah AWS IoT Greengrass perangkat pilih Perangkat inti.
4. Di tabel perangkat AWS IoT Greengrass inti pilih perangkat inti yang sesuai dengan gateway SiteWise Edge Anda.
5. Setelah berada di halaman perangkat, buka tab Deployment dan pilih ID Deployment Anda, ini akan membuka halaman Deployment dengan ID yang Anda pilih.

The screenshot shows the AWS IoT SiteWise console interface. On the left is a navigation menu with categories like Monitor, Connect, Test, Manage, and Security. The main content area displays the details for a core device named 'OriginalGatewayGreengrassCoreDevice-nu7HuEvoH'. The 'Deployments' tab is selected and highlighted with a red box. Below the tab, there is a table with one deployment entry:

Deployment ID	Name	Target	Status on this device	Status reported
5b3cbd52-607f-4c2c-bc8a-708298e4925a	-	OriginalGatewayGreengrassCoreDevice-nu7HuEvoH	Succeeded	4 days ago

- Setelah Anda berada di halaman Deployment, di kanan atas tekan tombol Tindakan, dan pilih opsi Revisasi. untuk memulai penerapan baru. Konfigurasikan penerapan. Jika Anda ingin mempertahankan penerapan apa adanya, lewati ke Review and Deploy.
- Tunggu Status Deployment menjadi Completed.

Note

Ini juga akan memakan waktu beberapa menit untuk semua komponen di SiteWise Edge untuk sepenuhnya setup dan berjalan.

Kembalikan AWS IoT SiteWise data

Gunakan prosedur berikut untuk memulihkan data pada mesin baru.

- Salin influxdb folder ke mesin baru.
- Hentikan SiteWise EdgeProcessor komponen, dengan menjalankan perintah berikut di terminal Anda:

Linux

```
sudo /greengrass/v2/bin/greengrass-cli component stop -n  
aws.iot.SiteWiseEdgeProcessor
```

Windows

```
C:\greengrass\v2\bin\greengrass-cli component stop -n  
aws.iot.SiteWiseEdgeProcesso
```

3. Temukan jalur tempat Anda mencadangkan data, dan jalankan perintah berikut:

Linux

```
sudo yes | sudo cp -rf <influxdb_backup_path> /greengrass/v2/work/  
aws.iot.SiteWiseEdgeProcessor/influxdb
```

PowerShell

```
Copy-Item -Recurse -Force <influxdb_backup_path>\* C:\greengrass  
\v2\work\aws.iot.SiteWiseEdgeProcessor\
```

Windows

```
robocopy <influxdb_backup_path> C:\greengrass\v2\work  
\aws.iot.SiteWiseEdgeProcessor\ /E
```

4. Mulai ulang SiteWiseEdgeProcessor komponen:

Linux

```
sudo /greengrass/v2/bin/greengrass-cli component restart -n  
aws.iot.SiteWiseEdgeProcessor
```

Windows

```
C:\greengrass\v2\bin\greengrass-cli component restart -n  
aws.iot.SiteWiseEdgeProcessor
```

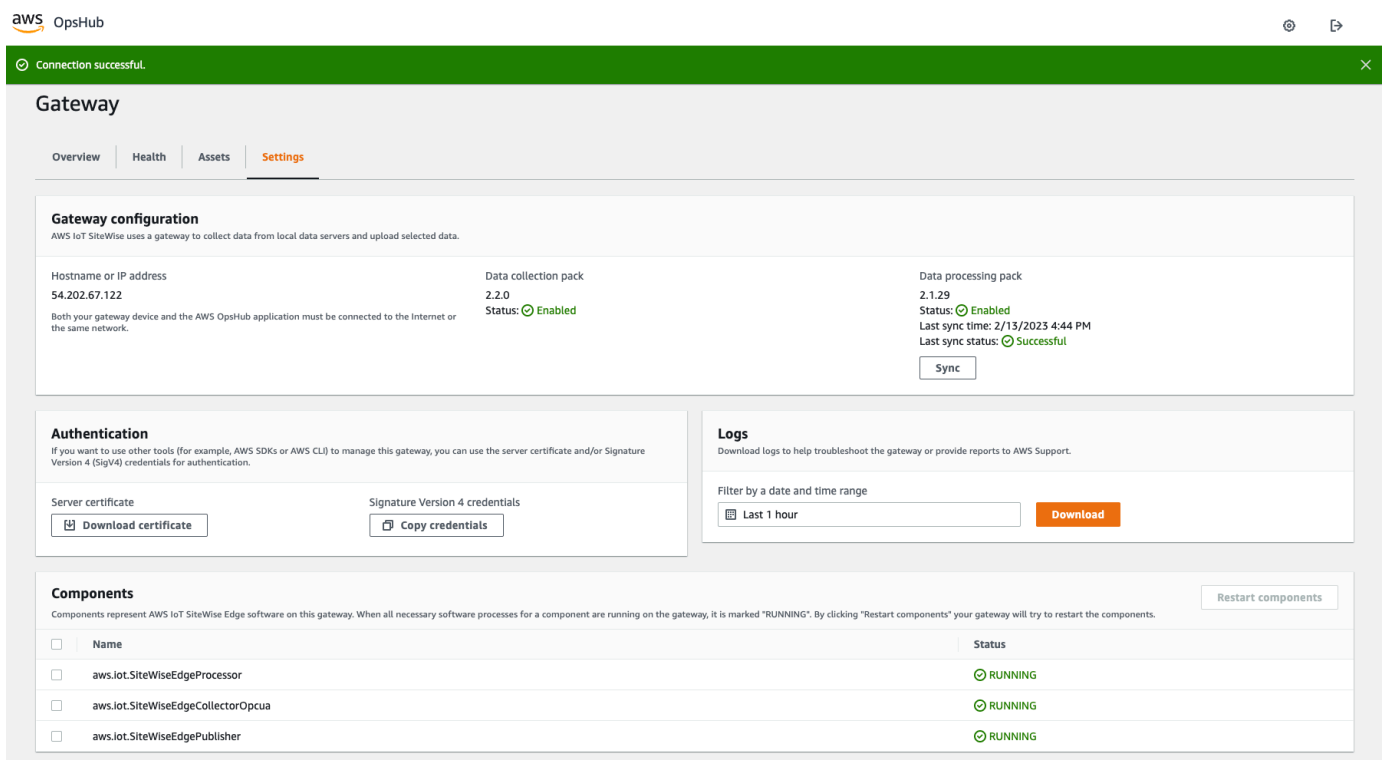
Validasi backup dan restorasi yang berhasil

Gunakan prosedur ini untuk memvalidasi data cadangan dan SiteWise restorasi gateway Edge Anda.

Note

Prosedur ini mengharuskan Anda menginstal AWS OpsHub untuk AWS IoT SiteWise. Untuk informasi selengkapnya, lihat, [Mengelola gateway SiteWise Edge menggunakan AWS OpsHub](#) for. AWS IoT SiteWise

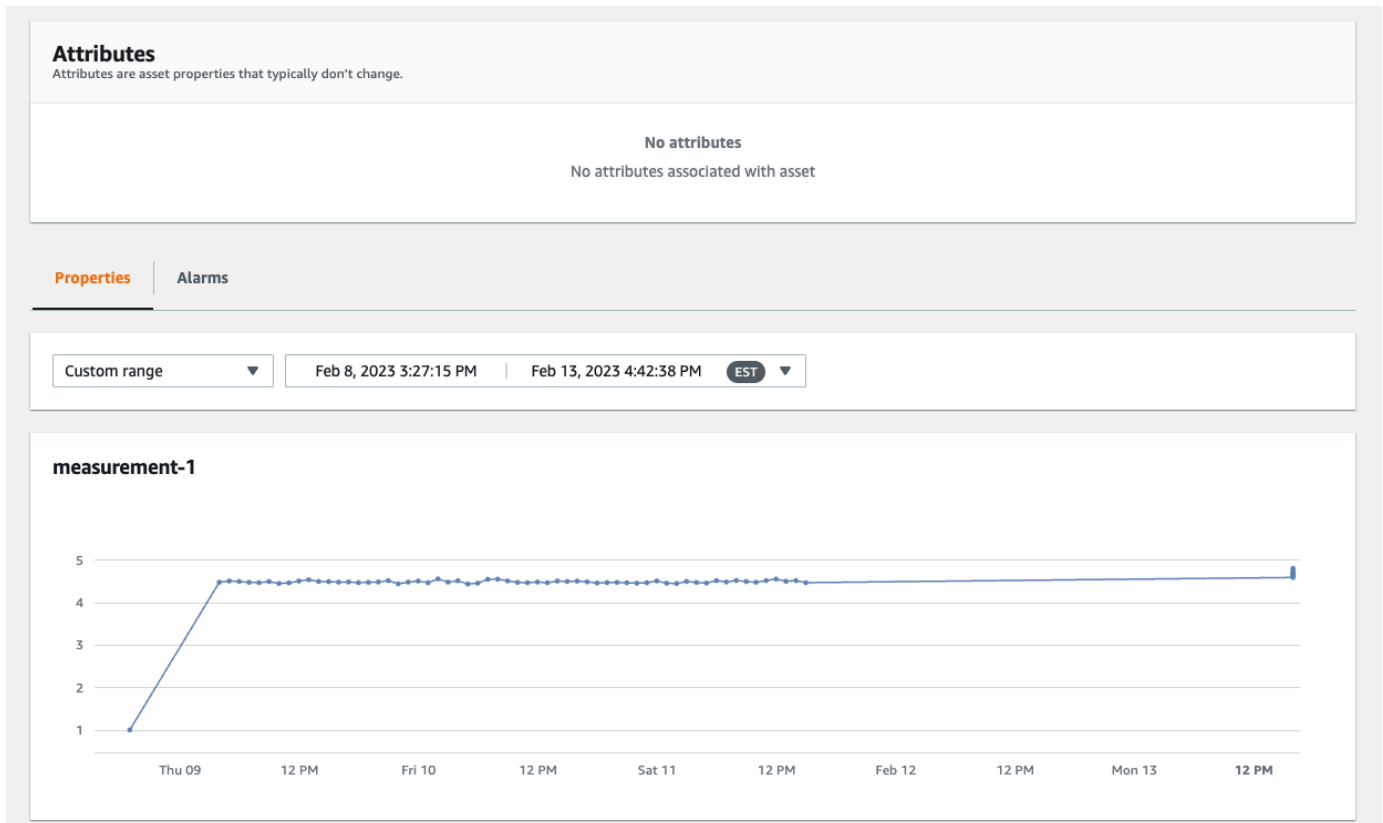
1. Terbuka AWS OpsHub untuk AWS IoT SiteWise.
2. Pada halaman Pengaturan SiteWise Edge Gateway, periksa status setiap komponen yang tercantum dalam tabel Komponen. Verifikasi bahwa warna status berwarna hijau dan pembacaan menampilkan RUNNING.



The screenshot displays the AWS OpsHub interface for a Gateway. At the top, there is a green notification bar that says "Connection successful." Below this, the "Gateway" section is active, with tabs for Overview, Health, Assets, and Settings. The "Gateway configuration" section includes fields for Hostname or IP address (54.202.67.122), Data collection pack (2.2.0, Status: Enabled), and Data processing pack (2.1.29, Status: Enabled). A "Sync" button is present. The "Authentication" section offers "Download certificate" and "Copy credentials" buttons. The "Logs" section has a "Download" button and a filter for "Last 1 hour". The "Components" section features a "Restart components" button and a table with the following data:

Name	Status
aws.iot.SiteWiseEdgeProcessor	RUNNING
aws.iot.SiteWiseEdgeCollectorOpcua	RUNNING
aws.iot.SiteWiseEdgePublisher	RUNNING

3. Validasi data masa lalu Anda di dasbor portal untuk memeriksa apakah data sebelumnya dan data baru keduanya diatur dengan benar. Akan ada downtime antara data masa lalu dan baru. Anda harus keculi untuk melihat durasi di mana tidak ada titik data yang dikumpulkan.



[Jika Anda mengalami masalah dengan mendaftarkan atau memulihkan gateway SiteWise Edge, lihat topik pemecahan masalah berikut Memecahkan masalah gateway Edge. AWS IoT SiteWise](#)

Menyiapkan gateway SiteWise Edge (AWS IoT Greengrass Version 1)

Note

SiteWise Gateway tepi yang berjalan hanya AWS IoT Greengrass V1 tersedia jika Anda mulai menggunakan fitur ini sebelum 29 Juli 2021. Jika tidak, Anda [mengatur gateway SiteWise Edge yang berjalan](#). AWS IoT Greengrass V2

Anda dapat mengirim data industri AWS IoT SiteWise menggunakan gateway SiteWise Edge untuk mengunggah data dari peralatan industri. Gateway SiteWise Edge berfungsi sebagai perantara antara AWS IoT SiteWise dan peralatan industri data Anda. AWS IoT SiteWise menyediakan AWS IoT Greengrass komponen yang dapat Anda terapkan di perangkat apa pun yang dapat dijalankan

AWS IoT Greengrass untuk menyiapkan gateway SiteWise Edge. AWS IoT SiteWise mendukung penautan dengan protokol server [OPC-UA](#).

Jika Anda memiliki gateway AWS IoT SiteWise Edge yang berjalan AWS IoT Greengrass V1, Anda dapat meningkatkan gateway SiteWise Edge Anda ke AWS IoT Greengrass V2. Untuk informasi selengkapnya, lihat [Petunjuk untuk meningkatkan gateway SiteWise Edge](#) dari ke AWS IoT Greengrass V1 AWS IoT Greengrass V2.

Topik

- [Memilih perangkat gateway AWS IoT Greengrass V1 SiteWise Edge](#)
- [Mengkonfigurasi gateway AWS IoT Greengrass V1 SiteWise Edge](#)
- [Mengkonfigurasi sumber data di gateway AWS IoT Greengrass V1 SiteWise Edge](#)

Memilih perangkat gateway AWS IoT Greengrass V1 SiteWise Edge

Pilih perangkat lokal yang paling sesuai dengan operasi industri Anda. Anda dapat mengonfigurasi gateway SiteWise Edge pada perangkat apa pun yang dapat dijalankan AWS IoT Greengrass. Semua perangkat lokal harus memenuhi persyaratan berikut:

- Mendukung perangkat lunak AWS IoT Greengrass Core v1.10.2 atau yang lebih baru. Untuk informasi selengkapnya, lihat [Platform dan persyaratan yang didukung](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.
- Memiliki setidaknya 4 GB RAM.
- Memiliki setidaknya 10 GB ruang disk kosong.
- Mendukung mesin virtual Java 8 (JVM).

Jika Anda berencana untuk memproses data di edge AWS IoT SiteWise, perangkat lokal Anda juga harus memenuhi persyaratan berikut:

- Memiliki prosesor quad-core x86 64 bit.
- Memiliki setidaknya 16 GB RAM.
- Memiliki setidaknya 32 GB untuk RAM jika menggunakan Windows.
- Memiliki setidaknya 256 GB ruang disk kosong.

Ruang disk yang diperlukan untuk caching data untuk konektivitas internet intermiten tergantung pada faktor-faktor berikut:

- Jumlah aliran data yang diunggah
- Titik data per aliran data per detik
- Ukuran setiap titik data
- Kecepatan komunikasi
- Waktu henti jaringan yang diharapkan

Kapasitas komputasi yang diperlukan untuk melakukan polling dan mengunggah data tergantung pada faktor-faktor berikut:

- Jumlah aliran data yang diunggah
- Titik data per aliran data per detik

Mengkonfigurasi gateway AWS IoT Greengrass V1 SiteWise Edge

Gerbang AWS IoT SiteWise Edge berfungsi sebagai perantara antara peralatan industri Anda dan AWS IoT SiteWise. Anda dapat menerapkan perangkat lunak gateway SiteWise Edge di perangkat apa pun yang dapat dijalankan AWS IoT Greengrass. Untuk informasi selengkapnya, lihat [Memilih perangkat gateway AWS IoT Greengrass V1 SiteWise Edge](#).

Anda dapat mengaktifkan AWS IoT SiteWise untuk memproses data secara lokal di perangkat edge Anda dengan menggunakan paket pemrosesan data di gateway SiteWise Edge Anda. Anda melakukan ini ketika Anda menambahkan gateway SiteWise Edge Anda ke AWS IoT SiteWise. Untuk informasi selengkapnya tentang pemrosesan data di edge, lihat [the section called “Mengaktifkan pemrosesan data tepi”](#).

Note

Kami menyarankan Anda menyelesaikan langkah-langkah berikut dengan seseorang yang memiliki akses administratif TI ke jaringan lokal dan perusahaan Anda. Langkah-langkah ini mungkin memerlukan seseorang dengan pengetahuan tentang peralatan industri Anda dan otoritas untuk mengkonfigurasi pengaturan firewall.

Topik

- [Menyiapkan lingkungan gateway SiteWise Edge](#)
- [Membuat kebijakan dan peran IAM](#)
- [Mengkonfigurasi grup AWS IoT Greengrass](#)
- [Mengkonfigurasi konektor AWS IoT SiteWise](#)
- [Menambahkan gateway SiteWise Edge ke AWS IoT SiteWise](#)

Menyiapkan lingkungan gateway SiteWise Edge

Dalam prosedur ini, Anda menginstal AWS IoT Greengrass dan mengonfigurasi gateway SiteWise Edge Anda untuk digunakan AWS IoT SiteWise.

Note

Bagian ini mencakup instruksi untuk menginstal paket menggunakan apt perintah. Ini berlaku untuk sistem yang menjalankan Ubuntu atau serupa. Jika Anda tidak menggunakan sistem serupa, lihat dokumentasi untuk distribusi Anda dan gunakan penginstal paket yang direkomendasikan.

Untuk mengatur gateway SiteWise Edge

1. Jika sesuai, ubah pengaturan [BIOS](#) gateway SiteWise Edge sebagai berikut.
 - a. Pastikan gateway SiteWise Edge secara otomatis restart setelah potensi kegagalan daya, jika berlaku.
 - b. Pastikan gateway SiteWise Edge tidak akan hibernasi atau tidur, jika ada.
2. Pastikan gateway SiteWise Edge terhubung ke internet.
3. (Opsional) Untuk menggunakan gateway SiteWise Edge tanpa mouse, keyboard, dan monitor, lakukan langkah-langkah berikut untuk mengatur ssh pada gateway SiteWise Edge:
 - a. Jika Anda belum menginstal paket SSH, jalankan perintah berikut.

```
sudo apt install ssh
```

- b. Jalankan perintah berikut.

```
service ssh status
```

- c. Cari Active: `active` (`running`) di output untuk mengonfirmasi bahwa server SSH sedang berjalan,
- d. Tekan Q untuk keluar.

Jalankan perintah berikut untuk menggunakan SSH untuk terhubung ke gateway SiteWise Edge dari komputer lain. Ganti *nama* pengguna dengan login pengguna dan *IP* dengan alamat IP gateway SiteWise Edge.

```
ssh username@IP
```

Anda dapat menggunakan `-p port-number` argumen untuk terhubung ke port selain port default 22.

4. Unduh dan instal perangkat lunak AWS IoT Greengrass Core v1.10.2 atau yang lebih baru, dan buat AWS IoT Greengrass grup untuk gateway SiteWise Edge Anda. Untuk melakukannya, ikuti petunjuk di [Memulai AWS IoT Greengrass](#) di Panduan AWS IoT Greengrass Pengembang.

Kami menyarankan Anda menjalankan skrip [pengaturan AWS IoT Greengrass perangkat](#) untuk memulai dengan cepat. Jika Anda ingin meninjau AWS IoT Greengrass persyaratan dan proses lebih dekat, Anda dapat menelusuri langkah-langkah dalam [Modul 1](#) dan [Modul 2](#) untuk disiapkan AWS IoT Greengrass.

Important

Tinjau [AWS Wilayah](#) AWS IoT SiteWise yang didukung. Ketika Anda memilih Wilayah untuk AWS IoT Greengrass, pastikan bahwa Wilayah juga mendukung AWS IoT SiteWise. Jika tidak, Anda tidak dapat menghubungkan gateway SiteWise Edge Anda ke AWS IoT SiteWise.

Sebelum Anda melanjutkan ke langkah berikutnya, Anda harus menginstal perangkat lunak AWS IoT Greengrass Core di gateway SiteWise Edge Anda.

5. Jalankan perintah berikut untuk menginstal Java 8.

```
sudo apt update
sudo apt install openjdk-8-jre
```

Perangkat lunak gateway SiteWise Edge yang Anda instal nanti dalam panduan ini menggunakan runtime Java 8.

6. Jalankan perintah berikut untuk memverifikasi bahwa Java berhasil diinstal.

```
java -version
```

7. Perangkat lunak AWS IoT Greengrass Core mengasumsikan java8 direktori. Jalankan perintah berikut untuk menautkan instalasi Java Anda ke java8 direktori itu.

```
sudo ln -s /usr/bin/java /usr/bin/java8
```

8. Jalankan perintah berikut untuk membuat direktori `/var/sitewise` data dan memberikan `ggc_user` izin untuk direktori itu. AWS IoT SiteWise menyimpan data dalam direktori ini. Anda membuat `ggc_user` ketika Anda mengatur AWS IoT Greengrass sebelumnya dalam prosedur ini.

```
sudo mkdir /var/sitewise
sudo chown ggc_user /var/sitewise
sudo chmod 700 /var/sitewise
```

`/var/sitewise` ini adalah direktori default yang AWS IoT SiteWise menggunakan. Anda dapat menyesuaikan jalur direktori (misalnya, ganti `/var/sitewise` dengan `/var/custom/path/`), tetapi melakukannya memerlukan langkah tambahan setelah gateway SiteWise Edge dibuat. Untuk informasi lebih lanjut, lihat langkah 6 di [Mengkonfigurasi konektor AWS IoT SiteWise](#).

9. Jika diperlukan, minta administrator TI Anda untuk menambahkan titik akhir dan port berikut ke daftar izin jaringan lokal Anda:

- Pelabuhan: 443, 8443, dan 8883

Important

Anda dapat mengonfigurasi AWS IoT Greengrass Core untuk hanya menggunakan port 443 untuk semua komunikasi jaringan. Untuk informasi selengkapnya, lihat [Connect on port 443 atau melalui proxy jaringan](#) di Panduan AWS IoT Greengrass Pengembang.

- Alamat IP gateway SiteWise Edge Anda (port 443). Untuk mendapatkan alamat IP, jalankan `ifconfig` perintah `ip address` or dan catat `inet` nilainya (misalnya, `203.0.113.0`).
- Titik akhir AWS IoT SiteWise data: `data.iotsitewise.region.amazonaws.com` (port 443).
- AWS Titik akhir berikut yang digunakan gateway SiteWise Edge. Anda dapat menemukan ini di `/greengrass-root/config/config.json` file. Ganti `greengrass-root` dengan `root` instalasi Anda. AWS IoT Greengrass
 - GGHost: `greengrass-ats.iot.region.amazonaws.com` (port 443, 8443, dan 8883).
 - `IoTHostprefix-ats.iot.region.amazonaws.com`: (port 443, 8443, dan 8883).

Untuk informasi lebih lanjut, lihat [AWS IoT Greengrass kuota dan titik akhir](#).

10. Jika perangkat lunak AWS IoT Greengrass Core belum berjalan, jalankan perintah berikut untuk memulai perangkat lunak AWS IoT Greengrass Core. Ganti `greengrass-root` dengan `root` instalasi Anda. AWS IoT Greengrass `Greengrass-root` default adalah `/greengrass`

```
cd /greengrass-root/ggc/core
sudo ./greengrassd start
```

Anda akan melihat pesan ini: `Greengrass successfully started with PID: some-PID-number`

11. Konfigurasi perangkat lunak AWS IoT Greengrass Core untuk memulai secara otomatis saat gateway SiteWise Edge Anda menyala. Konsultasikan dokumentasi untuk sistem operasi gateway SiteWise Edge Anda.

Membuat kebijakan dan peran IAM

Anda harus membuat kebijakan dan peran AWS Identity and Access Management (IAM) untuk memungkinkan gateway SiteWise Edge mengakses AWS IoT SiteWise atas nama Anda.

Untuk membuat kebijakan dan peran IAM

1. Arahkan ke [konsol IAM](#).
2. Di panel navigasi, pilih Kebijakan dan kemudian pilih Buat kebijakan.

The screenshot shows the AWS IAM console interface. In the top navigation bar, the 'Create policy' button is highlighted with a red box. On the left-hand sidebar, the 'Policies' menu item is also highlighted with a red box. The main content area displays a table of existing policies with columns for 'Policy name', 'Type', and 'Used as'. The table lists several AWS managed policies, such as 'AdministratorAccess', 'AlexaForBusinessDeviceSetup', and 'AmazonAPIGatewayAdministrator'.

Policy name	Type	Used as
AdministratorAccess	Job function	Permissions poli
AlexaForBusinessDeviceSetup	AWS managed	None
AlexaForBusinessFullAccess	AWS managed	None
AlexaForBusinessGatewayExecution	AWS managed	None
AlexaForBusinessReadOnlyAccess	AWS managed	None
AmazonAPIGatewayAdministrator	AWS managed	None
AmazonAPIGatewayInvokeFullAccess	AWS managed	None
AmazonAPIGatewayPushToCloudWatchLogs	AWS managed	None

3. Pada tab JSON, hapus konten saat ini dari bidang kebijakan, dan tempelkan kebijakan berikut ke dalam bidang.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iotsitewise:BatchPutAssetPropertyValue",
      "Resource": "*"
    }
  ]
}
```

Note

Untuk meningkatkan keamanan, Anda dapat menentukan jalur hierarki AWS IoT SiteWise aset di Condition properti. Contoh berikut adalah kebijakan kepercayaan yang menentukan jalur hierarki aset.

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```
{
  "Effect": "Allow",
  "Action": "iotsitewise:BatchPutAssetPropertyValue",
  "Resource": "*",
  "Condition": {
    "StringLike": {
      "iotsitewise:assetHierarchyPath": [
        "/root node asset ID",
        "/root node asset ID/*"
      ]
    }
  }
}
```

4. Pilih Tinjau kebijakan.
5. Masukkan nama dan deskripsi untuk kebijakan tersebut, lalu pilih Buat kebijakan.
6. Di panel navigasi, pilih Peran, lalu pilih Buat peran.

Roles

What are IAM roles?

IAM roles are a secure way to grant permissions to entities that you trust. Examples of entities include the following:

- IAM user in another account
- Application code running on an EC2 instance that needs to perform actions on AWS resources
- An AWS service that needs to act on resources in your account to provide its features
- Users from a corporate directory who use identity federation with SAML

IAM roles issue keys that are valid for short durations, making them a more secure way to grant access.

Additional resources:

- [IAM Roles FAQ](#)
- [IAM Roles Documentation](#)
- [Tutorial: Setting Up Cross Account Access](#)
- [Common Scenarios for Roles](#)

Create role Delete role

Search


Role name	Description
<input type="checkbox"/> Admin	
<input type="checkbox"/> AwsSecurityAudit	

7. Di bawah Pilih jenis entitas terpercaya, pilih AWS layanan. Di bawah Pilih layanan yang akan menggunakan peran, pilih Greengrass sebagai layanan yang akan menggunakan peran, lalu pilih Berikutnya: Izin.


Create role

1 2 3 4


Select type of trusted entity




AWS service
EC2, Lambda and others



Another AWS account
Belonging to you or 3rd party



Web identity
Cognito or any OpenID provider



SAML 2.0 federation
Your corporate directory

Allows AWS services to perform actions on your behalf. [Learn more](#)

Choose the service that will use this role

EC2

Allows EC2 instances to call AWS services on your behalf.

Lambda

Allows Lambda functions to call AWS services on your behalf.

API Gateway	CodeBuild	EC2 - Fleet	Inspector	Redshift
AWS Support	CodeDeploy	EKS	IoT	Rekognition
AppSync	Config	EMR	Kinesis	S3
Application Auto Scaling	Connect	ElasticCache	Lambda	SMS
Application Discovery Service	DMS	Elastic Beanstalk	Lex	SNS
Auto Scaling	Data Lifecycle Manager	Elastic Container Service	Machine Learning	SWF
Batch	Data Pipeline	Elastic Transcoder	Macie	SageMaker
CloudFormation	DeepLens	ElasticLoadBalancing	MediaConvert	Service Catalog
CloudHSM	Directory Service	Glue	OpsWorks	Step Functions
CloudTrail	DynamoDB	Greengrass	RAM	Storage Gateway
CloudWatch Events	EC2	GuardDuty	RDS	Trusted Advisor

Select your use case

* Required

Cancel

Next: Permissions

8. Cari kebijakan yang Anda buat, pilih kotak centang, lalu pilih Berikutnya: Tag.

Create role

1 2 3 4

▼ Attach permissions policies

Choose one or more policies to attach to your new role.

Create policy ↻

Filter policies Showing 1 result

	Policy name ▼	Used as	Description
<input checked="" type="checkbox"/>	SiteWiseDemo	None	Policy for the SiteWise demo.

▶ Set permissions boundary

* Required

Cancel

Previous

Next: Tags

- (Opsional) Tambahkan tag ke peran Anda, lalu pilih Berikutnya: Tinjau.
- Masukkan nama dan deskripsi untuk peran tersebut, lalu pilih Buat peran.

Create role



Review

Provide the required information below and review this role before you create it.

Role name* SiteWiseDemo

Use alphanumeric and '+,=,@-_' characters. Maximum 64 characters.

Role description Allows Greengrass to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+,=,@-_' characters.

Trusted entities AWS service: greengrass.amazonaws.com

Policies SiteWiseDemo [↗](#)

Permissions boundary Permissions boundary is not set

No tags were added.

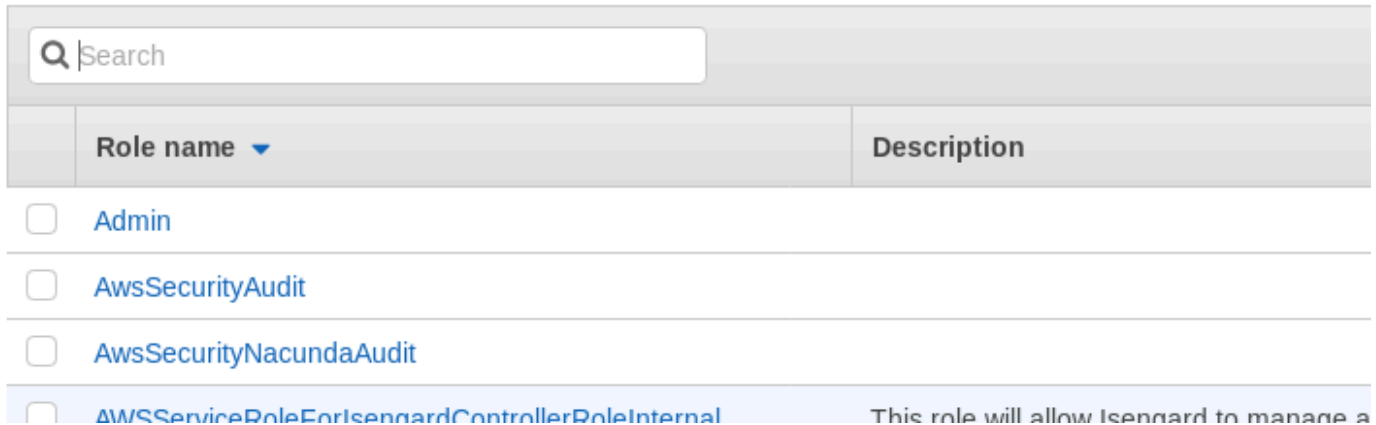
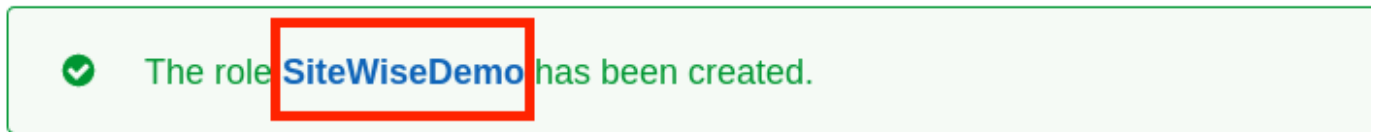
* Required

Cancel

Previous

Create role

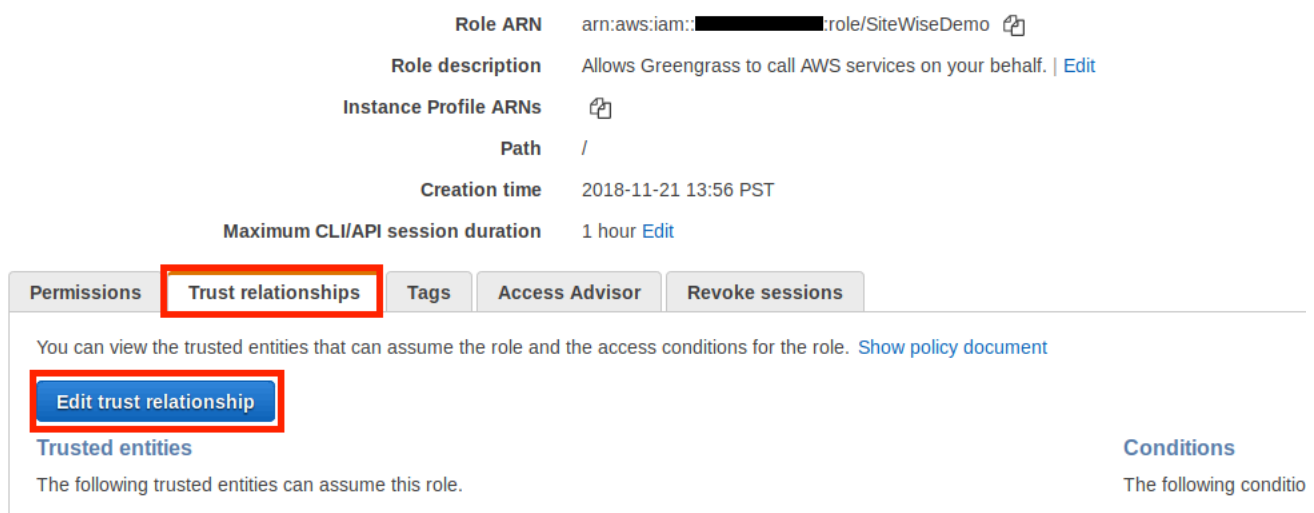
11. Di spanduk hijau, pilih tautan ke peran baru Anda. Anda juga dapat menggunakan bidang pencarian untuk menemukan peran.



12. Pilih tab Hubungan kepercayaan, dan kemudian pilih Ubah hubungan kepercayaan.

Roles > SiteWiseDemo

Summary



13. Ganti konten saat ini dari bidang kebijakan dengan yang berikut, lalu pilih Perbarui Kebijakan Kepercayaan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

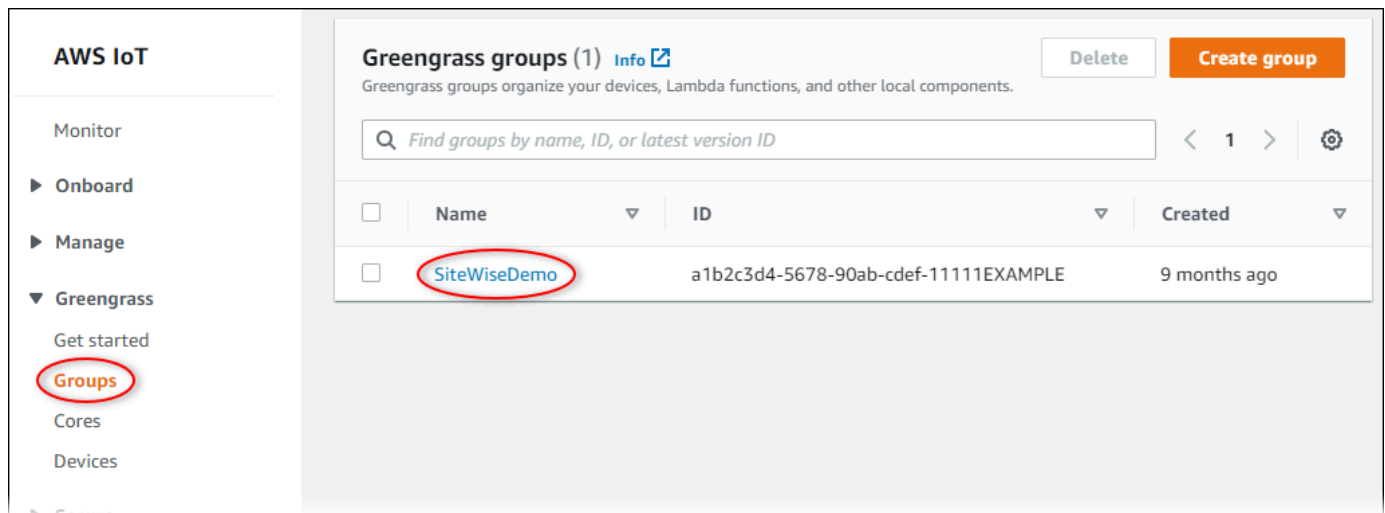
```
    "Principal": {
      "Service": "greengrass.amazonaws.com"
    },
    "Action": "sts:AssumeRole"
  }
]
```

Mengkonfigurasi grup AWS IoT Greengrass

Untuk melampirkan peran IAM ke grup dan mengaktifkan manajer aliran

1. Navigasikan ke [konsol AWS IoT Greengrass](#) tersebut.
2. Di panel navigasi kiri, di bawah Greengrass, pilih Grup, lalu pilih grup yang Anda buat.

[Menyiapkan lingkungan gateway SiteWise Edge](#)



The screenshot shows the AWS IoT Greengrass console interface. On the left, the navigation menu is visible with 'Groups' highlighted in red. The main content area displays 'Greengrass groups (1)' with a search bar and a table of groups. The table has columns for 'Name', 'ID', and 'Created'. One group is listed with the name 'SiteWiseDemo' (circled in red), ID 'a1b2c3d4-5678-90ab-cdef-11111EXAMPLE', and 'Created' '9 months ago'.

	Name	ID	Created
<input type="checkbox"/>	SiteWiseDemo	a1b2c3d4-5678-90ab-cdef-11111EXAMPLE	9 months ago

3. Pada panel navigasi kiri, pilih Pengaturan. Di bagian Peran Grup, pilih Tambahkan Peran.

GREENGRASS GROUP

SiteWiseDemo

Not deployed Actions ▾

- Deployments
- Subscriptions
- Cores
- Devices
- Lambdas
- Resources
- Connectors
- Tags
- Settings**

Group Role Add Role

No role has been attached to the SiteWiseDemo Group

Group ID

1ff7b6c9-06d9-46f5-9f3e-88894dc19b37

Certification authority (CA) and local connection configuration

Device certificate lifetime period
By changing this setting you control the period during which a Device can establish a communication with its Core. The next new period will be 7 days.

4. Pilih peran yang Anda buat [Membuat kebijakan dan peran IAM](#), lalu pilih Simpan.

Your Group's IAM Role

Adding an IAM Role to your Group establishes a trust relationship between your trusting account and the Core.

Select an IAM Role with a Greengrass Role Type

Search Role name

SiteWiseDemo

Cancel Back Save

5. Pada halaman Pengaturan, di bagian Manajer aliran, pilih Edit.

Stream manager adalah fitur AWS IoT Greengrass yang memungkinkan AWS IoT Greengrass Core Anda mengalirkan data ke AWS Cloud. SiteWise Gateway tepi mengharuskan manajer aliran diaktifkan. Untuk informasi selengkapnya, lihat [Mengelola aliran data pada AWS IoT Greengrass Inti](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

[Update default Lambda execution configuration](#)

Stream manager

[Edit](#)

Stream manager enables the Core to ingest and process data streams and export them to cloud targets. [Learn more](#)

Status

Disabled

CloudWatch logs configuration

[Edit](#)

- Pilih Aktifkan, lalu pilih Simpan.
- Di sudut kiri atas, pilih Layanan untuk mempersiapkan prosedur selanjutnya.

Mengkonfigurasi konektor AWS IoT SiteWise

Dalam prosedur ini, Anda mengonfigurasi AWS IoT SiteWise konektor pada grup Greengrass Anda. Komponen adalah modul bawaan yang mempercepat siklus hidup pengembangan untuk skenario common edge. Untuk informasi selengkapnya, lihat [AWS IoT Greengrass konektor](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

Untuk mengkonfigurasi AWS IoT SiteWise konektor

- Navigasikan ke [konsol AWS IoT Greengrass](#) tersebut.
- Di panel navigasi kiri, di bawah Greengrass, pilih Grup, lalu pilih grup yang Anda buat.

[Menyiapkan lingkungan gateway SiteWise Edge](#)

The screenshot shows the AWS IoT Greengrass console interface. On the left, the navigation pane is expanded to 'Greengrass', and the 'Groups' option is highlighted with a red circle. The main content area displays 'Greengrass groups (1)' with a search bar and a table of groups. The table has columns for 'Name', 'ID', and 'Created'. One group is listed with the name 'SiteWiseDemo' (circled in red), ID 'a1b2c3d4-5678-90ab-cdef-11111EXAMPLE', and 'Created' '9 months ago'. Buttons for 'Delete' and 'Create group' are visible at the top right.

<input type="checkbox"/>	Name	ID	Created
<input type="checkbox"/>	SiteWiseDemo	a1b2c3d4-5678-90ab-cdef-11111EXAMPLE	9 months ago

- Di halaman navigasi kiri, pilih Konektor. Pada halaman Konektor, pilih Tambahkan konektor.

GREENGRASS GROUP


SiteWiseDemo

Not deployed Actions ▾

- Deployments
- Subscriptions
- Cores
- Devices
- Lambdas
- Resources
- Connectors**
- Tags
- Settings

Connectors

Connectors are modules that provide built-in integration with services, protocols, or infrastructure. [Learn more](#)



Accelerate your development

Connectors make it easier to develop applications by providing built-in integration with services, protocols, or infrastructure. [Learn more](#)

Add a connector

4. Pilih IoT SiteWise dari daftar dan pilih Berikutnya.

ADD A CONNECTOR TO YOUR GREENGRASS GROUP

Select a connector

STEP 1/2

Select a connector to add to this group. Connectors that are already in the group are disabled in the list. [Learn more](#)

<input type="radio"/>	CloudWatch Metrics	Version: 2	Learn more
<input type="radio"/>	Device Defender	Version: 2	Learn more
<input type="radio"/>	Docker Application Deployment	Version: 1	Learn more
<input checked="" type="radio"/>	IoT SiteWise	Version: 2	Learn more
<input type="radio"/>	IoT Analytics	Version: 2	Learn more
<input type="radio"/>	Kinesis Firehose	Version: 3	Learn more
<input type="radio"/>	ML Feedback	Version: 1	Learn more
<input type="radio"/>	ML Image Classification ARMv7	Version: 2	Learn more
<input type="radio"/>	ML Image Classification Aarch64 JTX2	Version: 2	Learn more
<input type="radio"/>	ML Image Classification x86_64	Version: 2	Learn more

[Cancel](#) [Next](#)

5. Jika server Anda memerlukan otentikasi, Anda dapat membuat AWS Secrets Manager rahasia dengan nama pengguna dan kata sandi server. Kemudian, Anda dapat melampirkan setiap rahasia ke grup Greengrass Anda dan memilihnya di bawah Daftar ARN untuk rahasia nama pengguna/kata sandi. Untuk informasi selengkapnya tentang cara membuat dan mengonfigurasi rahasia, lihat [Mengkonfigurasi otentikasi sumber](#). Anda juga dapat menambahkan rahasia ke konektor Anda nanti.

List of ARNs for OPC-UA username/password secrets (optional)

List of AWS Secret ARNs

2 secrets selected		Create ↗	Refresh	Clear	Close
Search					
<input checked="" type="checkbox"/>	greengrass-factory1-auth				
<input checked="" type="checkbox"/>	greengrass-factory2-auth				

- Jika Anda mengatur gateway SiteWise Edge Anda dengan jalur yang berbeda dari `/var/sitewise`, masukkan jalur itu untuk jalur penyimpanan lokal.
- (Opsional) Masukkan ukuran buffer disk maksimum untuk konektor. Jika AWS IoT Greengrass inti kehilangan koneksi ke AWS Cloud, konektor menyimpan data hingga berhasil terhubung. Jika ukuran cache melebihi ukuran buffer disk maksimum, konektor membuang data terlama dari antrian.
- Pilih Tambahkan.
- Di sudut kanan atas, di menu Tindakan, pilih Deploy.
- Pilih Deteksi otomatis untuk memulai penerapan.

Jika penerapan gagal, pilih Deploy lagi. Jika penerapan terus gagal, lihat pemecahan masalah [AWS IoT Greengrass penerapan](#).

Menambahkan gateway SiteWise Edge ke AWS IoT SiteWise

Dalam prosedur ini, Anda menambahkan grup Greengrass gateway SiteWise Edge Anda ke AWS IoT SiteWise. Setelah Anda mendaftarkan gateway SiteWise Edge Anda AWS IoT SiteWise, layanan dapat menerapkan konfigurasi sumber data Anda ke gateway SiteWise Edge Anda.

Untuk menambahkan gateway SiteWise Edge ke AWS IoT SiteWise

- Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
- Pilih Tambahkan gateway.
- Pada halaman Add SiteWise gateway, lakukan hal berikut:

- a. Masukkan Nama untuk gateway SiteWise Edge. Pertimbangkan untuk memasukkan lokasi gateway SiteWise Edge dalam nama sehingga Anda dapat dengan mudah mengidentifikasinya.
- b. Untuk ID grup Greengrass, pilih grup Greengrass yang Anda buat sebelumnya.

Example

AWS IoT SiteWise > Gateways > Add SiteWise gateway

Add SiteWise gateway

Select a connected gateway

SiteWise utilizes an on-premises gateway that collects data from local data servers and uploads the selected data. Once you or your IT Administrator have installed the software, registered it to AWS IoT Greengrass and connected it to your local network you can add it to the SiteWise service.
[Learn more about this process and ordering hardware](#)

Gateway name
Using the deployment location as a name makes identifying your gateway easier.

Alexandria

Greengrass group ID
SiteWise gateway appliances must be connected to via AWS IoT Greengrass.

SiteWiseDemo

Cancel **Add gateway**

- c. (Opsional) Untuk kemampuan Edge, pilih Paket pemrosesan data. Ini memungkinkan komunikasi antara gateway SiteWise Edge Anda dan model aset dan aset apa pun yang dikonfigurasi untuk edge. Untuk informasi selengkapnya, lihat [the section called “Mengaktifkan pemrosesan data tepi”](#).

⚠ Important

Jika Anda menambahkan paket pemrosesan data ke gateway SiteWise Edge Anda, Anda harus mengonfigurasi dan menerapkan konektor SiteWise Edge pada AWS IoT Greengrass grup Anda. Ikuti langkah selanjutnya.

- d. Pilih Tambahkan gateway.

4. Jika Anda menambahkan paket pemrosesan data ke gateway SiteWise Edge Anda, konfigurasi dan gunakan konektor Prosesor AWS IoT SiteWise Data di AWS IoT Greengrass grup Anda. Ikuti langkah-langkah [the section called “Mengkonfigurasi konektor AWS IoT SiteWise”](#) untuk mengkonfigurasi konektor Prosesor AWS IoT SiteWise Data:
 - a. Untuk Pilih konektor di AWS IoT Greengrass konsol, pilih Prosesor AWS IoT SiteWise Data.
 - b. Untuk jalur penyimpanan lokal, masukkan jalur ke gateway SiteWise Edge Anda.
 - c. Pilih Tambahkan.
 - d. Di sudut kanan atas, di menu Tindakan, pilih Deploy, lalu pilih Deteksi otomatis untuk memulai penerapan.

Setelah gateway SiteWise Edge Anda digunakan, Anda dapat menambahkan sumber untuk setiap peralatan industri tempat Anda ingin gateway SiteWise Edge Anda untuk menelan data. Untuk informasi selengkapnya, lihat [Mengkonfigurasi sumber data](#).

Anda dapat melihat CloudWatch metrik Amazon untuk memverifikasi bahwa gateway SiteWise Edge Anda terhubung AWS IoT SiteWise. Untuk informasi selengkapnya, lihat [AWS IoT Greengrass Version 1 metrik gateway](#).

Mengkonfigurasi sumber data di gateway AWS IoT Greengrass V1 SiteWise Edge

Setelah menyiapkan gateway AWS IoT SiteWise Edge, Anda dapat mengonfigurasi sumber data sehingga gateway SiteWise Edge Anda dapat menelan data dari peralatan industri lokal ke AWS IoT SiteWise. Setiap sumber mewakili server lokal, seperti server OPC-UA, yang menghubungkan gateway SiteWise Edge Anda dan mengambil aliran data industri. Untuk informasi selengkapnya tentang menyiapkan gateway SiteWise Edge, lihat [Mengkonfigurasi gateway AWS IoT Greengrass V1 SiteWise Edge](#).

Note

AWS IoT SiteWise restart gateway SiteWise Edge Anda setiap kali Anda menambahkan atau mengedit sumber. Gateway SiteWise Edge Anda tidak akan menelan data saat dimulai ulang. Waktu untuk memulai ulang gateway SiteWise Edge Anda tergantung pada jumlah tag pada sumber gateway SiteWise Edge Anda. Waktu restart dapat berkisar dari beberapa detik

(untuk gateway SiteWise Edge dengan beberapa tag) hingga beberapa menit (untuk gateway SiteWise Edge dengan banyak tag).

Setelah membuat sumber, Anda dapat mengaitkan aliran data Anda dengan properti aset. Untuk informasi selengkapnya tentang cara membuat dan menggunakan aset, lihat [Pemodelan aset industri](#) dan [Memetakan aliran data industri ke properti aset](#).

Anda dapat melihat CloudWatch metrik untuk memverifikasi bahwa sumber data terhubung AWS IoT SiteWise. Untuk informasi selengkapnya, lihat [AWS IoT Greengrass Version 1 metrik gateway](#).

Saat ini, AWS IoT SiteWise mendukung protokol sumber data berikut:

- [OPC-UA](#) — Protokol komunikasi machine-to-machine (M2M) untuk otomasi industri.
- [Modbus TCP](#) — Protokol komunikasi data yang digunakan untuk berinteraksi dengan programmable logic controllers (PLC).
- [Ethernet/IP \(EIP\) — Protokol jaringan industri yang mengadaptasi Common Industrial Protocol \(CIP\)](#) ke Ethernet standar.

Note

SiteWise Edge gateway yang berjalan AWS IoT Greengrass V2 saat ini tidak mendukung sumber Modbus TCP dan Ethernet IP.

Topik

- [Konfigurasi sumber Modbus TCP](#)
- [Konfigurasi sumber Ethernet/IP \(EIP\)](#)
- [Mengkonfigurasi otentikasi sumber](#)
- [Memutakhirkan konektor](#)

Konfigurasi sumber Modbus TCP

Anda dapat menggunakan AWS IoT SiteWise konsol atau kemampuan gateway AWS IoT SiteWise Edge untuk menentukan dan menambahkan sumber Modbus TCP ke gateway SiteWise Edge Anda. Sumber ini mewakili server Modbus TCP lokal.

Note

- SiteWise Gateway tepi yang berjalan AWS IoT Greengrass V2 saat ini tidak mendukung sumber Modbus TCP.
- Anda harus menginstal AWS IoT SiteWise konektor untuk menggunakan sumber Modbus TCP.

Anda dapat menggunakan sumber Modbus TCP untuk mengonversi tipe data dari sumber Anda menjadi tipe data yang berbeda saat diterima di gateway SiteWise Edge Anda. Tipe data sumber menentukan tipe data yang dapat Anda pilih untuk data tujuan Anda. Anda juga dapat memilih untuk menukar byte menggunakan sumber Modbus TCP. Tabel berikut memberikan informasi lebih lanjut tentang tipe data sumber, tipe data tujuan, dan mode swap yang kompatibel.

Untuk informasi selengkapnya tentang mode swap, lihat artikel [How Real \(Floating Point\) dan 32-bit Data Encoded in Modbus RTU Messages tentang pengkodean pesan Modbus](#).

Jenis data sumber	Tipe data tujuan yang kompatibel	Mode swap yang kompatibel	Versi konektor yang kompatibel
ASCII	String	NoSwap	2
UTF8	String	NoSwap	2
ISO8859	String	NoSwap	2
Int16	Integer, Ganda, String	NoSwap	1 dan 2
Int32	Integer, Ganda, String	NoSwap,, ByteSwap byteWordSwap, WordSwap	1 dan 2
Desimal	Ganda, String	NoSwap,, ByteSwap byteWordSwap, WordSwap	1 dan 2
Boolean	Boolean	NoSwap	1 dan 2

Jenis data sumber	Tipe data tujuan yang kompatibel	Mode swap yang kompatibel	Versi konektor yang kompatibel
Hex-dump	String	NoSwap	1 dan 2

Topik

- [Konfigurasi sumber Modbus TCP \(konsol\)](#)
- [Konfigurasi sumber Modbus TCP \(CLI\)](#)

Konfigurasi sumber Modbus TCP (konsol)

Untuk mengkonfigurasi sumber Modbus TCP

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Gateway.
3. Pada gateway SiteWise Edge yang ingin Anda buat sumbernya, pilih Kelola, lalu pilih Lihat detail.
4. Pilih Sumber baru di sudut kanan atas.
5. Untuk opsi Protokol, pilih Modbus TCP.
6. Untuk konfigurasi sumber Modbus TCP, masukkan Nama untuk sumbernya.
7. Untuk alamat IP, masukkan alamat IP untuk server sumber data.
8. (Opsional) Masukkan Port dan ID Unit untuk server sumber.
9. (Opsional) Untuk durasi antar permintaan minimum, masukkan interval waktu antara permintaan berikutnya yang dikirim ke server Anda. Gateway SiteWise Edge Anda secara otomatis menghitung interval minimum yang diizinkan berdasarkan perangkat Anda dan jumlah register yang Anda miliki.
10. Untuk grup Properti, masukkan Nama.
11. Untuk Properti:
 - a. Untuk Tag, masukkan alias properti untuk set register Anda. Misalnya, **TT-001**.
 - b. Untuk alamat Register, masukkan alamat register yang memulai set register.
 - c. Untuk tipe data Sumber, pilih tipe data Modbus TCP yang ingin Anda konversi datanya. Ini default ke Hex dump.

Note

Tipe data sumber yang Anda pilih menentukan ukuran data, tipe data tujuan, dan mode swap yang dapat Anda pilih. Untuk informasi selengkapnya, lihat [the section called “Konfigurasi sumber Modbus TCP”](#).

- d. Untuk ukuran Data, masukkan jumlah register yang akan dibaca saat memulai dari alamat Register. Ini ditentukan oleh tipe data sumber yang Anda pilih untuk sumber ini.
 - e. Untuk tipe data Tujuan, pilih tipe AWS IoT SiteWise data yang ingin dikonversi ke data Anda. Defaultnya adalah String. Jenis tujuan harus kompatibel dengan tipe data sumber yang Anda pilih untuk sumber ini. Untuk informasi selengkapnya, lihat [the section called “Konfigurasi sumber Modbus TCP”](#).
 - f. Untuk mode Swap, pilih mode swap data yang ingin Anda gunakan untuk membaca data dari set register Anda. Mode swap harus kompatibel dengan tipe data sumber yang Anda pilih untuk sumber ini. Untuk informasi selengkapnya, lihat [the section called “Konfigurasi sumber Modbus TCP”](#).
12. Untuk tingkat Pemindaian, perbarui tingkat di mana Anda ingin gateway SiteWise Edge membaca register Anda. AWS IoT SiteWise secara otomatis menghitung tingkat pemindaian minimum yang diijinkan untuk gateway SiteWise Edge Anda.
 13. (Opsional) Untuk Tujuan, pilih tempat data sumber dikirim. Secara default, sumber Anda mengirim data ke AWS IoT SiteWise. Anda dapat menggunakan AWS IoT Greengrass aliran untuk mengeksport data Anda ke tujuan lokal atau ke AWS Cloud sebagai gantinya.

Note

Anda harus memilih AWS IoT SiteWise sebagai tujuan untuk data sumber Anda jika Anda ingin memproses data dari sumber ini di tepi AWS IoT SiteWise. Untuk informasi selengkapnya tentang pemrosesan data di edge, lihat [the section called “Mengaktifkan pemrosesan data tepi”](#).

Untuk mengirim data Anda ke tujuan lain:

- a. Untuk opsi Tujuan, pilih Tujuan lain.
- b. Untuk nama aliran Greengrass, masukkan nama yang tepat dari streaming Anda. AWS IoT Greengrass

Note

Anda dapat menggunakan aliran yang telah Anda buat, atau Anda dapat membuat AWS IoT Greengrass aliran baru untuk mengekspor data Anda. Jika Anda ingin menggunakan aliran yang ada, Anda harus memasukkan nama yang tepat dari aliran atau aliran baru akan dibuat.

Untuk informasi selengkapnya tentang bekerja dengan AWS IoT Greengrass aliran, lihat [Mengelola aliran data](#) di panduan AWS IoT Greengrass pengembang.

14. Pilih Tambahkan sumber.

AWS IoT SiteWise menyebarkan konfigurasi gateway SiteWise Edge ke AWS IoT Greengrass inti Anda. Anda tidak perlu meluncurkan penerapan secara manual.

Konfigurasi sumber Modbus TCP (CLI)

Anda dapat menentukan sumber data Modbus TCP dalam kemampuan gateway SiteWise Edge. Anda harus menentukan semua sumber Modbus TCP Anda dalam satu konfigurasi kemampuan.

Untuk informasi lebih lanjut tentang mendefinisikan sumber dengan AWS CLI, lihat [the section called "Mengkonfigurasi sumber data \(\)AWS CLI"](#).

Note

Anda harus menginstal AWS IoT SiteWise konektor untuk menggunakan sumber Modbus TCP.

Kemampuan ini memiliki versi berikut.

Versi	Namespace
1	<code>iotsitewise:modbuscollector:1</code>

Parameter konfigurasi kemampuan Modbus TCP

Ketika Anda menentukan sumber Modbus TCP dalam konfigurasi kemampuan, Anda harus menentukan informasi berikut dalam dokumen `capabilityConfiguration` JSON:

sumber

Daftar struktur definisi sumber Modbus-TCP yang masing-masing berisi informasi berikut:

name

Nama yang unik dan ramah untuk sumbernya.

measurementDataStreamAwalan

(Opsional) String untuk ditambahkan ke semua aliran data dari sumber. Gateway SiteWise Edge menambahkan awalan ini ke semua aliran data dari sumber ini. Gunakan awalan aliran data untuk membedakan antara aliran data yang memiliki nama yang sama dari sumber yang berbeda. Setiap aliran data harus memiliki nama unik dalam akun Anda.

tujuan

Struktur tujuan yang berisi informasi berikut:

tipe

Jenis tujuan.

StreamName

Nama AWS IoT Greengrass sungai.

streamBufferSize

Ukuran buffer aliran.

titik akhir

Struktur endpoint yang berisi informasi berikut:

IPAddress

Alamat IP dari sumber Modbus TCP.

port

(Opsional) Port sumber Modbus TCP.

UNITiD

(Opsional) UNITiD. Ini default ke nilai 1.

minimumInterRequestDurasi

Durasi minimum antara setiap permintaan dalam milidetik.

PropertyGroups

Daftar grup properti yang menentukan definisi tag yang diminta oleh protokol.

name

Nama grup properti. Ini harus menjadi pengidentifikasi unik.

tagPathDefinitions

Lokasi pengukuran di dalam sumber. Misalnya, urutan byte dan kata, alamat, dan tipe transformasi. Struktur masing-masing `MeasurementPathDefinition` ditentukan oleh konektor.

ScanMode

Mendefinisikan perilaku mode pemindaian dan parameter yang dapat dikonfigurasi untuk sumber.

Konfigurasi sumber Ethernet/IP (EIP)

Anda dapat menggunakan AWS IoT SiteWise konsol atau kemampuan gateway SiteWise Edge untuk menentukan dan menambahkan sumber IP Ethernet ke gateway SiteWise Edge Anda. Sumber ini mewakili server IP Ethernet lokal.

Note

- SiteWise Edge gateway yang berjalan AWS IoT Greengrass V2 saat ini tidak mendukung sumber IP Ethernet.
- Anda harus menginstal AWS IoT SiteWise konektor untuk menggunakan sumber IP Ethernet.

Topik

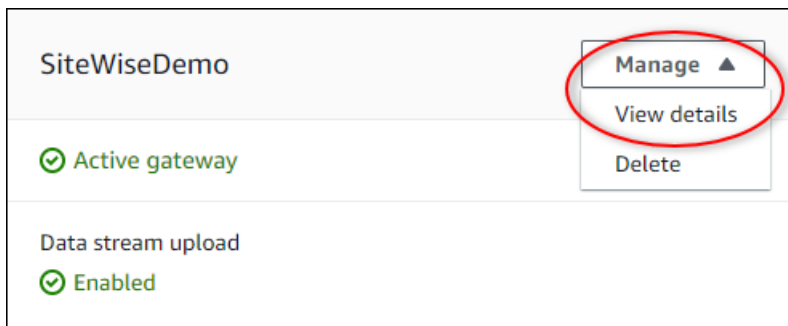
- [Konfigurasi sumber Ethernet/IP \(konsol\)](#)

- [Konfigurasi sumber Ethernet/IP \(CLI\)](#)

Konfigurasi sumber Ethernet/IP (konsol)


Untuk mengkonfigurasi sumber Ethernet/IP

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Gateway.
3. Pada gateway SiteWise Edge yang ingin Anda buat sumbernya, pilih Kelola, lalu pilih Lihat detail.



4. Pilih Sumber baru di sudut kanan atas.
5. Untuk opsi Protokol, pilih Ethernet/IP (EIP).
6. Untuk konfigurasi EtherNet sumber/IP, masukkan Nama untuk sumbernya.
7. Untuk alamat IP, masukkan alamat IP untuk server sumber data.
8. (Opsional) Masukkan Port untuk server sumber.
9. Untuk Durasi antar permintaan minimum, masukkan interval waktu antara permintaan berikutnya yang dikirim ke server Anda. Gateway SiteWise Edge Anda secara otomatis menghitung interval minimum yang diizinkan berdasarkan perangkat Anda dan jumlah register yang Anda miliki.
10. Untuk grup Properti, masukkan Nama.
11. Untuk Properti:
 - a. Untuk Tag, masukkan alias properti untuk set register Anda. Misalnya, **boiler.inlet.temperature.value**.
 - b. Untuk tipe data Tujuan, pilih tipe AWS IoT SiteWise data yang ingin dikonversi ke data Anda. Defaultnya adalah String.
12. Untuk tingkat Pemindaian, perbarui tingkat di mana Anda ingin gateway SiteWise Edge membaca register Anda. AWS IoT SiteWise secara otomatis menghitung tingkat pemindaian minimum yang diizinkan untuk gateway SiteWise Edge Anda.


13. (Opsional) Untuk Tujuan, pilih tempat data sumber dikirim. Secara default, sumber Anda mengirim data ke AWS IoT SiteWise. Anda dapat menggunakan AWS IoT Greengrass aliran untuk mengekspor data Anda ke tujuan lokal atau ke AWS Cloud sebagai gantinya.

 Note

Anda harus memilih AWS IoT SiteWise sebagai tujuan untuk data sumber Anda jika Anda ingin memproses data dari sumber ini di tepi AWS IoT SiteWise. Untuk informasi selengkapnya tentang pemrosesan data di edge, lihat [the section called “Mengaktifkan pemrosesan data tepi”](#).

Untuk mengirim data Anda ke tujuan lain:

- a. Untuk opsi Tujuan, pilih Tujuan lain.
- b. Untuk nama aliran Greengrass, masukkan nama yang tepat dari streaming Anda. AWS IoT Greengrass

 Note

Anda dapat menggunakan aliran yang telah Anda buat, atau Anda dapat membuat AWS IoT Greengrass aliran baru untuk mengekspor data Anda. Jika Anda ingin menggunakan aliran yang ada, Anda harus memasukkan nama yang tepat dari aliran atau aliran baru akan dibuat.

Untuk informasi selengkapnya tentang bekerja dengan AWS IoT Greengrass aliran, lihat [Mengelola aliran data](#) di panduan AWS IoT Greengrass pengembang.


14. Pilih Tambahkan sumber.

AWS IoT SiteWise menyebarkan konfigurasi gateway SiteWise Edge ke AWS IoT Greengrass inti Anda. Anda tidak perlu meluncurkan penerapan secara manual.

Konfigurasi sumber Ethernet/IP (CLI)

Anda dapat menentukan sumber data EIP dalam kemampuan gateway SiteWise Edge. Anda harus menentukan semua sumber EIP Anda dalam satu konfigurasi kemampuan.

Untuk informasi lebih lanjut tentang mendefinisikan sumber dengan AWS CLI, lihat [the section called “Mengkonfigurasi sumber data \(\)AWS CLI”](#).

 Note

Anda harus menginstal AWS IoT SiteWise konektor untuk menggunakan sumber IP Ethernet.

Kemampuan ini memiliki versi berikut.

Versi	Namespace
1	iotsitewise:eipcollector:1

Parameter konfigurasi kemampuan EIP

Ketika Anda menentukan sumber EIP dalam konfigurasi kemampuan, Anda harus menentukan informasi berikut dalam dokumen `capabilityConfiguration` JSON:

sumber

Daftar struktur definisi sumber EIP yang masing-masing berisi informasi berikut:

name

Nama yang unik dan ramah untuk sumbernya. Ini bisa sampai 256 karakter.

destinationPathPrefix

(Opsional) String untuk ditambahkan ke semua aliran data dari sumber. Gateway SiteWise Edge menambahkan awalan ini ke semua aliran data dari sumber ini. Gunakan awalan aliran data untuk membedakan antara aliran data yang memiliki nama yang sama dari sumber yang berbeda. Setiap aliran data harus memiliki nama unik dalam akun Anda.

tujuan

Struktur tujuan yang berisi informasi berikut:

tipe

Jenis tujuan.

StreamName

Nama AWS IoT Greengrass sungai.

streamBufferSize

Ukuran buffer aliran.

titik akhir

Struktur endpoint yang berisi informasi berikut:

IPAddress

Alamat IP dari sumber EIP.

port

(Opsional) Port sumber EIP. Nilai yang diterima adalah angka antara 1 dan 65535.

minimumInterRequestDurasi

(Opsional) Durasi minimum antara setiap permintaan dalam milidetik.

PropertyGroups

Daftar grup properti yang menentukan definisi tag yang diminta oleh protokol. Setiap sumber dapat memiliki satu grup properti.

name

Nama grup properti. Ini harus menjadi pengidentifikasi unik dengan panjang maksimum 256 karakter.

tagPathDefinitions

Daftar struktur yang menentukan data yang akan dikumpulkan dari perangkat Ethernet/IP dan bagaimana mengubahnya untuk output.

tipe

Jenis dari `tagPathDefinition`. Misalnya, `EIPTagPath`.

path

Jalan dari `tagPathDefinition`. Setiap tag di jalur dapat memiliki panjang maksimum 40 karakter dan dapat dimulai dengan huruf atau garis bawah. Tag tidak dapat

berisi garis bawah berurutan atau tertinggal. Jalur diawali dengan nilai apa pun dari `destinationPathPrefix`

`dstDataType`

Tipe data untuk menampilkan data tag. Nilai yang diterima adalah `integer`, `double`, `string`, dan `boolean`.

`ScanMode`

Mendefinisikan perilaku mode pemindaian dan parameter yang dapat dikonfigurasi untuk sumber.

`tipe`

Jenis perilaku mode pemindaian. Nilai yang diterima adalah `POLL`.

`tingkat`

Tingkat dalam milidetik konektor harus membaca tag dari sumber Ethernet/IP.

Mengkonfigurasi otentikasi sumber

Jika server OPC-UA Anda memerlukan kredensi otentikasi untuk terhubung, Anda dapat menentukan nama pengguna dan kata sandi secara rahasia untuk setiap sumber di AWS Secrets Manager. Kemudian, Anda menambahkan rahasia ke grup Greengrass dan konektor SiteWise IoT Anda untuk membuat rahasia tersedia untuk gateway Edge Anda. SiteWise Untuk informasi selengkapnya, lihat [Menerapkan rahasia ke AWS IoT Greengrass inti](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

Setelah rahasia tersedia untuk gateway SiteWise Edge Anda, Anda dapat memilihnya saat Anda mengonfigurasi sumber. Kemudian, gateway SiteWise Edge menggunakan kredensi otentikasi dari rahasia saat terhubung ke sumbernya. Untuk informasi selengkapnya, lihat [Mengkonfigurasi sumber data](#).

Topik

- [Membuat rahasia otentikasi sumber](#)
- [Menambahkan rahasia ke grup Greengrass](#)
- [Menambahkan rahasia ke konektor IoT SiteWise](#)

Membuat rahasia otentikasi sumber

Dalam prosedur ini, Anda membuat rahasia otentikasi untuk sumber Anda di Secrets Manager. Secara rahasia, tentukan **username** dan pasangan **password** nilai kunci yang berisi detail otentikasi untuk sumber Anda.

Untuk membuat rahasia otentikasi sumber

1. Arahkan ke [konsol Secrets Manager](#).
2. Pilih Simpan rahasia baru.
3. Di bawah Pilihan jenis rahasia, pilih Jenis rahasia lainnya.
4. Masukkan **username** dan pasangan **password** nilai kunci untuk nilai otentikasi server OPC-UA Anda, lalu pilih Berikutnya.

The screenshot shows the AWS Secrets Manager console interface. The 'Select secret type' section has four radio button options: 'Credentials for RDS database', 'Credentials for Redshift cluster', 'Credentials for DocumentDB database', and 'Other type of secrets (e.g. API key)'. The 'Other type of secrets' option is selected and highlighted with a red box. Below this, the 'Specify the key/value pairs to be stored in this secret' section is active, showing a table with two rows: 'username' and 'password'. Each row has a text input field and a 'Remove' button. The 'username' and 'password' text fields are also highlighted with red boxes. At the bottom of the form, there is a 'Select the encryption key' dropdown menu set to 'DefaultEncryptionKey' and a 'Next' button highlighted in orange.

5. Masukkan nama Rahasia yang dimulai dengan `greengrass-`, seperti `greengrass-factory1-auth`.

⚠ Important

Anda harus menggunakan `greengrass-` awalan untuk peran AWS IoT Greengrass layanan default untuk mengakses rahasia Anda. Jika Anda ingin memberi nama rahasia Anda tanpa awalan ini, Anda harus memberikan izin AWS IoT Greengrass khusus untuk mengakses rahasia Anda. Untuk informasi selengkapnya, lihat [AWS IoT Greengrass Mengizinkan untuk mendapatkan nilai rahasia](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

Store a new secret**Secret name and description** info**Secret name**

Give the secret a name that enables you to find and manage it easily.

Secret name must contain only alphanumeric characters and the characters `/_+=@-`

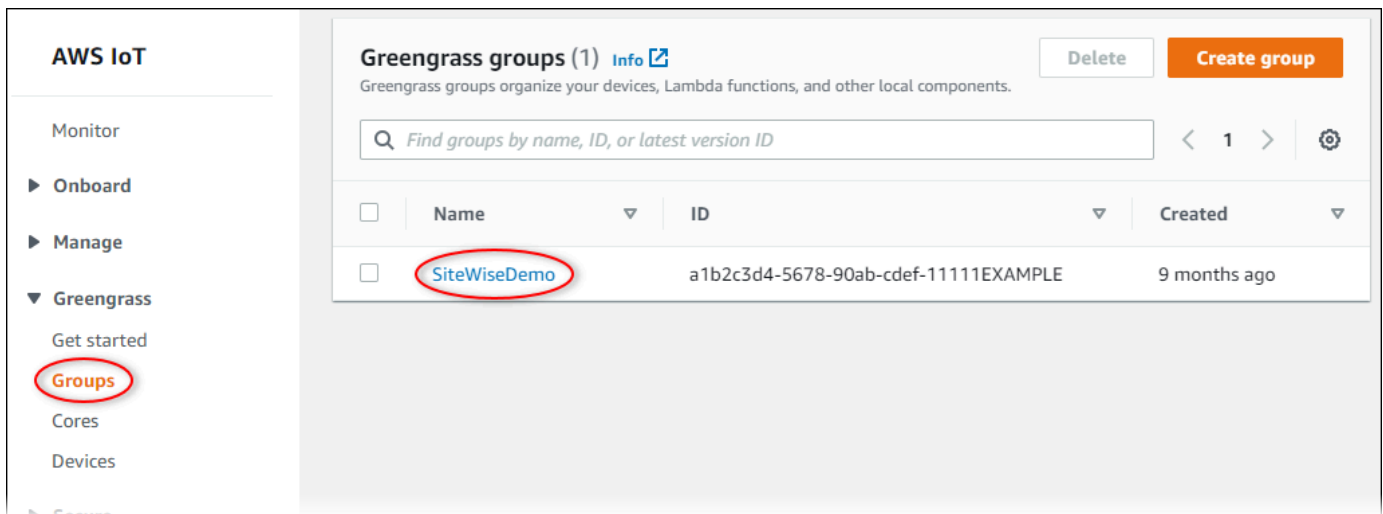
6. Masukkan Deskripsi dan pilih Berikutnya.
7. (Opsional) Pada halaman Konfigurasi rotasi otomatis, konfigurasi rotasi otomatis untuk rahasia Anda. Jika Anda mengonfigurasi rotasi otomatis, Anda harus menerapkan ulang grup Greengrass Anda setiap kali rahasia berputar.
8. Pada halaman Konfigurasi rotasi otomatis, pilih Berikutnya.
9. Tinjau rahasia baru Anda dan pilih Store.

Menambahkan rahasia ke grup Greengrass

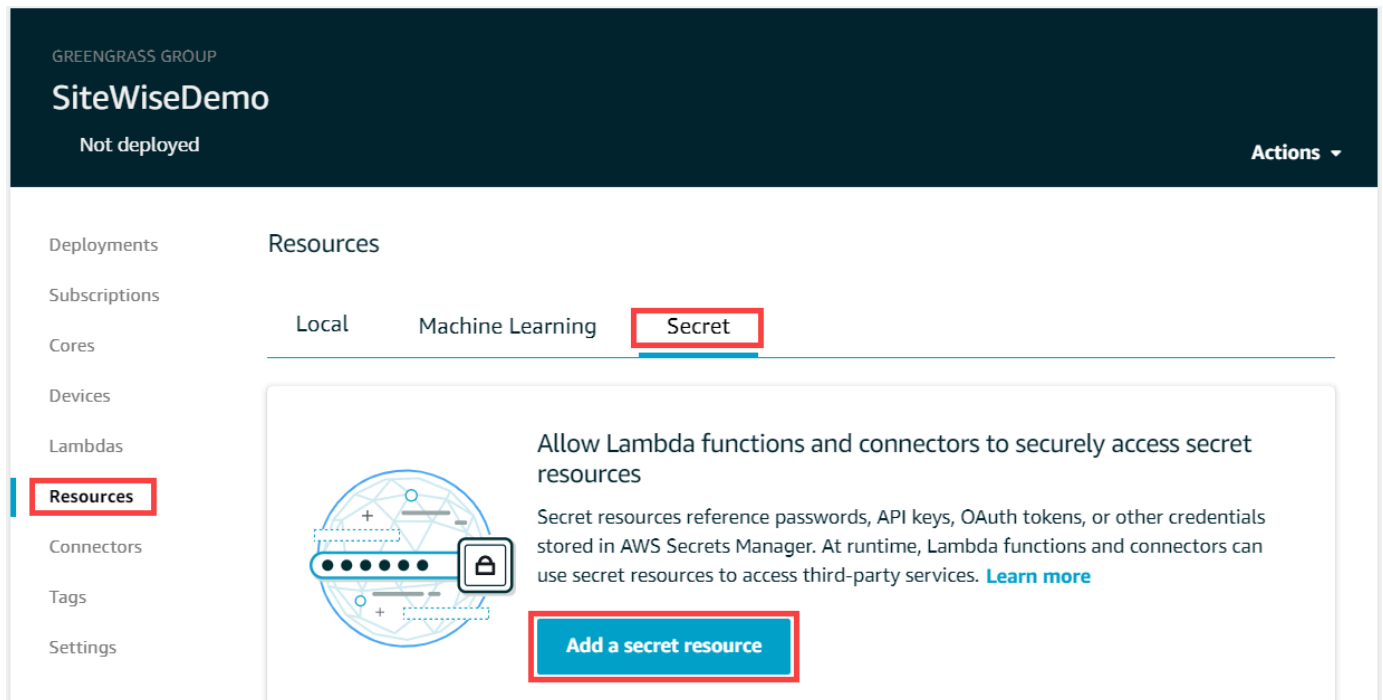
Dalam prosedur ini, Anda menambahkan rahasia otentikasi sumber Anda ke AWS IoT Greengrass grup Anda untuk membuatnya tersedia untuk konektor SiteWise IoT Anda.

Untuk menambahkan rahasia ke grup Greengrass Anda

1. Navigasikan ke [konsol AWS IoT Greengrass](#) tersebut.
2. Di panel navigasi, di bawah Greengrass, pilih Grup, lalu pilih grup Anda.



3. Di halaman navigasi, pilih Sumber Daya.
4. Pada halaman Sumber Daya, pilih tab Rahasia, lalu pilih Tambahkan sumber daya rahasia.



5. Pilih Pilih dan pilih rahasia Anda dari daftar.
6. Pilih Berikutnya.
7. Dalam nama sumber daya rahasia, masukkan nama untuk sumber daya rahasia Anda dan pilih Simpan.

ADD A RESOURCE TO YOUR GREENGRASS GROUP

Name your secret resource

STEP 3/3

Your secret resource will be added to the group. Give it a unique name so you can easily identify it. [Learn more](#)

Secret resource name

The name can contain alphanumeric characters, colons, underscores, and dashes.

Secret name
greengrass-factory1-auth

Labels
AWSCURRENT

[Cancel](#) [Back](#) [Save](#)

Menambahkan rahasia ke konektor IoT SiteWise

Dalam prosedur ini, Anda menambahkan rahasia otentikasi sumber Anda ke konektor SiteWise IoT Anda untuk membuatnya tersedia dan gateway AWS IoT SiteWise Edge SiteWise Anda.

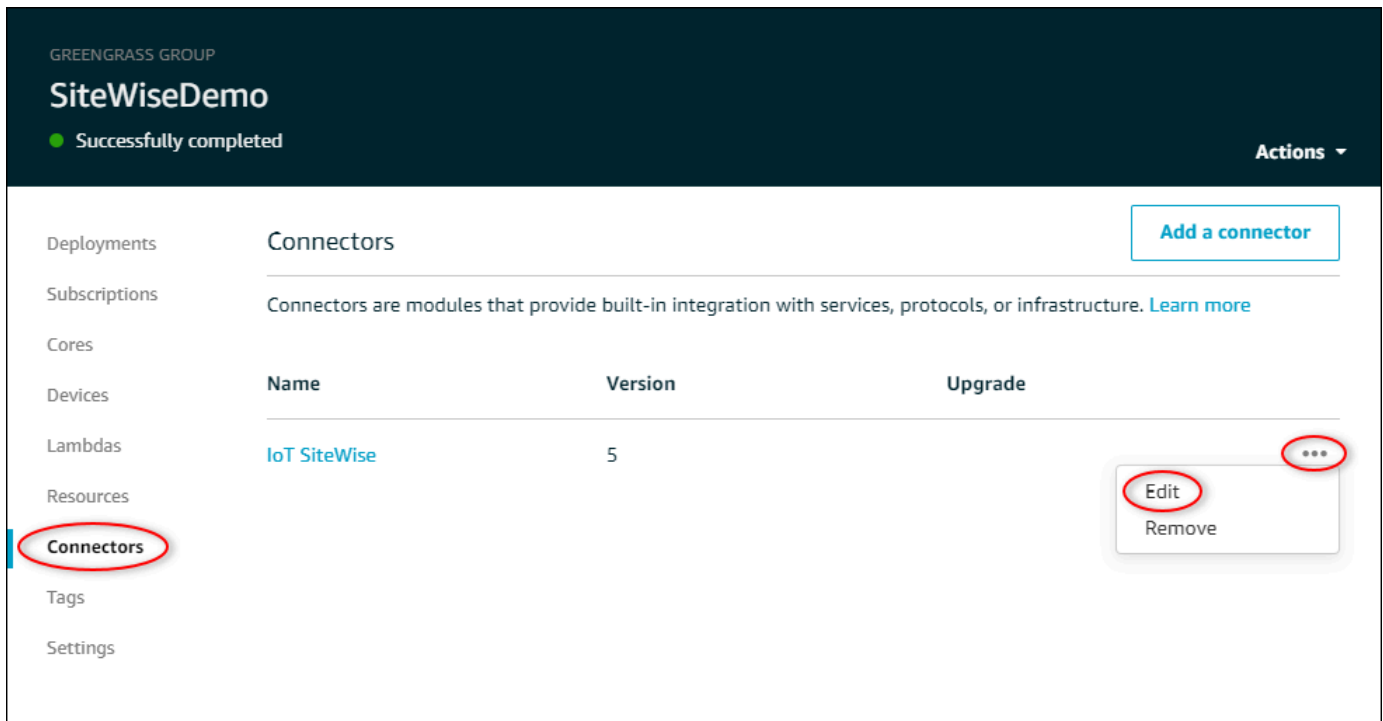
Untuk menambahkan rahasia ke konektor IoT SiteWise Anda

1. Navigasikan ke [konsol AWS IoT Greengrass](#) tersebut.
2. Di panel navigasi, di bawah Greengrass, pilih Grup, lalu pilih grup Anda.

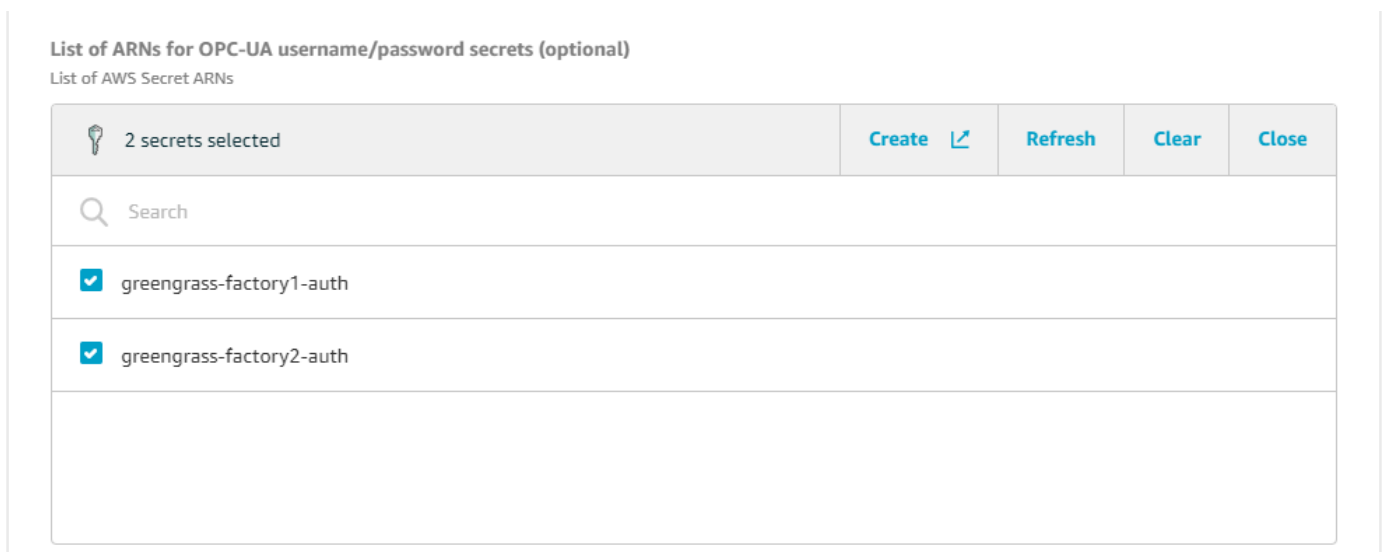
The screenshot shows the AWS IoT Greengrass console. On the left, the navigation pane is expanded to 'Greengrass', and the 'Groups' option is highlighted with a red circle. The main content area displays 'Greengrass groups (1)' with a search bar and a table of groups. The table has columns for 'Name', 'ID', and 'Created'. One group is listed with the name 'SiteWiseDemo' (circled in red), ID 'a1b2c3d4-5678-90ab-cdef-11111EXAMPLE', and 'Created' '9 months ago'. There are 'Delete' and 'Create group' buttons at the top right of the table.

	Name	ID	Created
<input type="checkbox"/>	SiteWiseDemo	a1b2c3d4-5678-90ab-cdef-11111EXAMPLE	9 months ago

- Di halaman navigasi, pilih Konektor.
- Pilih ikon elipsis untuk SiteWise konektor IoT untuk membuka menu opsi, lalu pilih Edit.



- Di bawah Daftar ARN untuk rahasia nama pengguna/kata sandi OPC-UA, pilih Pilih, lalu pilih setiap rahasia untuk ditambahkan ke gateway Edge ini. SiteWise Jika Anda perlu membuat rahasia, lihat [Membuat rahasia otentikasi sumber](#).



Jika rahasia Anda tidak muncul, pilih Refresh. Jika rahasia Anda masih belum muncul, periksa apakah Anda [menambahkan rahasia ke grup Greengrass Anda](#).

- Pilih Simpan.

7. Di sudut kanan atas, di menu Tindakan, pilih Deploy.
8. Pilih Deteksi otomatis untuk memulai penerapan.

Jika penerapan gagal, pilih Deploy lagi. Jika penerapan terus gagal, lihat pemecahan masalah [AWS IoT Greengrass penerapan](#).

Setelah grup Anda menyebarkan, Anda dapat mengonfigurasi sumber yang menggunakan rahasia baru. Untuk informasi selengkapnya, lihat [Mengkonfigurasi sumber data](#).

Memutakhirkan konektor

Important

[Versi 6 dari SiteWise konektor IoT memperkenalkan persyaratan baru: Perangkat lunak AWS IoT Greengrass inti v1.10.0 dan manajer aliran](#). Sebelum Anda meningkatkan konektor Anda, periksa apakah gateway SiteWise Edge Anda memenuhi persyaratan ini, atau Anda tidak akan dapat menggunakan gateway SiteWise Edge Anda.

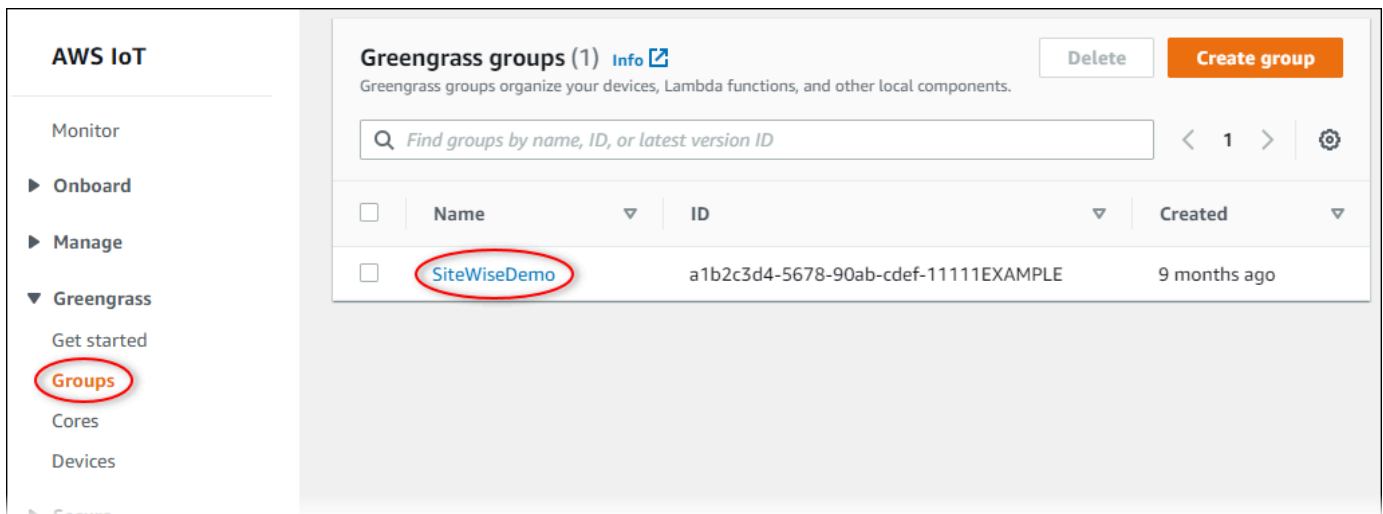
Anda dapat dengan mudah memutakhirkan konektor gateway SiteWise Edge Anda setelah versi SiteWise konektor IoT baru dirilis.

Note

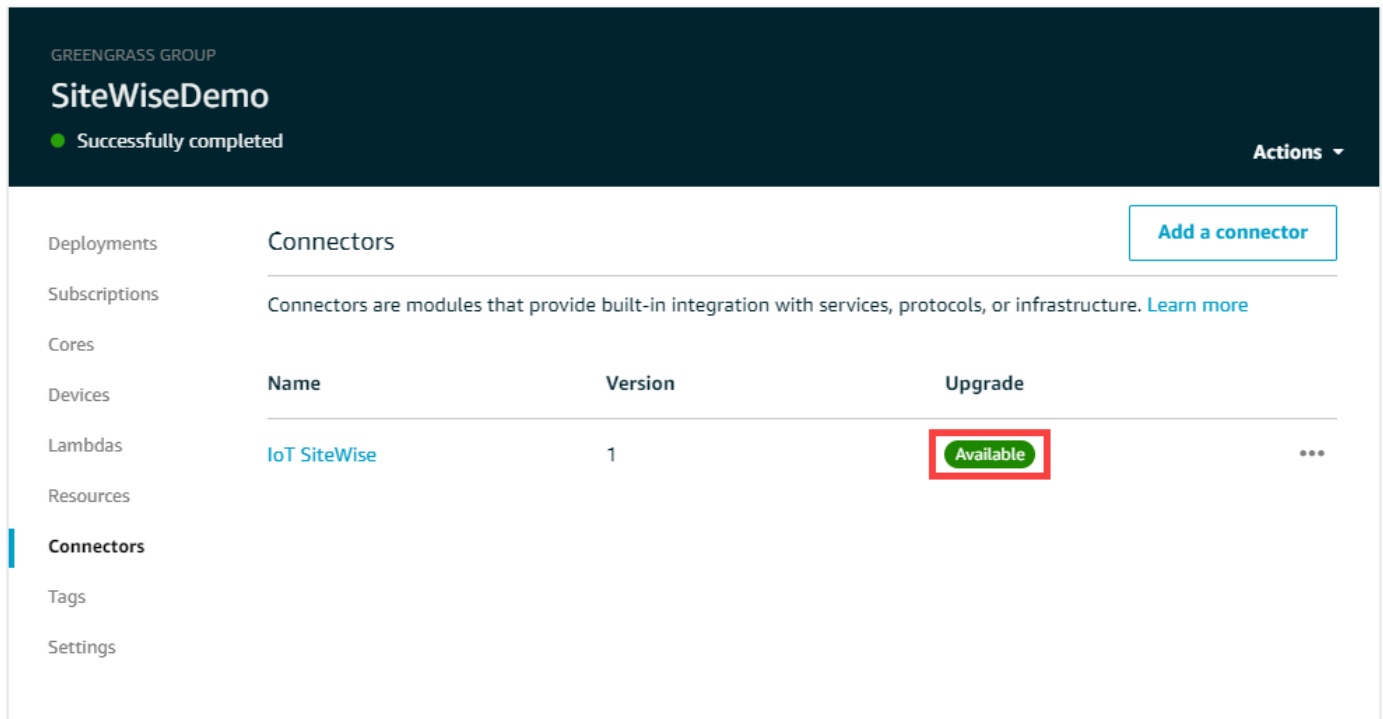
Dalam prosedur ini, Anda menerapkan ulang grup Greengrass Anda dan memulai ulang gateway Edge Anda. SiteWise Gateway SiteWise Edge Anda tidak akan menelan data saat dimulai ulang. Waktu untuk memulai ulang gateway SiteWise Edge Anda tergantung pada jumlah tag pada sumber gateway SiteWise Edge Anda. Waktu restart dapat berkisar dari beberapa detik (untuk gateway SiteWise Edge dengan beberapa tag) hingga beberapa menit (untuk gateway SiteWise Edge dengan banyak tag).

Untuk memutakhirkan konektor IoT SiteWise

1. Navigasikan ke [konsol AWS IoT Greengrass](#) tersebut.
2. Di panel navigasi, di bawah Greengrass, pilih Grup, lalu pilih grup yang Anda buat saat menyiapkan gateway Edge. SiteWise



3. Di panel navigasi, pilih Konektor.
4. Pada halaman Konektor, pilih Tersedia di sebelah konektor IoT SiteWise.



Jika Anda tidak melihat elemen Tersedia, konektor Anda sudah versi terbaru.

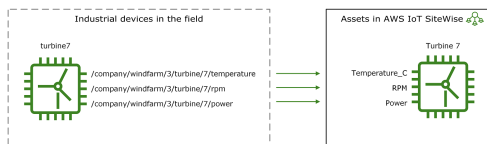
5. Pada halaman konektor Upgrade, masukkan parameter konektor Anda dan kemudian pilih Upgrade.
6. Di sudut kanan atas, di menu Tindakan, pilih Deploy.
7. Pilih Deteksi otomatis untuk memulai penerapan.

Jika penerapan gagal, pilih Deploy lagi. Jika penerapan terus gagal, lihat pemecahan masalah [AWS IoT Greengrass penerapan](#).

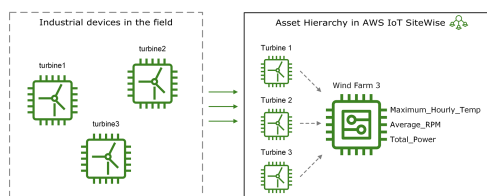
Pemodelan aset industri

Anda dapat membuat representasi virtual dari operasi industri Anda dengan AWS IoT SiteWise aset. Aset mewakili perangkat, peralatan, atau proses yang mengunggah satu atau lebih aliran data ke file. AWS Cloud Misalnya, perangkat aset dapat berupa turbin angin yang mengirimkan suhu udara, kecepatan rotasi baling-baling, dan pengukuran deret waktu keluaran daya ke properti aset di. AWS IoT SiteWise

Setiap aliran data sesuai dengan alias properti unik. Misalnya, alias `/company/windfarm/3/turbine/7/temperature` secara unik mengidentifikasi aliran data suhu yang berasal dari turbin #7 di ladang angin #3. Anda dapat mengonfigurasi AWS IoT SiteWise aset untuk mengubah data pengukuran yang masuk menggunakan ekspresi matematika, seperti mengonversi data suhu dari Celcius ke Fahrenheit.



Aset juga dapat mewakili pengelompokan perangkat yang logis, seperti seluruh ladang angin. Anda dapat mengaitkan aset dengan aset lain untuk membuat hierarki aset yang mewakili operasi industri yang kompleks. Aset dapat mengakses data dalam aset anak terkait. Dengan demikian, Anda dapat menggunakan AWS IoT SiteWise ekspresi untuk menghitung metrik agregat, seperti output daya bersih dari ladang angin.



Anda harus membuat setiap aset dari model aset. Model aset adalah struktur deklaratif yang menstandarisasi format aset Anda. Model aset menerapkan informasi yang konsisten di beberapa aset dengan jenis yang sama sehingga Anda dapat memproses data dalam aset yang mewakili grup perangkat. Pada diagram sebelumnya, Anda menggunakan model aset yang sama untuk ketiga turbin karena semua turbin memiliki satu set properti yang sama.

Anda juga dapat membuat model komponen. Model komponen adalah jenis model aset khusus yang dapat Anda sertakan dalam model aset atau model komponen lainnya. Anda dapat menggunakan

model komponen untuk menentukan sub-rakitan umum yang dapat digunakan kembali, seperti sensor, motor, dan sebagainya, yang Anda bagikan di beberapa model aset.

Setelah Anda menentukan model aset Anda, Anda dapat membuat aset industri Anda. Untuk membuat aset, pilih model ACTIVE aset untuk membuat aset dari model tersebut. Kemudian, Anda dapat mengisi informasi khusus aset seperti alias dan atribut aliran data. Pada diagram sebelumnya, Anda membuat tiga aset turbin dari satu model aset dan kemudian mengaitkan alias aliran data seperti `/company/windfarm/3/turbine/7/temperature` untuk setiap turbin.

Anda juga dapat memperbarui dan menghapus aset, model aset, dan model komponen yang ada. Saat Anda memperbarui model aset, setiap aset berdasarkan model aset tersebut mencerminkan perubahan apa pun yang Anda buat pada model yang mendasarinya. Saat Anda memperbarui model komponen, ini berlaku untuk setiap aset berdasarkan setiap model aset yang mereferensikan model komponen.

Model aset Anda mungkin sangat kompleks, misalnya saat memodelkan peralatan rumit yang memiliki banyak subkomponen. Untuk membantu menjaga agar model aset tersebut tetap teratur dan dapat dipelihara, Anda dapat menggunakan model komposit khusus untuk mengelompokkan properti terkait atau menggunakan kembali komponen bersama. Untuk informasi selengkapnya, lihat [Model komposit khusus \(Komponen\)](#).

Topik

- [Status aset dan model](#)
- [Model komposit khusus \(Komponen\)](#)
- [Bekerja dengan ID objek](#)
- [Membuat model aset dan model komponen](#)
- [Menciptakan aset](#)
- [Mencari aset](#)
- [Memetakan aliran data industri ke properti aset](#)
- [Memperbarui nilai atribut](#)
- [Megasosiasikan dan memisahkan aset](#)
- [Memperbarui aset dan model](#)
- [Menghapus aset dan model](#)
- [Operasi massal dengan aset dan model](#)

Status aset dan model

Saat Anda membuat, memperbarui, atau menghapus aset, model aset, atau model komponen, perubahan membutuhkan waktu untuk disebar. AWS IoT SiteWise menyelesaikan operasi ini secara asinkron dan memperbarui status setiap sumber daya. Setiap aset, model aset, dan model komponen memiliki bidang status yang berisi status sumber daya dan pesan kesalahan apa pun, jika berlaku. Negara dapat menjadi salah satu dari nilai berikut:

- **ACTIVE**— Sumber daya aktif. Ini adalah satu-satunya status di mana Anda dapat melakukan kueri dan berinteraksi dengan aset, model aset, dan model komponen.
- **CREATING**— Sumber daya sedang dibuat.
- **UPDATING**— Sumber daya sedang diperbarui.
- **DELETING**— Sumber daya sedang dihapus.
- **PROPAGATING**— (Model aset dan model komponen saja) Perubahan menyebar ke semua sumber daya dependen (dari model aset ke aset, atau dari model komponen ke model aset).
- **FAILED**— Sumber daya gagal memvalidasi selama operasi membuat atau memperbarui, mungkin karena referensi melingkar dalam ekspresi. Anda dapat menghapus sumber daya yang ada di **FAILED** negara bagian.

Beberapa operasi membuat, memperbarui, dan menghapus AWS IoT SiteWise menempatkan aset, model aset, atau model komponen dalam keadaan selain **ACTIVE** saat operasi diselesaikan. Untuk melakukan kueri atau berinteraksi dengan sumber daya setelah Anda melakukan salah satu operasi ini, Anda harus menunggu hingga status berubah **ACTIVE**. Jika tidak, permintaan Anda gagal.

Topik

- [Memeriksa status aset](#)
- [Memeriksa status model aset atau model komponen](#)

Memeriksa status aset

Anda dapat menggunakan AWS IoT SiteWise konsol atau API untuk memeriksa status aset.

Topik

- [Memeriksa status aset \(konsol\)](#)
- [Memeriksa status aset \(AWS CLI\)](#)

Memeriksa status aset (konsol)

Gunakan prosedur berikut untuk memeriksa status aset di AWS IoT SiteWise konsol.

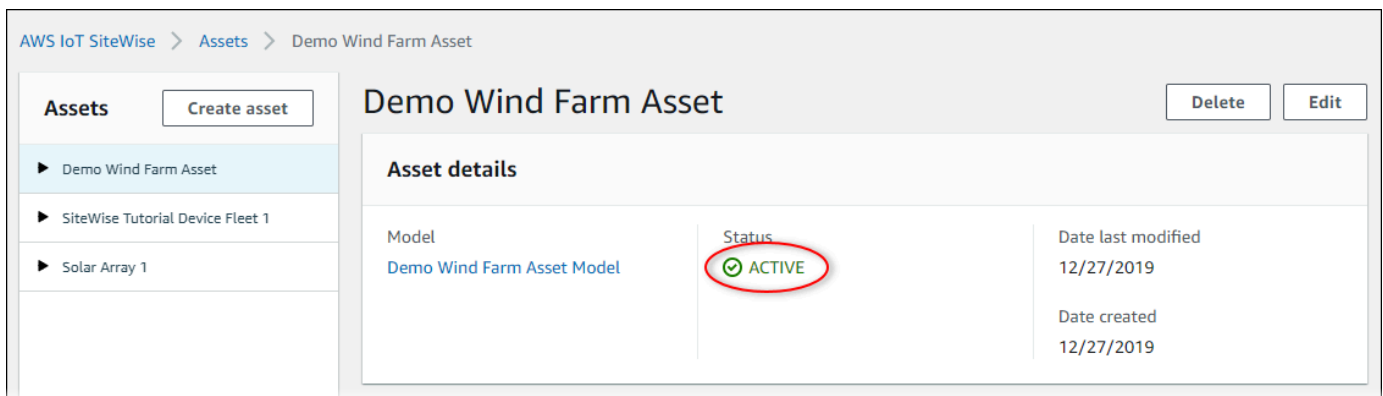
Untuk memeriksa status aset (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang akan diperiksa.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Temukan Status di panel detail Aset.



Memeriksa status aset (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk memeriksa status aset.

Untuk memeriksa status aset, gunakan [DescribeAsset](#) operasi dengan `assetId` parameter.

Untuk memeriksa status aset (AWS CLI)

- Jalankan perintah berikut untuk menggambarkan aset. Ganti *asset-id dengan ID* aset atau ID eksternal. ID eksternal adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

```
aws iotsitewise describe-asset --asset-id asset-id
```

Operasi mengembalikan respons yang berisi detail aset. Respons berisi `assetStatus` objek yang memiliki struktur berikut:

```
{
  ...
  "assetStatus": {
    "state": "String",
    "error": {
      "code": "String",
      "message": "String"
    }
  }
}
```

Status aset ada `assetStatus.state` di objek JSON.

Memeriksa status model aset atau model komponen

Anda dapat menggunakan AWS IoT SiteWise konsol atau API untuk memeriksa status model aset atau model komponen.

Topik

- [Memeriksa status model aset atau model komponen \(konsol\)](#)
- [Memeriksa status model aset atau model komponen \(AWS CLI\)](#)

Memeriksa status model aset atau model komponen (konsol)

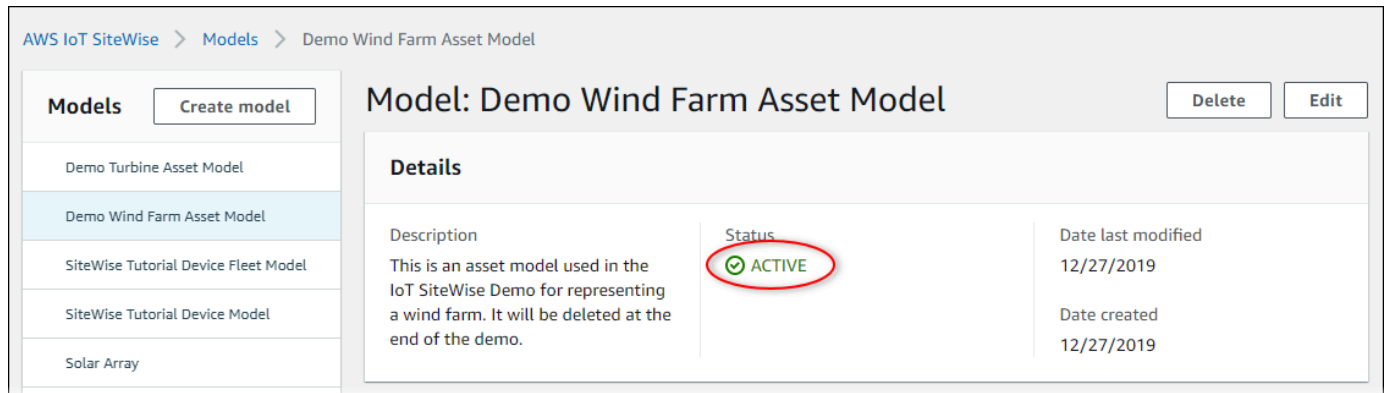
Gunakan prosedur berikut untuk memeriksa status model aset atau model komponen di AWS IoT SiteWise konsol.

Tip

Model aset dan model komponen keduanya tercantum di bawah Model di panel navigasi. Panel Detail dari model aset atau model komponen yang dipilih menunjukkan jenisnya.

Untuk memeriksa status model aset atau model komponen (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model.
3. Pilih model yang akan diperiksa.
4. Temukan Status di panel Detail.



Memeriksa status model aset atau model komponen (AWS CLI)

Anda dapat menggunakan AWS CLI untuk memeriksa status model aset atau model komponen.

Untuk memeriksa status model aset atau model komponen, gunakan [DescribeAssetModel](#) operasi dengan `assetModelId` parameter.

Tip

AWS CLI Mendefinisikan model komponen sebagai jenis model aset. Oleh karena itu, Anda menggunakan [DescribeAssetModel](#) operasi yang sama untuk kedua jenis model. `assetModelTypeBidang` dalam respons menunjukkan apakah itu an `ASSET_MODEL` atau `aCOMPONENT_MODEL`.

Untuk memeriksa status model aset atau model komponen (AWS CLI)

- Jalankan perintah berikut untuk menggambarkan model. Ganti `asset-model-id` dengan ID atau ID eksternal model aset atau model komponen. ID eksternal adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

```
aws iotsitewise describe-asset-model --asset-model-id asset-model-id
```

Operasi mengembalikan respons yang berisi detail model. Respons berisi `assetModelStatus` objek yang memiliki struktur berikut.

```
{
  ...
  "assetModelStatus": {
    "state": "String",
    "error": {
      "code": "String",
      "message": "String"
    }
  }
}
```

Status model ada `assetModelStatus.state` di objek JSON.

Model komposit khusus (Komponen)

Ketika Anda memodelkan aset industri yang sangat kompleks, seperti mesin rumit yang memiliki banyak bagian, dapat menjadi tantangan untuk menjaga model aset Anda tetap teratur dan dapat dipelihara.

Dalam kasus seperti itu, Anda dapat menambahkan model komposit khusus, atau komponen jika Anda menggunakan konsol, ke model aset dan model komponen yang ada. Ini membantu Anda tetap terorganisir dengan mengelompokkan properti terkait dan menggunakan kembali definisi subkomponen.

Ada dua jenis model komposit khusus:

- Model komposit kustom sebaris mendefinisikan satu set properti yang dikelompokkan yang berlaku untuk model aset atau model komponen yang menjadi milik model komposit kustom. Anda menggunakannya untuk mengelompokkan properti terkait. Mereka terdiri dari nama, deskripsi, dan satu set properti model aset. Mereka tidak dapat digunakan kembali.
- Model komposit `component-model-based` kustom C mereferensikan model komponen yang ingin Anda sertakan dalam model aset atau model komponen Anda. Anda menggunakannya untuk memasukkan subassemblies standar dalam model Anda. Mereka terdiri dari nama, deskripsi, dan

ID dari model komponen yang direferensikannya. Mereka tidak memiliki properti mereka sendiri; model komponen yang direferensikan menyediakan properti terkait untuk setiap aset yang dibuat.

Bagian berikut menggambarkan cara menggunakan model komposit khusus dalam desain Anda.

Topik

- [Model komposit kustom sebaris](#)
- [C model komposit component-model-based kustom](#)
- [Menggunakan jalur untuk mereferensikan properti model komposit khusus](#)

Model komposit kustom sebaris

Model komposit kustom sebaris menyediakan cara untuk mengatur model aset Anda dengan mengelompokkan properti terkait.

Misalnya, Anda ingin memodelkan aset robot. Robot termasuk servomotor, catu daya, dan baterai. Masing-masing bagian penyusun memiliki sifat sendiri yang ingin Anda sertakan dalam model. Anda dapat mendefinisikan model aset `robot_model` yang disebut yang memiliki properti seperti berikut ini.

- `robot_model`
 - `servo_status` (bilangan bulat)
 - `servo_position` (ganda)
 - `powersupply_status` (bilangan bulat)
 - `powersupply_temperature` (ganda)
 - `battery_status` (bilangan bulat)
 - `battery_charge` (ganda)

Namun, dalam beberapa kasus, mungkin ada banyak subassemblies, atau subassemblies itu sendiri mungkin memiliki banyak properti. Dalam kasus ini, mungkin ada begitu banyak properti sehingga menjadi rumit untuk direferensikan dan dipertahankan dalam satu daftar datar di root model, seperti pada contoh sebelumnya.

Untuk menghadapi situasi seperti itu, Anda dapat menggunakan model komposit kustom sebaris untuk mengelompokkan properti. Model komposit kustom inline adalah model komposit khusus yang mendefinisikan propertinya sendiri. Misalnya, Anda dapat memodelkan robot Anda seperti berikut ini.

- `robot_model`
 - `servo`
 - `status`(bilangan bulat)
 - `position`(ganda)
 - `powersupply`
 - `status`(bilangan bulat)
 - `temperature` (ganda)
 - `battery`
 - `status`(bilangan bulat)
 - `charge`(ganda)

Dalam contoh sebelumnya,, `servopowersupply`, dan `battery` adalah nama-nama model komposit kustom inline yang didefinisikan dalam model aset. `robot_model` Masing-masing model komposit ini kemudian mendefinisikan sifat-sifatnya sendiri.

Note

Dalam hal ini, setiap model komposit khusus mendefinisikan propertinya sendiri, sehingga semua properti adalah bagian dari model aset itu sendiri (`robot_model` dalam hal ini). Properti ini tidak dibagikan dengan model aset atau model komponen lainnya. Misalnya, jika Anda membuat beberapa model aset lain yang juga memiliki model komposit kustom sebaris yang disebut `servo`, maka membuat perubahan ke `servo` dalam `robot_model` tidak akan memengaruhi `servo` definisi model aset lainnya.

Jika Anda ingin menerapkan berbagi seperti itu (misalnya, hanya memiliki satu definisi untuk `servo`, yang dapat dibagikan oleh semua model aset Anda), Anda akan membuat model komponen untuknya, dan kemudian membuat model component-model-based komposit yang mereferensikannya. Lihat bagian berikut untuk detailnya.

Untuk informasi tentang cara membuat model komposit kustom inline, lihat [Membuat model komposit khusus \(Komponen\)](#).

C model komposit component-model-based kustom

Anda dapat membuat model komponen AWS IoT SiteWise untuk menentukan sub-rakitan standar yang dapat digunakan kembali. Setelah Anda membuat model komponen, Anda dapat menambahkan referensi ke model aset dan model komponen Anda yang lain. Anda melakukan ini dengan menambahkan model komposit component-model-based khusus ke model mana pun di mana Anda ingin mereferensikan komponen. Anda dapat menambahkan referensi ke komponen Anda dari banyak model, atau beberapa kali dalam model yang sama.

Dengan cara ini, Anda dapat menghindari duplikasi definisi yang sama di seluruh model. Ini juga menyederhanakan pemeliharaan model Anda, karena setiap perubahan yang Anda buat pada model komponen akan tercermin di semua model aset yang menggunakannya.

Misalnya, misalkan instalasi industri Anda memiliki banyak jenis peralatan yang semuanya menggunakan jenis motor servo yang sama. Beberapa dari mereka memiliki banyak motor servo dalam satu peralatan. Anda membuat model aset untuk setiap jenis peralatan, tetapi Anda tidak ingin menduplikasi definisi `servo` setiap saat. Anda ingin memodelkannya sekali saja dan menggunakannya dalam berbagai model aset Anda. Jika nanti Anda membuat perubahan pada definisi `servo`, itu akan diperbarui di semua model dan aset Anda.

Untuk memodelkan robot dari contoh sebelumnya dengan cara ini, Anda dapat mendefinisikan motor servo, catu daya, dan baterai sebagai model komponen, seperti ini.

- `servo_component_model`
 - `status`(bilangan bulat)
 - `position`(ganda)

- `powersupply_component_model`
 - `status`(bilangan bulat)
 - `temperature` (ganda)

- `battery__component_model`

- `status(bilangan bulat)`
- `charge(ganda)`

Anda kemudian dapat mendefinisikan model aset, seperti `robot_model`, yang mereferensikan komponen-komponen ini. Beberapa model aset dapat mereferensikan model komponen yang sama. Anda juga dapat mereferensikan model komponen yang sama beberapa kali dalam satu model aset, seperti jika robot Anda memiliki beberapa servomotor di dalamnya.

- `robot_model`
 - `servo1(referensi:servo_component_model)`
 - `servo2(referensi:servo_component_model)`
 - `servo3(referensi:servo_component_model)`
 - `powersupply (referensi:powersupply_component_model)`
 - `battery(referensi:battery_component_model)`

Untuk informasi tentang cara membuat model komponen, lihat [Membuat model komponen](#).

Untuk informasi tentang cara mereferensikan model komponen Anda di model lain, lihat [Membuat model komposit khusus \(Komponen\)](#).

Menggunakan jalur untuk mereferensikan properti model komposit khusus

Saat membuat properti pada model aset, model komponen, atau model komposit kustom, Anda dapat mereferensikannya dari properti lain yang menggunakan nilainya, seperti [transformasi](#) dan [metrik](#).

AWS IoT SiteWise menyediakan berbagai cara bagi Anda untuk mereferensikan properti Anda. Cara paling sederhana adalah sering menggunakan ID propertinya. Namun, jika properti yang ingin Anda referensikan ada pada model komposit khusus, Anda mungkin merasa lebih berguna untuk mereferensikannya berdasarkan jalur.

Path adalah urutan urutan segmen jalur yang menentukan properti dalam hal posisinya di antara model komposit bersarang dalam model aset dan model komposit.

Memperoleh jalur properti

Anda bisa mendapatkan jalur properti dari path bidangnya [AssetModelProperty](#).

Misalnya, Anda memiliki model aset `robot_model` yang berisi model komposit khusus servo, yang memiliki properti `position`. Jika Anda memanggil [DescribeAssetModelCompositeModel](#) servo, maka `position` properti akan mencantumkan path bidang yang terlihat seperti ini:

```
"path": [  
  {  
    "id": "asset model ID",  
    "name": "robot_model"  
  },  
  {  
    "id": "composite model ID",  
    "name": "servo"  
  },  
  {  
    "id": "property ID",  
    "name": "position"  
  }  
]
```

Menggunakan jalur properti

Anda dapat menggunakan jalur properti saat menentukan properti yang mereferensikan properti lain, seperti transformasi atau metrik.

Sebuah properti menggunakan variabel untuk referensi properti lain. Untuk informasi lebih lanjut tentang bekerja dengan variabel, lihat [Menggunakan variabel dalam ekspresi rumus](#).

Saat Anda mendefinisikan variabel untuk mereferensikan properti, Anda dapat menggunakan ID properti atau jalurnya.

Untuk menentukan variabel yang menggunakan jalur properti yang direferensikan, tentukan `propertyPath` bidang nilainya.

Misalnya, untuk menentukan model aset yang memiliki metrik yang mereferensikan properti menggunakan jalur, Anda dapat meneruskan payload seperti ini ke [CreateAssetModel](#):

```
{  
  ...  
  "assetModelProperties": [  
    {  
      ...  
    }  
  ]  
}
```

```
"type": {
  "metric": {
    ...
    "variables": [
      {
        "name": "variable name",
        "value": {
          "propertyPath": [
            path segments
          ]
        }
      }
    ],
    ...
  },
  ...
},
...
],
...
}
```

Bekerja dengan ID objek

AWS IoT SiteWise mendefinisikan berbagai jenis objek persisten, seperti aset, model aset, properti, dan hierarki. Semua objek tersebut memiliki pengidentifikasi unik yang dapat Anda gunakan untuk mengambil, memperbarui, dan menghapusnya.

AWS IoT SiteWise memiliki opsi berbeda untuk pelanggan untuk pembuatan ID. AWS IoT SiteWise menghasilkan satu untuk Anda secara default pada waktu pembuatan objek. Pengguna juga dapat memberikan ID mereka sendiri ke objek Anda.

Topik

- [Bekerja dengan objek UUID](#)
- [Menggunakan ID eksternal](#)

Bekerja dengan objek UUID

Setiap objek persisten AWS IoT SiteWise memiliki [UUID](#) untuk mengidentifikasinya. Misalnya, model aset memiliki ID model aset, aset memiliki ID aset, dan sebagainya. ID ini ditetapkan pada saat Anda membuat objek, dan tetap tidak berubah untuk masa pakai objek.

Saat Anda membuat objek baru, AWS IoT SiteWise buat ID unik untuk Anda secara default. Anda juga dapat memberikan ID Anda sendiri pada waktu pembuatan dalam format UUID.

Note

UUID harus unik secara global dalam AWS Wilayah tempat ia dibuat, dan untuk jenis objek yang sama. Ketika AWS IoT SiteWise otomatis membuat ID untuk Anda, itu selalu unik. Jika Anda memilih ID Anda sendiri, pastikan itu unik.

Misalnya, jika Anda membuat model aset baru dengan menelepon [CreateAssetModel](#), Anda dapat memberikan UUID Anda sendiri di `assetModelId` bidang opsional permintaan.

Sebaliknya, jika Anda menghilangkan `assetModelId` permintaan, AWS IoT SiteWise buat UUID untuk model aset baru.

Menggunakan ID eksternal

Untuk menentukan ID Anda sendiri dalam beberapa format selain UUID, Anda dapat menetapkan ID eksternal. Misalnya, Anda dapat melakukan ini jika Anda menggunakan kembali ID yang Anda gunakan dalam sistem yang tidak AWS, atau agar lebih mudah dibaca manusia. ID eksternal memiliki format yang lebih fleksibel. Anda dapat menggunakannya untuk mereferensikan objek Anda dalam operasi AWS IoT SiteWise API di mana Anda akan menggunakan UUID.

Seperti UUID, setiap ID eksternal harus unik dalam konteksnya. Misalnya, Anda tidak dapat memiliki dua model aset dengan ID eksternal yang sama. Juga, seperti UUID, objek hanya dapat memiliki satu ID eksternal dalam masa hidupnya, yang tidak dapat berubah.

Perbedaan antara ID eksternal dan UUID

ID eksternal berbeda dari UUID dengan cara berikut:

- Setiap objek memiliki UUID, tetapi ID eksternal bersifat opsional.

- AWS IoT SiteWise tidak pernah menghasilkan ID eksternal. Anda menyediakan ini sendiri.
- Jika objek belum memilikinya, Anda dapat menetapkan ID eksternal kapan saja.

Format ID eksternal

ID eksternal yang valid memiliki properti berikut:

- Panjangnya antara 2 dan 128 karakter.
- Karakter pertama dan terakhir harus alfanumerik (A-Z, a-z, 0-9).
- Karakter selain yang pertama dan terakhir harus alfanumerik, atau salah satu dari berikut ini: `_` `-` `.` `:`

Misalnya, ID eksternal harus sesuai dengan ekspresi reguler berikut:

```
[a-zA-Z0-9][a-zA-Z0-9_\-\. :]*[a-zA-Z0-9]+
```

Mereferensikan objek dengan ID eksternal

Di banyak tempat Anda dapat mereferensikan objek menggunakan UUID-nya, Anda dapat menggunakan ID eksternalnya sebagai gantinya, jika ada. Untuk melakukannya, tambahkan ID eksternal ke `stringexternalId`:

Misalnya, Anda memiliki model aset yang UUIDnya (ID model aset) `a1b2c3d4-5678-90ab-cdef-11111EXAMPLE`, yang juga memiliki ID eksternal. `myExternalId` Hubungi [DescribeAssetModel](#) untuk mendapatkan detail tentang hal itu. Anda dapat menggunakan salah satu dari berikut ini sebagai nilai `assetModelId`:

- Dengan ID model aset (UUID) itu sendiri: `a1b2c3d4-5678-90ab-cdef-11111EXAMPLE`
- Dengan ID eksternal: `externalId:myExternalId`

```
aws iotsitewise describe-asset-model --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
aws iotsitewise describe-asset-model --asset-model-id externalId:myExternalId
```

Note

`externalId`:Awalan itu sendiri bukan bagian dari ID eksternal. Anda hanya perlu memberikan awalan saat Anda menyediakan ID eksternal ke operasi API yang menerima

UUID atau ID eksternal. Misalnya, berikan awalan saat Anda menanyakan atau memperbarui objek yang ada.

Saat Anda menentukan ID eksternal untuk objek, seperti saat Anda membuat model aset, jangan sertakan awalan.

Anda dapat menggunakan ID eksternal sebagai pengganti UUID dengan cara ini untuk banyak operasi API di AWS IoT SiteWise, tetapi tidak semua. Misalnya, [GetAssetPropertyValue](#), harus menggunakan UUID; itu tidak mendukung penggunaan ID eksternal.

Untuk menentukan apakah operasi API tertentu mendukung penggunaan ini, lihat [Referensi API](#).

Membuat model aset dan model komponen

AWS IoT SiteWise model aset dan model komponen mendorong standardisasi data industri Anda. Model aset atau model komponen berisi nama, deskripsi, properti aset, dan (opsional) model komposit kustom yang mengelompokkan properti bersama-sama, atau model komponen referensi untuk subrakitan.

- Anda menggunakan model aset untuk membuat aset. Selain fitur yang tercantum di atas, model aset juga dapat berisi definisi hierarki yang mendefinisikan hubungan antar aset.
- Model komponen mewakili subassembly dalam model aset atau model komponen lain. Saat membuat model komponen, Anda dapat menambahkan referensi ke model aset dan model komponen lainnya. Namun, Anda tidak dapat membuat aset langsung dari model komponen.

Setelah Anda membuat model aset atau model komponen, Anda dapat membuat model komposit khusus untuk mengelompokkan properti bersama-sama atau untuk mereferensikan model komponen yang ada.

Untuk detail tentang cara membuat model aset dan model komponen, lihat bagian berikut.

Topik

- [Membuat model aset](#)
- [Membuat model komponen](#)
- [Mendefinisikan properti data](#)
- [Membuat model komposit khusus \(Komponen\)](#)

Membuat model aset

AWS IoT SiteWise model aset mendorong standardisasi data industri Anda. Model aset berisi nama, deskripsi, properti aset, dan definisi hierarki aset. Misalnya, Anda dapat menentukan model turbin angin dengan suhu, rotasi per menit (RPM), dan sifat daya. Kemudian, Anda dapat menentukan model ladang angin dengan properti output daya bersih dan definisi hierarki turbin angin.

Note

- Kami menyarankan Anda memodelkan operasi Anda dimulai dengan node tingkat terendah. Misalnya, buat model turbin angin Anda sebelum Anda membuat model ladang angin Anda. Definisi hierarki aset berisi referensi ke model aset yang ada. Dengan pendekatan ini, Anda dapat menentukan hierarki aset saat membuat model.
- Model aset tidak dapat berisi model aset lainnya. Jika Anda harus mendefinisikan model yang dapat Anda referensikan sebagai subassembly dalam model lain, Anda harus membuat komponen -> model sebagai gantinya. Untuk informasi selengkapnya, lihat [Membuat model komponen](#).

Bagian berikut menjelaskan cara menggunakan AWS IoT SiteWise konsol atau API untuk membuat model aset. Bagian berikut juga menjelaskan berbagai jenis properti aset dan hierarki aset yang dapat Anda gunakan untuk membuat model.

Topik

- [Membuat model aset \(konsol\)](#)
- [Membuat model aset \(AWS CLI\)](#)
- [Contoh model aset](#)
- [Mendefinisikan hierarki model aset](#)

Membuat model aset (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk membuat model aset. AWS IoT SiteWise Konsol menyediakan berbagai fitur, seperti formula auto completion, yang dapat membantu Anda menentukan model aset yang valid.

Untuk membuat model aset (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model.
3. Pilih Buat model.
4. Pada halaman Create model, lakukan hal berikut:
 - a. Masukkan Nama untuk model aset, seperti **Wind Turbine** atau **Wind Turbine Model**. Nama ini harus unik di semua model di akun Anda di Wilayah ini.
 - b. (Opsional) Tambahkan ID Eksternal untuk model. Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.
 - c. (Opsional) Tambahkan definisi Pengukuran untuk model. Pengukuran mewakili aliran data dari peralatan Anda. Untuk informasi selengkapnya, lihat [Mendefinisikan aliran data dari peralatan \(pengukuran\)](#).
 - d. (Opsional) Tambahkan definisi Transform untuk model. Transformasi adalah rumus yang memetakan data dari satu bentuk ke bentuk lainnya. Untuk informasi selengkapnya, lihat [Mengubah data \(transformasi\)](#).
 - e. (Opsional) Tambahkan definisi Metrik untuk model. Metrik adalah rumus yang mengumpulkan data dari interval waktu. Metrik dapat memasukkan data dari aset terkait, sehingga Anda dapat menghitung nilai yang mewakili operasi Anda atau bagian dari operasi Anda. Untuk informasi selengkapnya, lihat [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#).
 - f. (Opsional) Tambahkan definisi Hierarki untuk model. Hierarki adalah hubungan antar aset. Untuk informasi selengkapnya, lihat [Mendefinisikan hierarki model aset](#).
 - g. (Opsional) Tambahkan tag untuk model aset. Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda AWS IoT SiteWise](#).
 - h. Pilih Buat model.

Saat Anda membuat model aset, AWS IoT SiteWise konsol akan menavigasi ke halaman model baru. Pada halaman ini, Anda dapat melihat Status model, yang awalnya MENCIPTAKAN. Halaman ini diperbarui secara otomatis, sehingga Anda dapat menunggu status model diperbarui.

Note

Proses pembuatan model aset dapat memakan waktu hingga beberapa menit untuk model yang kompleks. Setelah status model aset AKTIF, Anda dapat menggunakan model aset untuk membuat aset. Untuk informasi selengkapnya, lihat [Status aset dan model](#).

5. (Opsional) Setelah membuat model aset, Anda dapat mengonfigurasi model aset untuk edge. Untuk informasi lebih lanjut tentang SiteWise Edge, lihat [Mengaktifkan pemrosesan data tepi](#).
 - a. Pada halaman model, pilih Configure for Edge.
 - b. Pada halaman konfigurasi model, pilih konfigurasi tepi untuk model Anda. Ini mengontrol tempat AWS IoT SiteWise dapat menghitung dan menyimpan properti yang terkait dengan model aset ini. Untuk informasi selengkapnya tentang mengonfigurasi model Anda untuk edge, lihat [the section called “Menyiapkan kemampuan tepi”](#).
 - c. Untuk konfigurasi Custom edge, pilih lokasi yang AWS IoT SiteWise ingin Anda hitung dan simpan setiap properti model aset Anda.

Note

Transformasi dan metrik yang terkait harus dikonfigurasi untuk lokasi yang sama. Untuk informasi selengkapnya tentang mengonfigurasi model Anda untuk edge, lihat [the section called “Menyiapkan kemampuan tepi”](#).

- d. Pilih Simpan. Pada halaman model, konfigurasi Edge Anda sekarang harus Dikonfigurasi.

Membuat model aset (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk membuat model aset.

Gunakan [CreateAssetModel](#) operasi untuk membuat model aset dengan properti dan hierarki. Operasi ini mengharapkan muatan dengan struktur berikut.

```
{
  "assetModelType": "ASSET_MODEL",
  "assetModelName": "String",
  "assetModelDescription": "String",
  "assetModelProperties": Array of AssetModelProperty,
```

```
"assetModelHierarchies": Array of AssetModelHierarchyDefinition
}
```

Untuk membuat model aset (AWS CLI)

1. Buat file bernama `asset-model-payload.json` dan kemudian salin objek JSON berikut ke dalam file.

```
{
  "assetModelType": "ASSET_MODEL",
  "assetModelName": "",
  "assetModelDescription": "",
  "assetModelProperties": [

  ],
  "assetModelHierarchies": [

  ],
  "assetModelCompositeModels": [

  ]
}
```

2. Gunakan editor teks JSON pilihan Anda untuk mengedit `asset-model-payload.json` file sebagai berikut:
 - a. Masukkan name (`assetModelName`) untuk model aset, seperti **Wind Turbine** atau **Wind Turbine Model**. Nama ini harus unik di semua model aset dan model komponen di akun Anda dalam hal ini Wilayah AWS.
 - b. (Opsional) Masukkan ID eksternal (`assetModelExternalId`) untuk model aset. Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.
 - c. (Opsional) Masukkan deskripsi (`assetModelDescription`) untuk model aset, atau hapus pasang `assetModelDescription` nilai kunci.
 - d. (Opsional) Tentukan properti aset (`assetModelProperties`) untuk model. Untuk informasi selengkapnya, lihat [Mendefinisikan properti data](#).
 - e. (Opsional) Tentukan hierarki aset (`assetModelHierarchies`) untuk model. Untuk informasi selengkapnya, lihat [Mendefinisikan hierarki model aset](#).

- f. (Opsional) Tentukan alarm untuk model. Alarm memantau properti lain sehingga Anda dapat mengidentifikasi kapan peralatan atau proses memerlukan perhatian. Setiap definisi alarm adalah model komposit (`assetModelCompositeModels`) yang menstandarisasi serangkaian properti yang digunakan alarm. Lihat informasi yang lebih lengkap di [Memantau data dengan alarm](#) dan [Mendefinisikan alarm pada model aset](#).
 - g. (Opsional) Tambahkan tag (`tags`) untuk model aset. Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda AWS IoT SiteWise](#).
3. Jalankan perintah berikut untuk membuat model aset dari definisi dalam file JSON.

```
aws iotsitewise create-asset-model --cli-input-json file://asset-model-payload.json
```

Operasi mengembalikan respons yang berisi `assetModelId` yang Anda rujuk saat membuat aset. Respons juga berisi status model (`assetModelStatus.state`), yang awalnya `CREATING`. Status model aset adalah `CREATING` sampai perubahan menyebar.

Note

Proses pembuatan model aset dapat memakan waktu hingga beberapa menit untuk model yang kompleks. Untuk memeriksa status model aset Anda saat ini, gunakan [DescribeAssetModel](#) operasi dengan menentukan `assetModelId`. Setelah status model aset `ACTIVE`, Anda dapat menggunakan model aset untuk membuat aset. Untuk informasi selengkapnya, lihat [Status aset dan model](#).

4. (Opsional) Buat model komposit khusus untuk model aset Anda. Dengan model komposit khusus, Anda dapat mengelompokkan properti dalam model, atau menyertakan subassembly dengan mereferensikan model komponen. Untuk informasi selengkapnya, lihat [Membuat model komposit khusus \(Komponen\)](#).

Contoh model aset

Bagian ini berisi contoh definisi model aset yang dapat Anda gunakan untuk membuat model aset dengan AWS CLI dan AWS IoT SiteWise SDK. Model aset ini mewakili turbin angin dan ladang angin. Aset turbin angin menelan data sensor mentah dan menghitung nilai seperti daya dan kecepatan angin rata-rata. Aset ladang angin menghitung nilai seperti daya total untuk semua turbin angin di ladang angin.

Topik

- [Model aset turbin angin](#)
- [Model aset peternakan angin](#)

Model aset turbin angin

Model aset berikut mewakili turbin di ladang angin. Turbin angin menelan data sensor untuk menghitung nilai seperti daya dan kecepatan angin rata-rata.

Note

Model contoh ini menyerupai model turbin angin dari AWS IoT SiteWise demo. Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT SiteWise demo](#).

```
{
  "assetModelType": "ASSET_MODEL",
  "assetModelName": "Wind Turbine Asset Model",
  "assetModelDescription": "Represents a turbine in a wind farm.",
  "assetModelProperties": [
    {
      "name": "Location",
      "dataType": "STRING",
      "type": {
        "attribute": {
          "defaultValue": "Renton"
        }
      }
    },
    {
      "name": "Make",
      "dataType": "STRING",
      "type": {
        "attribute": {
          "defaultValue": "Amazon"
        }
      }
    },
    {
      "name": "Model",
      "dataType": "INTEGER",
      "type": {
```

```
    "attribute": {
      "defaultValue": "500"
    }
  },
  {
    "name": "Torque (KiloNewton Meter)",
    "dataType": "DOUBLE",
    "unit": "kNm",
    "type": {
      "measurement": {}
    }
  },
  {
    "name": "Wind Direction",
    "dataType": "DOUBLE",
    "unit": "Degrees",
    "type": {
      "measurement": {}
    }
  },
  {
    "name": "RotationsPerMinute",
    "dataType": "DOUBLE",
    "unit": "RPM",
    "type": {
      "measurement": {}
    }
  },
  {
    "name": "Wind Speed",
    "dataType": "DOUBLE",
    "unit": "m/s",
    "type": {
      "measurement": {}
    }
  },
  {
    "name": "RotationsPerSecond",
    "dataType": "DOUBLE",
    "unit": "RPS",
    "type": {
      "transform": {
        "expression": "rpm / 60",

```

```

    "variables": [
      {
        "name": "rpm",
        "value": {
          "propertyId": "RotationsPerMinute"
        }
      }
    ]
  }
},
{
  "name": "Overdrive State",
  "dataType": "DOUBLE",
  "type": {
    "transform": {
      "expression": "gte(torque, 3)",
      "variables": [
        {
          "name": "torque",
          "value": {
            "propertyId": "Torque (KiloNewton Meter)"
          }
        }
      ]
    }
  }
},
{
  "name": "Average Power",
  "dataType": "DOUBLE",
  "unit": "Watts",
  "type": {
    "metric": {
      "expression": "avg(torque) * avg(rps) * 2 * 3.14",
      "variables": [
        {
          "name": "torque",
          "value": {
            "propertyId": "Torque (Newton Meter)"
          }
        },
        {
          "name": "rps",

```

```

        "value": {
            "propertyId": "RotationsPerSecond"
        }
    ],
    "window": {
        "tumbling": {
            "interval": "5m"
        }
    }
}
},
{
    "name": "Average Wind Speed",
    "dataType": "DOUBLE",
    "unit": "m/s",
    "type": {
        "metric": {
            "expression": "avg(windspeed)",
            "variables": [
                {
                    "name": "windspeed",
                    "value": {
                        "propertyId": "Wind Speed"
                    }
                }
            ],
            "window": {
                "tumbling": {
                    "interval": "5m"
                }
            }
        }
    }
},
{
    "name": "Torque (Newton Meter)",
    "dataType": "DOUBLE",
    "unit": "Nm",
    "type": {
        "transform": {
            "expression": "knm * 1000",
            "variables": [

```

```

        {
            "name": "knm",
            "value": {
                "propertyId": "Torque (KiloNewton Meter)"
            }
        }
    ]
}
},
{
    "name": "Overdrive State Time",
    "dataType": "DOUBLE",
    "unit": "Seconds",
    "type": {
        "metric": {
            "expression": "statetime(overdrive_state)",
            "variables": [
                {
                    "name": "overdrive_state",
                    "value": {
                        "propertyId": "Overdrive State"
                    }
                }
            ],
            "window": {
                "tumbling": {
                    "interval": "5m"
                }
            }
        }
    }
},
"assetModelHierarchies": []
}

```

Model aset peternakan angin

Model aset berikut mewakili ladang angin yang terdiri dari beberapa turbin angin. Model aset ini mendefinisikan [hierarki](#) model turbin angin. Hal ini memungkinkan ladang angin menghitung nilai (seperti daya rata-rata) dari data untuk semua turbin angin di ladang angin.

Note

Model contoh ini menyerupai model ladang angin dari AWS IoT SiteWise demo. Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT SiteWise demo](#).

Model aset ini tergantung pada [Model aset turbin angin](#). Ganti `propertyId` dan `childAssetModelId` nilai dengan yang berasal dari model aset turbin angin yang ada.

```
{
  "assetModelName": "Wind Farm Asset Model",
  "assetModelDescription": "Represents a wind farm.",
  "assetModelProperties": [
    {
      "name": "Code",
      "dataType": "INTEGER",
      "type": {
        "attribute": {
          "defaultValue": "300"
        }
      }
    },
    {
      "name": "Location",
      "dataType": "STRING",
      "type": {
        "attribute": {
          "defaultValue": "Renton"
        }
      }
    },
    {
      "name": "Reliability Manager",
      "dataType": "STRING",
      "type": {
        "attribute": {
          "defaultValue": "Mary Major"
        }
      }
    },
    {
      "name": "Total Overdrive State Time",
```

```

    "dataType": "DOUBLE",
    "unit": "seconds",
    "type": {
      "metric": {
        "expression": "sum(overdrive_state_time)",
        "variables": [
          {
            "name": "overdrive_state_time",
            "value": {
              "propertyId": "ID of Overdrive State Time property in Wind Turbine
Asset Model",
              "hierarchyId": "Turbine Asset Model"
            }
          }
        ],
        "window": {
          "tumbling": {
            "interval": "5m"
          }
        }
      }
    }
  },
  {
    "name": "Total Average Power",
    "dataType": "DOUBLE",
    "unit": "Watts",
    "type": {
      "metric": {
        "expression": "sum(turbine_avg_power)",
        "variables": [
          {
            "name": "turbine_avg_power",
            "value": {
              "propertyId": "ID of Average Power property in Wind Turbine Asset
Model",
              "hierarchyId": "Turbine Asset Model"
            }
          }
        ],
        "window": {
          "tumbling": {
            "interval": "5m"
          }
        }
      }
    }
  }
}

```

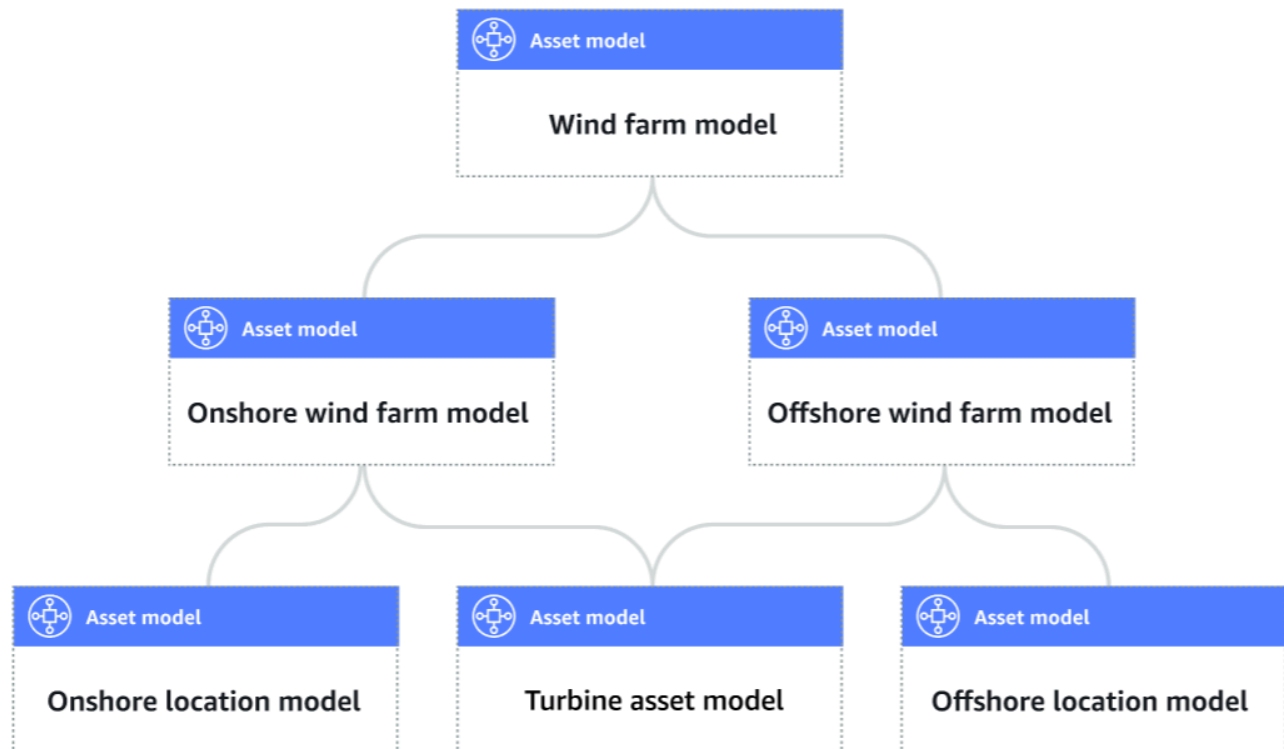
```
    }
  }
}
],
"assetModelHierarchies": [
  {
    "name": "Turbine Asset Model",
    "childAssetModelId": "ID of Wind Turbine Asset Model"
  }
]
}
```

Mendefinisikan hierarki model aset

Anda dapat menentukan hierarki model aset untuk membuat asosiasi logis antara model aset dalam operasi industri Anda. Misalnya, Anda dapat mendefinisikan ladang angin yang terdiri dari ladang angin darat dan lepas pantai. Sebuah ladang angin darat berisi turbin dan lokasi darat. Sebuah ladang angin lepas pantai berisi turbin dan lokasi lepas pantai.



Asset model hierarchy



Saat Anda mengaitkan model aset anak ke model aset induk melalui hierarki, metrik model aset induk dapat memasukkan data dari metrik model aset anak. Anda dapat menggunakan hierarki dan metrik model aset untuk menghitung statistik yang memberikan wawasan tentang operasi Anda atau subset operasi Anda. Untuk informasi selengkapnya, lihat [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#).

Setiap hierarki mendefinisikan hubungan antara model aset induk dan model aset anak yang sama. Dalam model aset induk, Anda dapat menentukan beberapa hierarki ke model aset anak yang sama. Misalnya, jika Anda memiliki dua jenis turbin angin yang berbeda di ladang angin Anda, di mana semua turbin angin diwakili oleh model aset yang sama, Anda dapat menentukan hierarki untuk setiap jenis. Kemudian, Anda dapat menentukan metrik dalam model ladang angin untuk menghitung statistik independen dan gabungan untuk setiap jenis turbin angin.

Model aset induk dapat dikaitkan dengan beberapa model aset anak. Misalnya, jika Anda memiliki ladang angin darat dan ladang angin lepas pantai yang diwakili oleh dua model aset yang berbeda, Anda dapat mengaitkan model aset ini dengan model aset ladang angin induk yang sama.

Model aset anak juga dapat dikaitkan dengan beberapa model aset induk. Misalnya, jika Anda memiliki dua jenis ladang angin yang berbeda, di mana semua turbin angin diwakili oleh model aset yang sama, Anda dapat mengaitkan model aset turbin angin dengan model aset ladang angin yang berbeda.

Note

Saat Anda menentukan hierarki model aset, model aset anak harus ACTIVE atau memiliki ACTIVE versi sebelumnya. Untuk informasi selengkapnya, lihat [Status aset dan model](#).

Setelah menentukan model aset hierarkis dan membuat aset, Anda dapat mengaitkan aset untuk menyelesaikan hubungan induk-anak. Lihat informasi yang lebih lengkap di [Menciptakan aset](#) dan [Mengasosiasikan dan memisahkan aset](#).

Topik

- [Mendefinisikan hierarki model aset \(konsol\)](#)
- [Mendefinisikan hierarki aset \(AWS CLI\)](#)

Mendefinisikan hierarki model aset (konsol)

Saat menentukan hierarki untuk model aset di AWS IoT SiteWise konsol, Anda menentukan parameter berikut:

- Nama hierarki — Nama hierarki, seperti. **Wind Turbines**
- Model hierarki — Model aset anak.
- Hierarki ID Eksternal (Opsional) - Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

Untuk informasi selengkapnya, lihat [Membuat model aset \(konsol\)](#).

Mendefinisikan hierarki aset ()AWS CLI

Saat Anda menentukan hierarki untuk model aset dengan AWS IoT SiteWise API, Anda menentukan parameter berikut:

- **name**— Nama hierarki, seperti **Wind Turbines**.
- **childAssetModelId**— ID atau ID eksternal model aset anak untuk hierarki. Anda dapat menggunakan [ListAssetModels](#) operasi untuk menemukan ID model aset yang ada.

Example Contoh definisi hierarki

Contoh berikut menunjukkan hierarki model aset yang mewakili hubungan ladang angin dengan turbin angin. Objek ini adalah contoh dari sebuah [AssetModelHierarchy](#). Untuk informasi selengkapnya, lihat [Membuat model aset \(AWS CLI\)](#).

```
{
  ...
  "assetModelHierarchies": [
    {
      "name": "Wind Turbines",
      "childAssetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
    },
  ],
}
```

Membuat model komponen

Gunakan model AWS IoT SiteWise komponen untuk menentukan subrakitan yang dapat Anda referensikan dari model aset atau model komponen lainnya. Dengan cara ini, Anda dapat menggunakan kembali definisi komponen di beberapa model lain, atau beberapa kali dalam model yang sama.

Proses mendefinisikan model komponen sangat mirip dengan mendefinisikan model aset. Seperti model aset, model komponen memiliki nama, deskripsi, dan properti aset. Namun, model komponen tidak dapat menyertakan definisi hierarki aset, karena model komponen itu sendiri tidak dapat digunakan untuk membuat aset secara langsung. Model komponen juga tidak dapat mendefinisikan alarm.

Misalnya, Anda dapat menentukan komponen untuk motor servo dengan suhu motor, suhu encoder, dan sifat resistansi isolasi. Kemudian, Anda dapat menentukan model aset untuk peralatan yang berisi motor servo, seperti mesin CNC.

Note

- Kami menyarankan Anda memodelkan operasi Anda dimulai dengan node tingkat terendah. Misalnya, buat komponen motor servo Anda sebelum Anda membuat model aset mesin CNC Anda. Model aset berisi referensi ke model komponen yang ada.
- Anda tidak dapat membuat aset langsung dari model komponen. Untuk membuat aset yang menggunakan komponen Anda, Anda harus membuat model aset untuk aset Anda. Kemudian, Anda membuat model komposit khusus untuk itu yang mereferensikan komponen Anda. Untuk informasi selengkapnya tentang membuat model aset, lihat [Membuat model aset](#) Untuk informasi selengkapnya tentang membuat model komposit kustom, lihat [Membuat model komposit khusus \(Komponen\)](#).

Bagian berikut menjelaskan cara menggunakan AWS IoT SiteWise API untuk membuat model komponen.

Topik

- [Membuat model komponen \(AWS CLI\)](#)
- [Contoh model komponen](#)

Membuat model komponen (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk membuat model komponen.

Gunakan [CreateAssetModel](#) operasi untuk membuat model komponen dengan properti. Operasi ini mengharapkan muatan dengan struktur berikut:

```
{
  "assetModelType": "COMPONENT_MODEL",
  "assetModelName": "String",
  "assetModelDescription": "String",
  "assetModelProperties": Array of AssetModelProperty,
```

```
}
```

Untuk membuat model komponen (AWS CLI)

1. Buat file bernama `component-model-payload.json` dan kemudian salin objek JSON berikut ke dalam file:

```
{
  "assetModelType": "COMPONENT_MODEL",
  "assetModelName": "",
  "assetModelDescription": "",
  "assetModelProperties": [

]
}
```

2. Gunakan editor teks JSON pilihan Anda untuk mengedit `component-model-payload.json` file sebagai berikut:
 - a. Masukkan nama (`assetModelName`) untuk model komponen, seperti **Servo Motor** atau **Servo Motor Model**. Nama ini harus unik di semua model aset dan model komponen di akun Anda dalam hal ini Wilayah AWS.
 - b. (Opsional) Masukkan ID eksternal (`assetModelExternalId`) untuk model komponen. Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.
 - c. (Opsional) Masukkan deskripsi (`assetModelDescription`) untuk model aset, atau hapus pasangan `assetModelDescription` nilai kunci.
 - d. (Opsional) Tentukan properti aset (`assetModelProperties`) untuk model komponen. Untuk informasi selengkapnya, lihat [Mendefinisikan properti data](#).
 - e. (Opsional) Tambahkan tag (`tags`) untuk model aset. Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda AWS IoT SiteWise](#).
3. Jalankan perintah berikut untuk membuat model komponen dari definisi dalam file JSON.

```
aws iotsitewise create-asset-model --cli-input-json file://component-model-payload.json
```

Operasi mengembalikan respons yang berisi `assetModelId` yang Anda rujuk saat menambahkan referensi ke model komponen Anda dalam model aset atau model komponen

lain. Respons juga berisi status model (`assetModelStatus.state`), yang awalnya `CREATING`. Status model komponen adalah `CREATING` sampai perubahan menyebar.

Note

Proses pembuatan model komponen dapat memakan waktu hingga beberapa menit untuk model yang kompleks. Untuk memeriksa status model komponen Anda saat ini, gunakan [DescribeAssetModel](#) operasi dengan menentukan `assetModelId`. Setelah status model komponen `ACTIVE`, Anda dapat menambahkan referensi ke model komponen Anda dalam model aset atau model komponen lainnya. Untuk informasi selengkapnya, lihat [Status aset dan model](#).

4. (Opsional) Buat model komposit khusus untuk model komponen Anda. Dengan model komposit khusus, Anda dapat mengelompokkan properti dalam model, atau menyertakan subassembly dengan mereferensikan model komponen lain. Untuk informasi selengkapnya, lihat [Membuat model komposit khusus \(Komponen\)](#).

Contoh model komponen

Bagian ini berisi contoh definisi model komponen yang dapat Anda gunakan untuk membuat model komponen dengan AWS IoT SiteWise SDK AWS CLI dan. Model komponen ini mewakili motor servo yang dapat digunakan dalam peralatan lain, seperti mesin CNC.

Topik

- [Model komponen motor servo](#)

Model komponen motor servo

Model komponen berikut mewakili motor servo yang dapat digunakan dalam peralatan seperti mesin CNC. Motor servo menyediakan berbagai pengukuran, seperti suhu dan hambatan listrik. Pengukuran ini tersedia sebagai properti pada aset yang dibuat dari model aset yang merujuk pada model komponen motor servo.

```
{
  "assetModelName": "ServoMotor",
  "assetModelType": "COMPONENT_MODEL",
  "assetModelProperties": [
    {
```

```
    "dataType": "DOUBLE",
    "name": "Servo Motor Temperature",
    "type": {
      "measurement": {}
    },
    "unit": "Celsius"
  },
  {
    "dataType": "DOUBLE",
    "name": "Spindle speed",
    "type": {
      "measurement": {}
    },
    "unit": "rpm"
  }
]
```

Mendefinisikan properti data

Properti aset adalah struktur dalam setiap aset yang berisi data aset. Properti aset dapat berupa salah satu dari jenis berikut:

- **Atribut** — Properti aset yang umumnya statis, seperti produsen perangkat atau wilayah geografis. Untuk informasi selengkapnya, lihat [Mendefinisikan data statis \(atribut\)](#).
- **Pengukuran** — Aliran data sensor perangkat mentah aset, seperti nilai kecepatan rotasi stempel waktu atau nilai suhu stempel waktu dalam Celcius. Pengukuran didefinisikan oleh alias aliran data. Untuk informasi selengkapnya, lihat [Mendefinisikan aliran data dari peralatan \(pengukuran\)](#).
- **Transformasi** — Nilai deret waktu yang ditransformasikan aset, seperti nilai suhu stempel waktu di Fahrenheit. Transformasi didefinisikan oleh ekspresi dan variabel untuk dikonsumsi dengan ekspresi itu. Untuk informasi selengkapnya, lihat [Mengubah data \(transformasi\)](#).
- **Metrik** — Data aset yang dikumpulkan selama interval waktu tertentu, seperti suhu rata-rata per jam. Metrik didefinisikan oleh interval waktu, ekspresi, dan variabel untuk dikonsumsi dengan ekspresi itu. Ekspresi metrik dapat memasukkan properti metrik aset terkait, sehingga Anda dapat menghitung metrik yang mewakili operasi Anda atau subset operasi Anda. Untuk informasi selengkapnya, lihat [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#).

Untuk informasi selengkapnya, lihat [Membuat model aset](#).

Untuk contoh cara menggunakan pengukuran, transformasi, dan metrik untuk menghitung Overall Equipment Effectivity (OEE), lihat. [Menghitung OEE di AWS IoT SiteWise](#)

Topik

- [Mendefinisikan data statis \(atribut\)](#)
- [Mendefinisikan aliran data dari peralatan \(pengukuran\)](#)
- [Mengubah data \(transformasi\)](#)
- [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#)
- [Menggunakan ekspresi rumus](#)

Mendefinisikan data statis (atribut)

Atribut aset mewakili informasi yang umumnya statis, seperti produsen perangkat atau lokasi geografis. Setiap aset yang Anda buat dari model aset berisi atribut model tersebut.

Topik

- [Mendefinisikan atribut \(konsol\)](#)
- [Mendefinisikan atribut \(AWS CLI\)](#)

Mendefinisikan atribut (konsol)

Saat menentukan atribut untuk model aset di AWS IoT SiteWise konsol, Anda menentukan parameter berikut:

- Nama - Nama properti.
- Nilai default — (Opsional) Nilai default untuk atribut ini. Aset yang dibuat dari model memiliki nilai ini untuk atribut. Untuk informasi selengkapnya tentang cara mengganti nilai default dalam aset yang dibuat dari model, lihat [Memperbarui nilai atribut](#).
- Tipe data — Tipe data properti, yang merupakan salah satu dari berikut ini:
 - String — Sebuah string dengan hingga 1024 byte.
 - Integer — Bilangan bulat 32-bit yang ditandatangani dengan rentang [-2,147,483.648, 2.147,483.647].
 - Ganda — Nomor floating point dengan jangkauan [-10¹⁰⁰, 10¹⁰⁰] dan presisi ganda IEEE 754.
 - Boolean — `true` atau `false`.

- ID Eksternal — (Opsional) Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

Untuk informasi selengkapnya, lihat [Membuat model aset \(konsol\)](#).

Mendefinisikan atribut (AWS CLI)

Saat menentukan atribut untuk model aset dengan AWS IoT SiteWise API, Anda menentukan parameter berikut:

- name- Nama properti.
- defaultVaLue— (Opsional) Nilai default untuk atribut ini. Aset yang dibuat dari model memiliki nilai ini untuk atribut. Untuk informasi selengkapnya tentang cara mengganti nilai default dalam aset yang dibuat dari model, lihat [Memperbarui nilai atribut](#).
- dataTipe— Tipe data properti, yang merupakan salah satu dari berikut ini:
 - STRING— Sebuah string dengan hingga 1024 byte.
 - INTEGER— Bilangan bulat 32-bit yang ditandatangani dengan rentang [-2,147.483.648, 2.147.483.647].
 - DOUBLE— Nomor floating point dengan jangkauan [-10¹⁰⁰, 10¹⁰⁰] dan presisi ganda IEEE 754.
 - BOOLEAN— true atau false.
- externalId— (Opsional) Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

Example Contoh definisi atribut

Contoh berikut menunjukkan atribut yang mewakili nomor model aset dengan nilai default. Objek ini adalah contoh dari sebuah [AssetModelProperty](#) yang berisi [Atribut](#). Anda dapat menentukan objek ini sebagai bagian dari payload [CreateAssetModel](#) permintaan untuk membuat properti atribut. Untuk informasi selengkapnya, lihat [Membuat model aset \(AWS CLI\)](#).

```
{
  ...
  "assetModelProperty": [
    {
```

```
"name": "Model number",
"dataType": "STRING",
"type": {
  "attribute": {
    "defaultValue": "BLT123"
  }
}
],
...
}
```

Mendefinisikan aliran data dari peralatan (pengukuran)

Pengukuran mewakili aliran data sensor mentah perangkat, seperti nilai suhu stempel waktu atau nilai rotasi per menit (RPM) stempel waktu.

Topik

- [Mendefinisikan pengukuran \(konsol\)](#)
- [Mendefinisikan pengukuran \(\)AWS CLI](#)

Mendefinisikan pengukuran (konsol)

Saat menentukan pengukuran untuk model aset di AWS IoT SiteWise konsol, Anda menentukan parameter berikut:

- Nama - Nama properti.
- Unit — (Opsional) Unit ilmiah untuk properti, seperti mm atau Celcius.
- Tipe data — Tipe data properti, yang merupakan salah satu dari berikut ini:
 - String — Sebuah string dengan hingga 1024 byte.
 - Integer — Bilangan bulat 32-bit yang ditandatangani dengan rentang [-2,147,483.648, 2.147,483.647].
 - Ganda — Nomor floating point dengan jangkauan [-10¹⁰⁰, 10¹⁰⁰] dan presisi ganda IEEE 754.
 - Boolean — true atau false.
- ID Eksternal — (Opsional) Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

Untuk informasi selengkapnya, lihat [Membuat model aset \(konsol\)](#).

Mendefinisikan pengukuran (AWS CLI)

Saat menentukan pengukuran untuk model aset dengan AWS IoT SiteWise API, Anda menentukan parameter berikut:

- **name**— Nama properti.
- **dataType**— Tipe data properti, yang merupakan salah satu dari berikut ini:
 - **STRING**— Sebuah string dengan hingga 1024 byte.
 - **INTEGER**— Bilangan bulat 32-bit yang ditandatangani dengan rentang [-2,147.483.648, 2.147.483.647].
 - **DOUBLE**— Nomor floating point dengan jangkauan [-10¹⁰⁰, 10¹⁰⁰] dan presisi ganda IEEE 754.
 - **BOOLEAN**— `true` atau `false`.
- **unit**— (Opsional) Unit ilmiah untuk properti, seperti mm atau Celcius.
- **externalId**— (Opsional) Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

Example Contoh definisi pengukuran

Contoh berikut menunjukkan pengukuran yang mewakili pembacaan sensor suhu aset. Objek ini adalah contoh dari sebuah [AssetModelProperty](#) yang berisi [Pengukuran](#). Anda dapat menentukan objek ini sebagai bagian dari payload [CreateAssetModel](#) permintaan untuk membuat properti pengukuran. Untuk informasi selengkapnya, lihat [Membuat model aset \(AWS CLI\)](#).

Struktur [Pengukuran](#) adalah struktur kosong saat Anda menentukan model aset karena nanti Anda mengonfigurasi setiap aset untuk menggunakan aliran data perangkat yang unik. Untuk informasi selengkapnya tentang cara menghubungkan properti pengukuran aset ke aliran data sensor perangkat, lihat [Memetakan aliran data industri ke properti aset](#).

```
{
  ...
  "assetModelProperties": [
    {
      "name": "Temperature C",
      "dataType": "DOUBLE",
```

```
    "type": {
      "measurement": {}
    },
    "unit": "Celsius"
  }
],
  ...
}
```

Mengubah data (transformasi)

Transformasi adalah ekspresi matematika yang memetakan titik data properti aset dari satu bentuk ke bentuk lainnya. Ekspresi transformasi terdiri dari variabel properti aset, literal, operator, dan fungsi. Titik data yang ditransformasikan memiliki one-to-one hubungan dengan titik data input. AWS IoT SiteWise menghitung titik data baru yang diubah setiap kali salah satu properti input menerima titik data baru.


Misalnya, jika aset Anda memiliki aliran pengukuran suhu yang diberi nama `Temperature_C` dengan satuan dalam Celcius, Anda dapat mengonversi setiap titik data menjadi Fahrenheit dengan rumus. $Temperature_F = 9/5 * Temperature_C + 32$ Setiap kali AWS IoT SiteWise menerima titik data dalam aliran `Temperature_C` pengukuran, `Temperature_F` nilai yang sesuai dihitung dalam beberapa detik dan tersedia sebagai `Temperature_F` properti.

Jika transformasi Anda berisi lebih dari satu variabel, titik data yang tiba lebih awal segera memulai perhitungan. Pertimbangkan contoh di mana produsen suku cadang menggunakan transformasi untuk memantau kualitas produk. Menggunakan standar yang berbeda berdasarkan jenis bagian, pabrikan menggunakan pengukuran berikut untuk mewakili proses:

- `Part_Number`- Sebuah string yang mengidentifikasi tipe bagian.
- `Good_Count`- Bilangan bulat yang bertambah satu jika bagian tersebut memenuhi standar.
- `Bad_Count`- Bilangan bulat yang bertambah satu jika bagian tidak memenuhi standar.

Pabrikan juga menciptakan transformasi, `Quality_Monitor`, yang sama. `if(eq(Part_Number, "BLT123") and (Bad_Count / (Good_Count + Bad_Count) > 0.1), "Caution", "Normal")`

Transformasi ini memantau persentase bagian buruk yang diproduksi untuk jenis bagian tertentu. Jika nomor bagian adalah BLT123 dan persentase bagian yang buruk melebihi 10 persen (0,1), transformasi kembali. "Caution" Jika tidak, transformasi akan kembali "Normal".

 Note

- Jika `Part_Number` menerima titik data baru sebelum pengukuran lain, `Quality_Monitor` transformasi menggunakan `Part_Number` nilai baru `Good_Count` dan `Bad_Count` nilai terbaru. Untuk menghindari kesalahan, setel ulang `Good_Count` dan `Bad_Count` sebelum pembuatan berikutnya dijalankan.
- Gunakan [metrik](#) jika Anda ingin mengevaluasi ekspresi hanya setelah semua variabel menerima titik data baru.


Topik

- [Mendefinisikan transformasi \(konsol\)](#)
- [Mendefinisikan transformasi \(\)AWS CLI](#)

Mendefinisikan transformasi (konsol)

Saat menentukan transformasi untuk model aset di AWS IoT SiteWise konsol, Anda menentukan parameter berikut:

- Nama - Nama properti.
- Unit — (Opsional) Unit ilmiah untuk properti, seperti mm atau Celcius.
- Tipe data — Tipe data dari transformasi, yang dapat berupa Double atau String.
- ID Eksternal — (Opsional) Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.
- Formula - Ekspresi transformasi. Ekspresi transformasi tidak dapat menggunakan fungsi agregasi atau fungsi temporal. Untuk membuka fitur auto complete, mulai mengetik atau tekan tombol panah bawah. Untuk informasi selengkapnya, lihat [Menggunakan ekspresi rumus](#).

 Important

Transformasi dapat memasukkan properti yang integer, ganda, Boolean, atau tipe string. Boolean mengkonversi ke 0 (false) dan 1 (true).

Transformasi harus memasukkan satu atau lebih properti yang bukan atribut dan sejumlah properti atribut. AWS IoT SiteWise menghitung titik data baru yang diubah setiap kali properti input yang bukan atribut menerima titik data baru. Nilai atribut baru tidak

meluncurkan pembaruan transformasi. Tingkat permintaan yang sama untuk operasi API data properti aset berlaku untuk hasil komputasi transformasi.

Ekspresi rumus hanya dapat menampilkan nilai ganda atau string. Ekspresi bersarang dapat menampilkan tipe data lain, seperti string, tetapi rumus secara keseluruhan harus mengevaluasi ke angka atau string. Anda dapat menggunakan [fungsi jp](#) untuk mengonversi string ke angka. Nilai Boolean harus 1 (true) atau 0 (false). Untuk informasi selengkapnya, lihat [Nilai tidak terdefinisi, tak terbatas, dan meluap](#).

Untuk informasi selengkapnya, lihat [Membuat model aset \(konsol\)](#).

Mendefinisikan transformasi ()AWS CLI

Saat menentukan transformasi untuk model aset dengan AWS IoT SiteWise API, Anda menentukan parameter berikut:

- `name`— Nama properti.
- `unit`— (Opsional) Unit ilmiah untuk properti, seperti mm atau Celcius.
- `dataType`— Tipe data dari transformasi, yang harus DOUBLE atau STRING.
- `externalId`— (Opsional) Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.
- `expression`— Ekspresi transformasi. Ekspresi transformasi tidak dapat menggunakan fungsi agregasi atau fungsi temporal. Untuk informasi selengkapnya, lihat [Menggunakan ekspresi rumus](#).
- `variables`— Daftar variabel yang mendefinisikan properti lain dari aset Anda untuk digunakan dalam ekspresi. Setiap struktur variabel berisi nama sederhana untuk digunakan dalam ekspresi dan `value` struktur yang mengidentifikasi properti mana yang akan ditautkan ke variabel itu. `value` Struktur berisi informasi berikut:
 - `propertyId`— ID properti dari mana untuk memasukkan nilai. Anda dapat menggunakan nama properti alih-alih ID-nya.

Important

Transformasi dapat memasukkan properti yang integer, ganda, Boolean, atau tipe string. Boolean mengkonversi ke 0 (false) dan 1 (true).

Transformasi harus memasukkan satu atau lebih properti yang bukan atribut dan sejumlah properti atribut. AWS IoT SiteWise menghitung titik data baru yang diubah setiap

kali properti input yang bukan atribut menerima titik data baru. Nilai atribut baru tidak meluncurkan pembaruan transformasi. Tingkat permintaan yang sama untuk operasi API data properti aset berlaku untuk hasil komputasi transformasi.

Ekspresi rumus hanya dapat menampilkan nilai ganda atau string. Ekspresi bersarang dapat menampilkan tipe data lain, seperti string, tetapi rumus secara keseluruhan harus mengevaluasi ke angka atau string. Anda dapat menggunakan [fungsi jp](#) untuk mengonversi string ke angka. Nilai Boolean harus 1 (true) atau 0 (false). Untuk informasi selengkapnya, lihat [Nilai tidak terdefinisi, tak terbatas, dan meluap](#).

Example mengubah definisi

Contoh berikut menunjukkan properti transformasi yang mengubah data pengukuran suhu aset dari Celcius ke Fahrenheit. Objek ini adalah contoh dari sebuah [AssetModelProperty](#) yang berisi [Transform](#). Anda dapat menentukan objek ini sebagai bagian dari payload [CreateAssetModel](#) permintaan untuk membuat properti transform. Untuk informasi selengkapnya, lihat [Membuat model aset \(AWS CLI\)](#).

```
{
  ...
  "assetModelProperty": [
    ...
    {
      "name": "Temperature F",
      "dataType": "DOUBLE",
      "type": {
        "transform": {
          "expression": "9/5 * temp_c + 32",
          "variables": [
            {
              "name": "temp_c",
              "value": {
                "propertyId": "Temperature C"
              }
            }
          ]
        }
      },
      "unit": "Fahrenheit"
    }
  ],
}
```

```

...
}

```

Example mengubah definisi yang berisi tiga variabel

Contoh berikut menunjukkan properti transform yang mengembalikan pesan peringatan ("Caution") jika lebih dari 10 persen bagian BLT123 tidak memenuhi standar. Jika tidak, ia mengembalikan pesan informasi ("Normal").

```

{
  ...
  "assetModelProperties": [
    ...
    {
      "name": "Quality_Monitor",
      "dataType": "STRING",
      "type": {
        "transform": {
          "expression": "if(eq(Part_Number,\"BLT123\") and (Bad_Count / (Good_Count +
Bad_Count) > 0.1), \"Caution\", \"Normal\")",
          "variables": [
            {
              "name": "Part_Number",
              "value": {
                "propertyId": "Part Number"
              }
            },
            {
              "name": "Good_Count",
              "value": {
                "propertyId": "Good Count"
              }
            },
            {
              "name": "Bad_Count",
              "value": {
                "propertyId": "Bad Count"
              }
            }
          ]
        }
      }
    }
  ]
}
}

```



```
...  
}
```

Menggabungkan data dari properti dan aset lainnya (metrik)

Metrik adalah ekspresi matematika yang menggunakan fungsi agregasi untuk memproses semua titik data input dan output satu titik data per interval waktu tertentu. Misalnya, metrik dapat menghitung suhu rata-rata per jam dari aliran data suhu.

Metrik dapat memasukkan data dari metrik aset terkait, sehingga Anda dapat menghitung statistik yang memberikan wawasan tentang operasi atau subset operasi Anda. Misalnya, metrik dapat menghitung suhu rata-rata per jam di semua turbin angin di ladang angin. Untuk informasi selengkapnya tentang cara mendefinisikan asosiasi antar aset, lihat [Mendefinisikan hierarki model aset](#).

Metrik juga dapat memasukkan data dari properti lain tanpa menggabungkan data setiap interval waktu. Jika Anda menentukan [atribut](#) dalam rumus, AWS IoT SiteWise gunakan nilai [terbaru](#) untuk atribut tersebut saat menghitung rumus. Jika Anda menentukan metrik dalam rumus, AWS IoT SiteWise gunakan nilai [terakhir](#) untuk interval waktu di mana ia menghitung rumus. Ini berarti Anda dapat menentukan metrik seperti $OEE = Availability * Quality * Performance$, di mana *Availability*, *Quality*, dan *Performance* semua metrik lain pada model aset yang sama.

AWS IoT SiteWise juga secara otomatis menghitung satu set metrik agregasi dasar untuk semua properti aset. Untuk mengurangi biaya komputasi, Anda dapat menggunakan agregat ini alih-alih mendefinisikan metrik khusus untuk perhitungan dasar. Untuk informasi selengkapnya, lihat [Kueri agregat properti aset](#).

Topik

- [Mendefinisikan metrik \(konsol\)](#)
- [Mendefinisikan metrik \(\)AWS CLI](#)

Mendefinisikan metrik (konsol)

Saat menentukan metrik untuk model aset di AWS IoT SiteWise konsol, Anda menentukan parameter berikut:

- Nama - Nama properti.
- Tipe data — Tipe data dari transformasi, yang dapat berupa Double atau String.

- ID Eksternal — (Opsional) Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.
- Rumus — Ekspresi metrik. Ekspresi metrik dapat menggunakan [fungsi agregasi](#) untuk memasukkan data dari properti untuk semua aset terkait dalam hierarki. Mulai mengetik atau tekan tombol panah bawah untuk membuka fitur auto complete. Untuk informasi selengkapnya, lihat [Menggunakan ekspresi rumus](#).

Important

Metrik hanya dapat berupa properti yang integer, double, Boolean, atau tipe string. Boolean mengkonversi ke 0 (false) dan 1 (true).

Jika Anda menentukan variabel input metrik apa pun dalam ekspresi metrik, input tersebut harus memiliki interval waktu yang sama dengan metrik keluaran.

Ekspresi rumus hanya dapat menampilkan nilai ganda atau string. Ekspresi bersarang dapat menampilkan tipe data lain, seperti string, tetapi rumus secara keseluruhan harus mengevaluasi ke angka atau string. Anda dapat menggunakan [fungsi jp](#) untuk mengonversi string ke angka. Nilai Boolean harus 1 (true) atau 0 (false). Untuk informasi selengkapnya, lihat [Nilai tidak terdefinisi, tak terbatas, dan meluap](#).

- Interval waktu — Interval waktu metrik. AWS IoT SiteWise mendukung interval waktu jendela jatuh berikut, di mana setiap interval dimulai ketika yang sebelumnya berakhir:
 - 1 menit — 1 menit, dihitung pada akhir setiap menit (12:00:00 AM, 12:01:00 AM, 12:02:00 AM, dan seterusnya).
 - 5 menit — 5 menit, dihitung pada akhir setiap lima menit dimulai pada jam (12:00:00 AM, 12:05:00 AM, 12:10:00 AM, dan seterusnya).
 - 15 menit — 15 menit, dihitung pada akhir setiap lima belas menit dimulai pada jam (12:00:00 AM, 12:15:00 AM, 12:30:00 AM, dan seterusnya).
 - 1 jam — 1 jam (60 menit), dihitung pada akhir setiap jam di UTC (12:00:00 AM, 01:00:00 AM, 02:00:00 AM, dan seterusnya).
 - 1 hari — 1 hari (24 jam), dihitung pada akhir setiap hari di UTC (12:00:00 Senin, 12:00:00 Selasa, dan seterusnya).
 - 1 minggu — 1 minggu (7 hari), dihitung pada akhir setiap hari Minggu di UTC (setiap 12:00:00 Senin).
 - Interval khusus - Anda dapat memasukkan interval waktu antara satu menit dan seminggu.
- Tanggal offset — (Opsional) Tanggal referensi untuk mengumpulkan data.

- Waktu offset — (Opsional) Waktu referensi untuk mengumpulkan data. Waktu offset harus antara 00:00:00 dan 23:59:59.
- Offset zona waktu - (Opsional) Zona waktu untuk offset. Jika tidak ditentukan, zona waktu offset default adalah Universal Coordinated Time (UTC).

Zona waktu yang didukung

- (UTC+ 00:00) Waktu Terkoordinasi Universal
- (UTC+ 01:00) Waktu Tengah Eropa
- (UTC+ 02:00) Eropa Timur
- (UTC03+00) Waktu Afrika Timur
- (UTC+ 04:00) Waktu Timur Dekat
- (UTC+ 05:00) Waktu Lahore Pakistan
- (UTC+ 05:30) Waktu Standar India
- (UTC+ 06:00) Waktu Standar Bangladesh
- (UTC+ 07:00) Waktu Standar Vietnam
-
- (UTC+09:00) Waktu Standar Jepang
- (UTC+ 09:30) Waktu Tengah Australia
- (UTC+10:00) Waktu Timur Australia
- (UTC+ 11:00) Waktu Standar Solomon
- (UTC+12:00) Waktu Standar Selandia Baru
- (UTC- 11:00) Waktu Kepulauan Midway
- (UTC- 10:00) Waktu Standar Hawaii
- (UTC- 09:00) Waktu Standar Alaska
- (UTC- 08:00) Waktu Standar Pasifik
- (UTC- 07:00) Waktu Standar Phoenix
- (UTC- 06:00) Waktu Standar Pusat
- (UTC- 05:00) Waktu Standar Timur
- (UTC- 04:00) Puerto Riko dan Waktu Kepulauan Virgin AS
- (UTC- 03:00) Waktu Standar Argentina
- (UTC- 02:00) Waktu Georgia Selatan


- (UTC- 01:00) Waktu Afrika Tengah

Example interval waktu khusus dengan offset (konsol)

Contoh berikut menunjukkan kepada Anda cara menentukan interval waktu 12 jam dengan offset pada 20 Februari 2021, pukul 18:30:30 (PST).

Untuk menentukan interval kustom dengan offset

1. Untuk Interval waktu, pilih Interval khusus.
2. Untuk Interval waktu, lakukan salah satu hal berikut:
 - Masuk **12**, lalu pilih jam.
 - Masuk **720**, lalu pilih menit.
 - Masukkan **43200**, lalu pilih detik.

 Important

Interval waktu harus berupa bilangan bulat terlepas dari unit.

3. Untuk tanggal Offset, pilih 2021/02/20.
4. Untuk waktu Offset, masukkan **18:30:30**.
5. Untuk zona waktu Offset, pilih (UTC- 08:00) Waktu Standar Pasifik.

Jika Anda membuat metrik pada 1 Juli 2021, sebelum atau pukul 18:30:30 (PST), Anda mendapatkan hasil agregasi pertama pada 1 Juli 2021, pukul 18.30:30 (PST). Hasil agregasi kedua adalah pada 2 Juli 2021 pukul 06.30:30 WIB, dan seterusnya.

Mendefinisikan metrik ()AWS CLI

Saat menentukan metrik untuk model aset dengan AWS IoT SiteWise API, Anda menentukan parameter berikut:

- name- Nama properti.
- dataType— Tipe data metrik, yang bisa DOUBLE atauSTRING.

- `externalId`— (Opsional) Ini adalah ID yang ditentukan pengguna. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.
- `expression`— Ekspresi metrik. Ekspresi metrik dapat menggunakan [fungsi agregasi](#) untuk memasukkan data dari properti untuk semua aset terkait dalam hierarki. Untuk informasi selengkapnya, lihat [Menggunakan ekspresi rumus](#).
- `window`— Interval waktu dan offset untuk jendela jatuh metrik, di mana setiap interval dimulai ketika yang sebelumnya berakhir:
 - `interval`— Interval waktu untuk jendela jatuh. Interval waktu harus antara satu menit dan satu minggu.
 - `offsets`— Offset untuk jendela yang jatuh.

Untuk informasi selengkapnya, lihat [TumblingWindow](#) di Referensi AWS IoT SiteWise API.

Example interval waktu kustom dengan offset ()AWS CLI

Contoh berikut menunjukkan kepada Anda cara menentukan interval waktu 12 jam dengan offset pada 20 Februari 2021, pukul 06:30:30 (PST).

```
{
  "window": {
    "tumbling": {
      "interval": "12h",
      "offset": " 2021-07-23T18:30:30-08"
    }
  }
}
```

Jika Anda membuat metrik pada 1 Juli 2021, sebelum atau pukul 18:30:30 (PST), Anda mendapatkan hasil agregasi pertama pada 1 Juli 2021, pukul 18.30:30 (PST). Hasil agregasi kedua adalah pada 2 Juli 2021 pukul 06.30:30 WIB, dan seterusnya.

- `variables`— Daftar variabel yang mendefinisikan properti lain dari aset atau aset anak Anda untuk digunakan dalam ekspresi. Setiap struktur variabel berisi nama sederhana untuk digunakan dalam ekspresi dan `value` struktur yang mengidentifikasi properti mana yang akan ditautkan ke variabel tersebut. `value` Struktur berisi informasi berikut:

- `propertyId`— ID properti dari mana untuk menarik nilai. Anda dapat menggunakan nama properti alih-alih ID-nya jika properti didefinisikan dalam model saat ini (bukan didefinisikan dalam model dari hierarki).
- `hierarchyId`— (Opsional) ID hierarki untuk menanyakan aset turunan untuk properti. Anda dapat menggunakan nama definisi hierarki alih-alih ID-nya. Jika Anda menghilangkan nilai ini, AWS IoT SiteWise temukan properti dalam model saat ini.

Important

Metrik hanya dapat berupa properti yang integer, double, Boolean, atau tipe string. Boolean mengkonversi ke 0 (false) dan 1 (true).

Jika Anda menentukan variabel input metrik apa pun dalam ekspresi metrik, input tersebut harus memiliki interval waktu yang sama dengan metrik keluaran.

Ekspresi rumus hanya dapat menampilkan nilai ganda atau string. Ekspresi bersarang dapat menampilkan tipe data lain, seperti string, tetapi rumus secara keseluruhan harus mengevaluasi ke angka atau string. Anda dapat menggunakan [fungsi jp](#) untuk mengonversi string ke angka. Nilai Boolean harus 1 (true) atau 0 (false). Untuk informasi selengkapnya, lihat [Nilai tidak terdefinisi, tak terbatas, dan meluap](#).

- `unit`— (Opsional) Unit ilmiah untuk properti, seperti mm atau Celcius.

Example Contoh definisi metrik

Contoh berikut menunjukkan properti metrik yang mengumpulkan data pengukuran suhu aset untuk menghitung suhu per jam maksimum di Fahrenheit. Objek ini adalah contoh dari sebuah [AssetModelProperty](#) yang berisi [Metrik](#). Anda dapat menentukan objek ini sebagai bagian dari payload [CreateAssetModel](#) permintaan untuk membuat properti metrik. Untuk informasi selengkapnya, lihat [Membuat model aset \(AWS CLI\)](#).

```
{
  ...
  "assetModelProperties": [
    ...
    {
      "name": "Max temperature",
      "dataType": "DOUBLE",
      "type": {
```

```

    "metric": {
      "expression": "max(temp_f)",
      "variables": [
        {
          "name": "temp_f",
          "value": {
            "propertyId": "Temperature F"
          }
        }
      ],
      "window": {
        "tumbling": {
          "interval": "1h"
        }
      }
    },
    "unit": "Fahrenheit"
  }
  ...
}

```

Example Contoh definisi metrik yang memasukkan data dari aset terkait

Contoh berikut menunjukkan properti metrik yang mengumpulkan beberapa data daya rata-rata turbin angin untuk menghitung total daya rata-rata untuk ladang angin. Objek ini adalah contoh dari sebuah [AssetModelProperty](#) yang berisi [Metrik](#). Anda dapat menentukan objek ini sebagai bagian dari payload [CreateAssetModel](#) permintaan untuk membuat properti metrik.

```

{
  ...
  "assetModelProperty": [
    ...
    {
      "name": "Total Average Power",
      "dataType": "DOUBLE",
      "type": {
        "metric": {
          "expression": "avg(power)",
          "variables": [
            {
              "name": "power",

```

```
        "value": {
          "propertyId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
          "hierarchyId": "Turbine Asset Model"
        }
      },
      "window": {
        "tumbling": {
          "interval": "5m"
        }
      }
    },
    "unit": "kWh"
  },
  ...
}
```

Menggunakan ekspresi rumus

Dengan ekspresi rumus, Anda dapat menentukan fungsi matematika untuk mengubah dan menggabungkan data industri mentah Anda untuk mendapatkan wawasan tentang operasi Anda. Ekspresi rumus menggabungkan literal, operator, fungsi, dan variabel untuk memproses data. Untuk informasi selengkapnya tentang cara mendefinisikan properti aset yang menggunakan ekspresi rumus, lihat [Mengubah data \(transformasi\)](#) dan [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#). Transformasi dan metrik adalah properti rumus.

Topik

- [Menggunakan variabel dalam ekspresi rumus](#)
- [Menggunakan Literal dalam ekspresi rumus](#)
- [Menggunakan operator dalam ekspresi rumus](#)
- [Menggunakan konstanta dalam ekspresi rumus](#)
- [Menggunakan fungsi dalam ekspresi rumus](#)
- [Tutorial ekspresi rumus](#)

Menggunakan variabel dalam ekspresi rumus

Variabel mewakili properti AWS IoT SiteWise aset dalam ekspresi rumus. Gunakan variabel untuk memasukkan nilai dari properti aset lain dalam ekspresi Anda, sehingga Anda dapat memproses data dari properti konstan ([atribut](#)), aliran data mentah ([pengukuran](#)), dan properti rumus lainnya.

Variabel dapat mewakili properti aset dari model aset yang sama atau dari model aset anak terkait. Hanya rumus metrik yang dapat memasukkan variabel dari model aset anak.

Anda mengidentifikasi variabel dengan nama yang berbeda di konsol dan API.

- AWS IoT SiteWise konsol — Gunakan nama properti aset sebagai variabel dalam ekspresi Anda.
- AWS IoT SiteWise API (AWS CLI, AWS SDK) — Tentukan variabel dengan [ExpressionVariable](#) struktur, yang memerlukan nama variabel dan referensi ke properti aset. Nama variabel dapat berisi huruf kecil, angka, dan garis bawah. Kemudian, gunakan nama variabel untuk mereferensikan properti aset dalam ekspresi Anda.

Nama variabel peka huruf besar/kecil.

Untuk informasi selengkapnya, lihat [Mendefinisikan transformasi](#) dan [Mendefinisikan](#) metrik.

Menggunakan variabel untuk referensi properti

Nilai variabel mendefinisikan properti yang direferensikannya. AWS IoT SiteWise menyediakan berbagai cara untuk melakukan ini.

- Berdasarkan ID properti: Anda dapat menentukan ID unik properti (UUID) untuk mengidentifikasinya.
- Dengan nama: Jika properti berada pada model aset yang sama, Anda dapat menentukan namanya di bidang ID properti.
- Dengan jalur: Nilai variabel dapat merujuk ke properti dengan jalurnya. Untuk informasi selengkapnya, lihat [Menggunakan jalur untuk mereferensikan properti model komposit khusus](#).

Note

Variabel tidak didukung oleh AWS IoT SiteWise konsol. Mereka digunakan oleh AWS IoT SiteWise API, termasuk AWS Command Line Interface (AWS CLI) dan AWS SDK.

Variabel yang Anda terima dalam respons AWS IoT SiteWise mencakup informasi lengkap tentang nilai, termasuk ID dan jalur.

Namun, ketika Anda meneruskan variabel ke AWS IoT SiteWise (misalnya, dalam panggilan “buat” atau “perbarui”), Anda hanya perlu menentukan salah satunya. Misalnya, jika Anda menentukan jalur, Anda tidak perlu memberikan ID.

Menggunakan Literal dalam ekspresi rumus

Anda dapat menentukan angka dan string literal dalam ekspresi rumus.

- Angka

Gunakan angka dan notasi ilmiah untuk mendefinisikan bilangan bulat dan ganda. Anda dapat menggunakan [notasi E](#) untuk mengekspresikan angka dengan notasi ilmiah.

Contoh: `1,2.0,.9,-23.1,7.89e3,3.4E-5`

- String

Gunakan karakter ' (kutipan) dan " (kutipan ganda) untuk menentukan string. Jenis kutipan untuk awal dan akhir harus cocok. Untuk menghindari kutipan yang cocok dengan yang Anda gunakan untuk mendeklarasikan string, sertakan karakter kutipan itu dua kali. Ini adalah satu-satunya karakter pelarian dalam AWS IoT SiteWise string.

Contoh: `'active','inactive','{"temp": 52}','{"temp": "high"}'`

Menggunakan operator dalam ekspresi rumus

Anda dapat menggunakan operator umum berikut dalam ekspresi rumus.

Operator	Deskripsi
+	<p>Jika kedua operan adalah angka, operator ini menambahkan operan kiri dan kanan.</p> <p>Jika salah satu operan adalah string, operator ini menggabungkan operan kiri dan kanan sebagai string. Misalnya, ekspresi <code>1 + 2 + "</code></p>

Operator	Deskripsi
	<p>"is three" mengevaluasi. "3 is three" String gabungan dapat memiliki hingga 1024 karakter. Jika string melebihi 1024 karakter, maka AWS IoT SiteWise tidak menampilkan titik data untuk perhitungan itu.</p>
-	<p>Mengurangi operan kanan dari operan kiri.</p> <p>Anda hanya dapat menggunakan operator ini dengan operan numerik.</p>
/	<p>Membagi operan kiri dengan operan kanan.</p> <p>Anda hanya dapat menggunakan operator ini dengan operan numerik.</p>
*	<p>Mengalikan operan kiri dan kanan.</p> <p>Anda hanya dapat menggunakan operator ini dengan operan numerik.</p>
^	<p>Meningkatkan operan kiri ke kekuatan operan kanan (eksponensial).</p> <p>Anda hanya dapat menggunakan operator ini dengan operan numerik.</p>
%	<p>Mengembalikan sisanya dari membagi operan kiri dengan operan kanan. Hasilnya memiliki tanda yang sama dengan operan kiri. Perilaku ini berbeda dari operasi modulo.</p> <p>Anda hanya dapat menggunakan operator ini dengan operan numerik.</p>
$x < y$	<p>Mengembalikan 1 x jika kurang dari y, jika tidak 0.</p>

Operator	Deskripsi
$x > y$	Mengembalikan 1 jika x lebih besar dari y , jika tidak 0.
$x \leq y$	Mengembalikan 1 jika x kurang dari atau sama dengan y , jika tidak 0.
$x \geq y$	Mengembalikan 1 jika x lebih besar dari atau sama dengan y , jika tidak 0.
$x == y$	Mengembalikan 1 jika x sama dengan y , jika tidak 0.
$x != y$	Mengembalikan 1 jika x tidak sama dengan y , jika tidak 0.
$!x$	<p>Mengembalikan 1 jika x dievaluasi ke 0 (false), jika tidak 0.</p> <p>x dievaluasi menjadi false jika:</p> <ul style="list-style-type: none"> x adalah operan numerik dan dievaluasi ke 0. x dievaluasi ke string kosong. x dievaluasi ke array kosong. x dievaluasi untuk None.
$x \text{ and } y$	<p>Mengembalikan 0 jika x dievaluasi ke 0 (false). Jika tidak, mengembalikan hasil yang dievaluasi dari y.</p> <p>x atau y dievaluasi menjadi false jika:</p> <ul style="list-style-type: none"> x atau y operan numerik dan dievaluasi menjadi 0. x atau y dievaluasi ke string kosong. x atau y dievaluasi ke array kosong. x atau y dievaluasi untuk None.

Operator	Deskripsi
<code>x or y</code>	<p>Mengembalikan 1 jika x dievaluasi ke 1 (true). Jika tidak, mengembalikan hasil yang dievaluasi dari y</p> <p>x atau y dievaluasi menjadi false jika:</p> <ul style="list-style-type: none">• x atau y operan numerik dan dievaluasi menjadi 0• x atau y dievaluasi ke string kosong.• x atau y dievaluasi ke array kosong.• x atau y dievaluasi untuk None.
<code>not x</code>	<p>Mengembalikan 1 jika x dievaluasi ke 0 (false), jika tidak 0.</p> <p>x dievaluasi menjadi false jika:</p> <ul style="list-style-type: none">• x adalah operan numerik dan dievaluasi 0• x dievaluasi ke string kosong.• x dievaluasi ke array kosong.• x dievaluasi untuk None.
<code>[]</code> <code>s[index]</code>	<p>Mengembalikan karakter <code>index</code> pada indeks strings. Ini setara dengan sintaks indeks di Python.</p> <p>Example Contoh-contoh</p> <ul style="list-style-type: none">• "Hello!"[1] kembalio.• "Hello!"[-2] kembalio.

Operator	Deskripsi
<pre data-bbox="115 306 440 422">[] s[start:end:step]</pre>	<p data-bbox="829 226 1451 359">Mengembalikan sepotong strings. Ini setara dengan sintaks slice di Python. Operator ini memiliki argumen berikut:</p> <ul data-bbox="829 405 1507 884" style="list-style-type: none">• <code>start</code>— (Opsional) Indeks awal inklusif dari irisan. Default ke 0.• <code>end</code>— (Opsional) Indeks akhir eksklusif dari irisan. Default ke panjang string.• <code>step</code>— (Opsional) Jumlah yang akan bertambah untuk setiap langkah dalam irisan. Misalnya, Anda dapat menentukan 2 untuk mengembalikan irisan dengan setiap karakter lain, atau menentukan -1 untuk membalikkan irisan. Default ke 1. <p data-bbox="829 968 1495 1094">Anda dapat menghilangkan <code>step</code> argumen untuk menggunakan nilai defaultnya. Misalnya, <code>s[1:4:1]</code> setara dengan <code>s[1:4]</code>.</p> <p data-bbox="829 1142 1479 1318">Argumen harus bilangan bulat atau tidak ada konstanta. Jika Anda menentukan <code>none</code>, AWS IoT SiteWise menggunakan nilai default untuk argumen itu.</p> <p data-bbox="829 1367 1170 1398">Example Contoh-contoh</p> <ul data-bbox="829 1446 1422 1766" style="list-style-type: none">• <code>"Hello!"[1:4]</code> kembali <code>"ell"</code>.• <code>"Hello!"[:2]</code> kembali <code>"He"</code>.• <code>"Hello!"[3:]</code> kembali <code>"lo!"</code>.• <code>"Hello!"[:-4]</code> kembali <code>"He"</code>.• <code>"Hello!"[::2]</code> kembali <code>"Hlo"</code>.• <code>"Hello!"[::-1]</code> kembali <code>"!olleH"</code>.

Menggunakan konstanta dalam ekspresi rumus

Anda dapat menggunakan konstanta matematika umum berikut dalam ekspresi Anda. Semua konstanta tidak peka huruf besar/kecil.

Note


Jika Anda mendefinisikan variabel dengan nama yang sama dengan konstanta, variabel mengesampingkan konstanta.

Konstan	Deskripsi
pi	Angka pi (π): 3.141592653589793
e	Angka e: 2.718281828459045
true	Setara dengan angka 1. Pada tahun AWS IoT SiteWise, Boolean mengonversi ke jumlah yang setara.
false	Setara dengan angka 0. Pada tahun AWS IoT SiteWise, Boolean mengonversi ke jumlah yang setara.
none	Setara dengan tidak ada nilai. Anda dapat menggunakan konstanta ini untuk tidak menghasilkan apa pun sebagai hasil dari ekspresi bersyarat .

Menggunakan fungsi dalam ekspresi rumus



Anda dapat menggunakan fungsi berikut untuk mengoperasikan data dalam ekspresi rumus Anda.

Transformasi dan metrik mendukung fungsi yang berbeda. Tabel berikut menunjukkan jenis fungsi yang kompatibel dengan setiap jenis properti rumus.

 Note

Anda dapat menyertakan maksimal 10 fungsi dalam ekspresi rumus.

Jenis fungsi	Transformasi	Metrik
<u>Menggunakan fungsi umum dalam ekspresi rumus</u>	 Ya	 Ya
<u>Menggunakan fungsi perbandingan dalam ekspresi rumus</u>	 Ya	 Ya
<u>Menggunakan fungsi kondisional dalam ekspresi rumus</u>	 Ya	 Ya
<u>Menggunakan fungsi string dalam ekspresi rumus</u>	 Ya	 Ya
<u>Menggunakan fungsi agregasi dalam ekspresi rumus</u>	 Tidak	 Ya
<u>Menggunakan fungsi temporal dalam ekspresi rumus</u>	 Ya	 Ya

Jenis fungsi	Transformasi	Metrik
Menggunakan fungsi tanggal dan waktu dalam ekspresi rumus	 Ya	 Ya

Sintaks fungsi

Anda dapat menggunakan sintaks berikut untuk membuat fungsi:

Sintaks biasa

Dengan sintaks reguler, nama fungsi diikuti oleh tanda kurung dengan nol atau lebih argumen.

function_name(argument1, argument2, argument3, ...). Misalnya, fungsi dengan sintaks biasa mungkin terlihat seperti `log(x)` dan `contains(s, substring)`.

Sintaks panggilan fungsi seragam (UFCS)

UFCS memungkinkan Anda untuk memanggil fungsi menggunakan sintaks untuk panggilan metode dalam pemrograman berorientasi objek. Dengan UFCS, argumen pertama diikuti oleh dot (`.`), lalu nama fungsi dan argumen yang tersisa (jika ada) di dalam tanda kurung.

argument1.function_name(argument2, argument3, ...). Misalnya, fungsi dengan UFCS mungkin terlihat seperti `x.log()` dan `s.contains(substring)`

Anda juga dapat menggunakan UFCS untuk merantai fungsi selanjutnya. AWS IoT SiteWise menggunakan hasil evaluasi dari fungsi saat ini sebagai argumen pertama untuk fungsi berikutnya.

Misalnya, Anda dapat menggunakan `message.jp('$.status').lower().contains('fail')` sebagai pengganti `contains(lower(jp(message, '$.status')), 'fail')`.

Untuk informasi lebih lanjut, kunjungi situs web [D Programming Language](#).

Note

Anda dapat menggunakan UFCS untuk semua AWS IoT SiteWise fungsi.

AWS IoT SiteWise fungsi tidak peka huruf besar/kecil. Misalnya, Anda dapat menggunakan `lower(s)` dan secara `Lower(s)` bergantian.

Menggunakan fungsi umum dalam ekspresi rumus

Dalam [transformasi](#) dan [metrik](#), Anda dapat menggunakan fungsi berikut untuk menghitung fungsi matematika umum dalam transformasi dan metrik.

Fungsi	Deskripsi
<code>abs(x)</code>	Mengembalikan nilai absolut dari x .
<code>acos(x)</code>	Mengembalikan arccosine dari x .
<code>asin(x)</code>	Mengembalikan arcsine dari x .
<code>atan(x)</code>	Mengembalikan arctangent dari x .
<code>cbrt(x)</code>	Mengembalikan akar kubik x .
<code>ceil(x)</code>	Mengembalikan integer terdekat lebih besar dari x .
<code>cos(x)</code>	Mengembalikan cosinus dari x .
<code>cosh(x)</code>	Mengembalikan kosinus hiperbolik dari x .
<code>cot(x)</code>	Mengembalikan kotangen dari x .
<code>exp(x)</code>	Kembali e ke kekuatan x .
<code>expm1(x)</code>	Pengembalian $\exp(x) - 1$. Gunakan fungsi ini untuk menghitung lebih akurat $\exp(x) - 1$ untuk nilai kecil x .
<code>floor(x)</code>	Mengembalikan integer terdekat kurang dari x .
<code>log(x)</code>	Mengembalikan \log_e (basis e) dari x .
<code>log10(x)</code>	Mengembalikan \log_{10} (basis 10) dari x .

Fungsi	Deskripsi
<code>log1p(x)</code>	Pengembalian $\log(1 + x)$. Gunakan fungsi ini untuk menghitung lebih akurat $\log(1 + x)$ untuk nilai kecil x .
<code>log2(x)</code>	Mengembalikan \log_2 (basis2) dari x .
<code>pow(x, y)</code>	Kembali x ke kekuatannya. Ini setara dengan x^y .
<code>signum(x)</code>	Mengembalikan tanda x (-1 untuk input negatif, 0 untuk input nol, +1 untuk input positif).
<code>sin(x)</code>	Mengembalikan sinus dari x
<code>sinh(x)</code>	Mengembalikan sinus hiperbolik dari x
<code>sqrt(x)</code>	Mengembalikan akar kuadrat dari x .
<code>tan(x)</code>	Mengembalikan garis singgung x
<code>tanh(x)</code>	Mengembalikan tangen hiperbolik dari x

Menggunakan fungsi perbandingan dalam ekspresi rumus

Dalam [transformasi](#) dan [metrik](#), Anda dapat menggunakan fungsi perbandingan berikut untuk membandingkan dua nilai dan output 1 (true) atau 0 (false). AWS IoT SiteWise membandingkan string dengan urutan [leksikografi](#).

Fungsi	Deskripsi
<code>gt(x, y)</code>	Mengembalikan 1 jika x lebih besar dari y , jika tidak 0 ($x > y$). Fungsi ini tidak mengembalikan nilai if x dan tipe y yang tidak kompatibel, seperti angka dan string.

Fungsi	Deskripsi
<code>gte(x, y)</code>	<p>Mengembalikan 1 jika x lebih besar dari atau sama dengan y, jika tidak 0 ($x \geq y$).</p> <p>AWS IoT SiteWise menganggap argumen sama jika mereka berada dalam toleransi relatif $1E-9$. Ini berperilaku mirip dengan fungsi isclose di Python.</p> <p>Fungsi ini tidak mengembalikan nilai if x dan tipe y yang tidak kompatibel, seperti angka dan string.</p>
<code>eq(x, y)</code>	<p>Mengembalikan 1 jika sama dengan y, jika tidak 0 ($x == y$).</p> <p>AWS IoT SiteWise menganggap argumen sama jika mereka berada dalam toleransi relatif $1E-9$. Ini berperilaku mirip dengan fungsi isclose di Python.</p> <p>Fungsi ini tidak mengembalikan nilai if x dan tipe y yang tidak kompatibel, seperti angka dan string.</p>
<code>lt(x, y)</code>	<p>Mengembalikan 1 jika kurang dari y, jika tidak 0 ($x < y$).</p> <p>Fungsi ini tidak mengembalikan nilai if x dan tipe y yang tidak kompatibel, seperti angka dan string.</p>


Fungsi	Deskripsi
<code>lte(x, y)</code>	<p>Mengembalikan 1 x jika kurang dari atau sama dengany, jika tidak 0 ($x \leq y$).</p> <p>AWS IoT SiteWise menganggap argumen sama jika mereka berada dalam toleransi relatif $1E-9$. Ini berperilaku mirip dengan fungsi isclose di Python.</p> <p>Fungsi ini tidak mengembalikan nilai if x dan tipe y yang tidak kompatibel, seperti angka dan string.</p>
<code>isnan(x)</code>	<p>Mengembalikan 1 x jika sama denganNaN, jika tidak0.</p> <p>Fungsi ini tidak mengembalikan nilai jika x adalah string.</p>

Menggunakan fungsi kondisional dalam ekspresi rumus

Dalam [transformasi](#) dan [metrik](#), Anda dapat menggunakan fungsi berikut untuk memeriksa kondisi dan mengembalikan hasil yang berbeda, apakah kondisi dievaluasi menjadi benar atau salah.

Fungsi	Deskripsi
<code>if(condition, result_if_true, result_if_false)</code>	<p>Mengevaluasi <code>condition</code> dan mengembalikan <code>result_if_true</code> jika kondisi dievaluasi menjadi benar atau <code>result_if_false</code> jika kondisi dievaluasi. <code>false</code></p> <p><code>condition</code> harus berupa angka. Fungsi ini mempertimbangkan 0 dan string kosong sebagai <code>false</code> dan segala sesuatu yang lain (termasukNaN) sebagai<code>true</code>. Boolean mengkonversi ke 0 (<code>false</code>) dan 1 (<code>true</code>).</p>

Fungsi	Deskripsi
	<p>Anda dapat mengembalikan konstanta <code>none</code> dari fungsi ini untuk membuang output untuk kondisi tertentu. Ini berarti Anda dapat memfilter titik data yang tidak memenuhi suatu kondisi. Untuk informasi selengkapnya, lihat Memfilter titik data.</p> <p>Example Contoh-contoh</p> <ul style="list-style-type: none">• <code>if(0, x, y)</code> mengembalikan variabel <code>y</code>.• <code>if(5, x, y)</code> mengembalikan variabel <code>x</code>.• <code>if(gt(temp, 300), x, y)</code> mengembalikan variabel <code>x</code> jika variabel <code>temp</code> lebih besar dari <code>300</code>.• <code>if(gt(temp, 300), temp, none)</code> mengembalikan variabel <code>temp</code> jika lebih besar dari atau sama dengan <code>300</code>, atau <code>none</code> (tidak ada nilai) jika <code>temp</code> kurang dari <code>300</code>. <p>Kami menyarankan Anda menggunakan UFCS untuk fungsi bersyarat bersarang di mana satu atau lebih argumen adalah fungsi bersyarat. Anda dapat menggunakan <code>if(condition, result_if_true)</code> untuk mengevaluasi suatu kondisi dan <code>elif(condition, result_if_true, result_if_false)</code> untuk mengevaluasi kondisi tambahan.</p> <p>Misalnya, Anda dapat menggunakan <code>if(condition1, result1_if_true).elif(condition2, result2_if_true, result2_if_false)</code> sebagai pengganti <code>if(condition1, result1_if_true,</code></p>

Fungsi	Deskripsi
	<p><code>if(condition2, result2_if_true, result2_if_false))</code> .</p> <p>Anda juga dapat menghubungkan fungsi kondisional menengah tambahan. Misalnya, Anda dapat menggunakan <code>if(condition1, result1_if_true).elif(condition2, result2_if_true).elif(condition3, result3_if_true, result3_if_false)</code> alih-alih menarasikan beberapa <code>if</code> pernyataan, seperti <code>if(condition1, result1_if_true, if(condition2, result2_if_true, if(condition3, result3_if_true, result3_if_false)))</code> .</p> <div data-bbox="829 940 1511 1213" style="border: 1px solid #f08080; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Important</p> <p>Anda harus menggunakan <code>elif(condition, result_if_true, result_if_false)</code> dengan UFCS.</p> </div>

Menggunakan fungsi string dalam ekspresi rumus

Dalam [transformasi](#) dan [metrik](#), Anda dapat menggunakan fungsi berikut untuk beroperasi pada string. Untuk informasi selengkapnya, lihat [Menggunakan string dalam rumus](#).

Important

Ekspresi rumus hanya dapat menampilkan nilai ganda atau string. Ekspresi bersarang dapat menampilkan tipe data lain, seperti string, tetapi rumus secara keseluruhan harus mengevaluasi ke angka atau string. Anda dapat menggunakan [fungsi jp](#) untuk mengonversi string ke angka. Nilai Boolean harus 1 (true) atau 0 (false). Untuk informasi selengkapnya, lihat [Nilai tidak terdefinisi, tak terbatas, dan meluap](#).

Fungsi	Deskripsi
<code>len(s)</code>	Mengembalikan panjang strings.
<code>find(s, substring)</code>	Mengembalikan indeks string <code>substring</code> dalam strings.
<code>contains(s, substring)</code>	Mengembalikan 1 jika string <code>s</code> berisi <code>substring</code> , jika tidak 0.
<code>upper(s)</code>	Mengembalikan string <code>s</code> dalam bentuk huruf besar.
<code>lower(s)</code>	Mengembalikan string <code>s</code> dalam bentuk huruf kecil.
<code>jp(s, json_path)</code>	<p>Mengevaluasi string <code>s</code> dengan JsonPath ekspresi <code>json_path</code> dan mengembalikan hasilnya.</p> <p>Gunakan fungsi ini untuk melakukan hal berikut:</p> <ul style="list-style-type: none">• Ekstrak nilai, array, atau objek dari struktur JSON serial.• Mengkonversi string ke angka. Misalnya, rumus <code>jp('111', '\$')</code> kembali 111 sebagai angka. <p>Untuk mengekstrak nilai string dari struktur JSON dan mengembalikannya sebagai angka, Anda harus menggunakan beberapa fungsi bersarang <code>jp</code>. <code>jp</code> Fungsi luar mengekstrak string dari struktur JSON, dan <code>jp</code> fungsi dalam mengubah string menjadi angka.</p>

Fungsi	Deskripsi
	<p>String <code>json_path</code> harus berisi string literal. Ini berarti bahwa tidak <code>json_path</code> bisa menjadi ekspresi yang mengevaluasi ke string.</p> <p>Example Contoh-contoh</p> <ul style="list-style-type: none"> • <code>jp('{"status":"active","value":15}', '\$.value')</code> kembali15. • <code>jp('{"measurement":{"reading":25,"confidence":0.95}}', '\$.measurement.reading')</code> kembali25. • <code>jp('[2,8,23]', '\$[2]')</code> kembali23. • <code>jp('{"values":[3,6,7]}', '\$.values[1]')</code> kembali6. • <code>jp('111', '\$')</code>kembali111. • <code>jp(jp('{"measurement":{"reading":25,"confidence":"0.95"}}', '\$.measurement.confidence'), '\$')</code> kembali0.95.
<pre>join(s0, s1, s2, s3, ...)</pre>	<p>Mengembalikan string gabungan dengan pembatas. Fungsi ini menggunakan string input pertama sebagai pembatas dan menggabungkan string input yang tersisa bersama-sama. Ini berperilaku mirip dengan fungsi join (CharSequence delimiter, CharSequence... elements) di Java.</p> <p>Example Contoh-contoh</p> <ul style="list-style-type: none"> • <code>join("-", "aa", "bb", "cc")</code>kembali aa-bb-cc

Fungsi	Deskripsi
<code>format(expression: "format")</code> atau <code>format("format", expression)</code>	<p>Mengembalikan string dalam format yang ditentukan. Fungsi ini mengevaluasi <code>expression</code> nilai, dan kemudian mengembalikan nilai dalam format yang ditentukan. Ini berperilaku mirip dengan fungsi format (String format, Object... args) di Java. Untuk informasi selengkapnya tentang format yang didukung, lihat Konversi di bawah Formatter Kelas di Platform Java, Spesifikasi API Edisi Standar 7.</p> <p>Example Contoh-contoh</p> <ul style="list-style-type: none">• <code>format(100+1: "d")</code> mengembalikan string,101.• <code>format("The result is %d", 100+1)</code> mengembalikan string,The result is 101.

Fungsi	Deskripsi
f'expression'	<p>Mengembalikan string gabungan. Dengan fungsi diformat ini, Anda dapat menggunakan an ekspresi sederhana untuk menggabungkan dan memformat string. Fungsi-fungsi ini mungkin berisi ekspresi bersarang. Anda dapat menggunakan {} (kurawal kurawal) untuk menginterpolasi ekspresi. Ini berperilaku mirip dengan literal string yang diformat di Python.</p> <p>Example Contoh-contoh</p> <ul style="list-style-type: none"> • f'abc{1+2: "f"}d' kembali abc3.000000d . Untuk mengevaluasi ekspresi contoh ini, lakukan hal berikut: <ol style="list-style-type: none"> 1. format(1+2: "f") mengembalikan nomor floating point, 3.000000. 2. join(' ', "abc", 1+2, 'd') mengembalikan string, abc3.000000d . <p>Anda juga dapat menulis ekspresi dengan cara berikut: join(' ', "abc", format(1+2: "f"), 'd') .</p>

Menggunakan fungsi agregasi dalam ekspresi rumus

Hanya dalam [metrik](#), Anda dapat menggunakan fungsi berikut yang menggabungkan nilai input setiap interval waktu dan menghitung nilai output tunggal. Fungsi agregasi dapat mengumpulkan data dari aset terkait.

Argumen fungsi agregasi dapat berupa [variabel](#), [literal angka](#), [fungsi temporal](#), ekspresi bersarang, atau fungsi agregasi. Rumus `max(latest(x), latest(y), latest(z))` menggunakan fungsi agregasi sebagai argumen dan mengembalikan nilai saat ini terbesar dari `x`, `y`, dan `z` properti.

Anda dapat menggunakan ekspresi bersarang dalam fungsi agregasi. Bila Anda menggunakan ekspresi bersarang, aturan berikut berlaku:

- Setiap argumen hanya dapat memiliki satu variabel.

Example

Misalnya, $\text{avg}(x*(x-1))$ dan $\text{sum}(x/2)/\text{avg}(y^2)$ didukung.

Misalnya, $\text{min}(x/y)$ tidak didukung.

- Setiap argumen dapat memiliki ekspresi bersarang bertingkat.

Example

Misalnya, $\text{sum}(\text{avg}(x^2)/2)$ didukung.

- Argumen yang berbeda dapat memiliki variabel yang berbeda.

Example

Misalnya, $\text{sum}(x/2, y*2)$ didukung.

Note

- Jika ekspresi Anda berisi pengukuran, AWS IoT SiteWise gunakan nilai terakhir selama interval waktu saat ini untuk pengukuran untuk menghitung agregat.
- Jika ekspresi Anda berisi atribut, AWS IoT SiteWise gunakan nilai terbaru untuk atribut untuk menghitung agregat.

Fungsi	Deskripsi
$\text{avg}(x_0, \dots, x_n)$	Mengembalikan rata-rata nilai variabel yang diberikan selama interval waktu saat ini. Fungsi ini menghasilkan titik data hanya jika variabel yang diberikan memiliki setidaknya satu titik data selama interval waktu saat ini.
$\text{sum}(x_0, \dots, x_n)$	Mengembalikan jumlah nilai variabel yang diberikan selama interval waktu saat ini.

Fungsi	Deskripsi
$\min(x_0, \dots, x_n)$	<p>Fungsi ini menghasilkan titik data hanya jika variabel yang diberikan memiliki setidaknya satu titik data selama interval waktu saat ini.</p> <p>Mengembalikan nilai minimum variabel yang diberikan selama interval waktu saat ini.</p> <p>Fungsi ini menghasilkan titik data hanya jika variabel yang diberikan memiliki setidaknya satu titik data selama interval waktu saat ini.</p>
$\max(x_0, \dots, x_n)$	<p>Mengembalikan nilai maksimum variabel yang diberikan selama interval waktu saat ini.</p> <p>Fungsi ini menghasilkan titik data hanya jika variabel yang diberikan memiliki setidaknya satu titik data selama interval waktu saat ini.</p>
$\text{count}(x_0, \dots, x_n)$	<p>Mengembalikan jumlah total titik data untuk variabel yang diberikan selama interval waktu saat ini. Untuk informasi selengkapnya tentang cara menghitung jumlah titik data yang memenuhi suatu kondisi, lihat Menghitung titik data yang cocok dengan suatu kondisi.</p> <p>Fungsi ini menghitung titik data untuk setiap interval waktu.</p>
$\text{stdev}(x_0, \dots, x_n)$	<p>Mengembalikan standar deviasi nilai variabel yang diberikan selama interval waktu saat ini.</p> <p>Fungsi ini menghasilkan titik data hanya jika variabel yang diberikan memiliki setidaknya satu titik data selama interval waktu saat ini.</p>

Menggunakan fungsi temporal dalam ekspresi rumus

Gunakan fungsi temporal untuk mengembalikan nilai berdasarkan stempel waktu titik data.

Menggunakan fungsi temporal dalam metrik

Hanya dalam [metrik](#), Anda dapat menggunakan fungsi berikut yang mengembalikan nilai berdasarkan stempel waktu titik data.

Argumen fungsi temporal harus berupa properti dari model aset lokal atau ekspresi bersarang. Ini berarti Anda tidak dapat menggunakan properti dari model aset anak dalam fungsi temporal.

Anda dapat menggunakan ekspresi bersarang dalam fungsi temporal. Bila Anda menggunakan ekspresi bersarang, aturan berikut berlaku:

- Setiap argumen hanya dapat memiliki satu variabel.

Misalnya, `latest(t*9/5 + 32)` didukung.

- Argumen tidak bisa berupa fungsi agregasi.

Misalnya, `first(sum(x))` tidak didukung.

Fungsi	Deskripsi
<code>first(x)</code>	Mengembalikan nilai variabel yang diberikan dengan stempel waktu paling awal selama interval waktu saat ini.
<code>last(x)</code>	Mengembalikan nilai variabel yang diberikan dengan timestamp terbaru selama interval waktu saat ini.
<code>earliest(x)</code>	Mengembalikan nilai terakhir variabel yang diberikan sebelum dimulainya interval waktu saat ini. Fungsi ini menghitung titik data untuk setiap interval waktu, jika properti input memiliki setidaknya satu titik data dalam sejarahnya. Lihat time-range-defintion untuk detail.

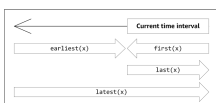
Fungsi	Deskripsi
<code>latest(x)</code>	<p>Mengembalikan nilai terakhir variabel yang diberikan dengan stempel waktu terbaru sebelum akhir interval waktu saat ini.</p> <p>Fungsi ini menghitung titik data untuk setiap interval waktu, jika properti input memiliki setidaknya satu titik data dalam sejarahnya. Lihat time-range-defintion untuk detail.</p>
<code>statetime(x)</code>	<p>Mengembalikan jumlah waktu dalam detik bahwa variabel yang diberikan positif selama interval waktu saat ini. Anda dapat menggunakan an fungsi perbandingan untuk membuat properti transform untuk <code>statetime</code> fungsi yang akan digunakan.</p> <p>Misalnya, jika Anda memiliki <code>Idle</code> properti yang <code>0</code> atau <code>1</code>, Anda dapat menghitung waktu idle per interval waktu dengan ekspresi ini: <code>IdleTime = statetime(Idle)</code> . Untuk informasi selengkapnya, lihat contoh skenario statetime.</p> <p>Fungsi ini tidak mendukung properti metrik sebagai variabel input.</p> <p>Fungsi ini menghitung titik data untuk setiap interval waktu, jika properti input memiliki setidaknya satu titik data dalam sejarahnya.</p>

Fungsi	Deskripsi
<code>TimeWeightedAvg(x, [interpolation])</code>	<p>Mengembalikan rata-rata data masukan yang ditimbang dengan interval waktu antar titik. Lihat Parameter fungsi berbobot waktu untuk detail komputasi dan interval.</p> <p>Argumen opsional <code>interpolation</code> harus berupa konstanta string:</p> <ul style="list-style-type: none">• <code>locf</code>— Ini adalah defaultnya. Perhitungan menggunakan algoritma komputasi Last Observed Carry Forward untuk interval antar titik data. Dalam pendekatan ini, titik data dihitung sebagai nilai terakhir yang diamati hingga cap waktu titik data input berikutnya. <p>Nilai setelah titik data yang baik diekstrapolasi sebagai nilainya hingga stempel waktu titik data berikutnya.</p> <ul style="list-style-type: none">• <code>linear</code>— Perhitungan menggunakan algoritma komputasi interpolasi linier untuk interval antar titik data. <p>Nilai antara dua titik data yang baik diekstrapolasi sebagai interpolasi linier antara nilai titik data tersebut.</p> <p>Nilai antara titik data baik dan buruk atau nilai setelah titik data baik terakhir akan diekstrapolasi sebagai titik data yang baik.</p>

Fungsi	Deskripsi
<code>TimeWeightedStDev(x, [algo])</code>	<p>Mengembalikan standar deviasi data masukan tertimbang dengan interval waktu antar titik.</p> <p>Lihat Parameter fungsi berbobot waktu untuk detail komputasi dan interval.</p> <p>Perhitungan menggunakan algoritma komputasi Last Observed Carry Forward untuk interval antar titik data. Dalam pendekatan ini, titik data dihitung sebagai nilai terakhir yang diamati hingga cap waktu titik data input berikutnya. Berat dihitung sebagai interval waktu dalam hitungan detik antara titik data atau batas jendela.</p> <p>Argumen opsional <code>algo</code> harus berupa konstanta string:</p> <ul style="list-style-type: none">• <code>f</code> Ini adalah defaultnya. Ini mengembalikan varians sampel tertimbang yang tidak bias dengan bobot Frekuensi, di mana <code>TimeWeight</code> dihitung dalam hitungan detik. Algoritma ini biasanya diasumsikan di bawah standar deviasi dan dikenal sebagai koreksi standar deviasi Bessel untuk sampel tertimbang.• <code>p</code>— Mengembalikan varians sampel berbobot bias, juga dikenal sebagai Varians populasi. <p>Rumus berikut digunakan untuk perhitungan di mana:</p> <ul style="list-style-type: none">• S_p = deviasi standar populasi• S_f = standar deviasi frekuensi• X_i = data masuk

Fungsi	Deskripsi
	<ul style="list-style-type: none"> • ω_i = berat yang sama dengan interval waktu dalam detik • μ^* = rata-rata tertimbang dari data yang masuk <p>Persamaan untuk standar deviasi populasi:</p> $S_p^2 = \frac{\sum_{i=1}^N \omega_i (x_i - \mu^*)^2}{\sum_{i=1}^N \omega_i}$ <p>Persamaan untuk standar deviasi frekuensi:</p> $S_f^2 = \frac{\sum_{i=1}^N \omega_i (x_i - \mu^*)^2}{\sum_{i=1}^N \omega_i - 1}$

Diagram berikut menunjukkan bagaimana AWS IoT SiteWise menghitung fungsi `temporalfirst`, `last`, dan `earliestlatest`, relatif terhadap interval waktu saat ini.



i Note

- Rentang waktu untuk `first(x)`, `last(x)` adalah (mulai jendela saat ini, akhir jendela saat ini].
- Rentang waktu untuk `latest(x)` adalah (awal waktu, akhir jendela saat ini].
- Rentang waktu untuk `earliest(x)` adalah (awal waktu, akhir jendela sebelumnya].

Parameter fungsi berbobot waktu

Fungsi berbobot waktu yang dihitung untuk jendela agregat memperhitungkan hal-hal berikut:

- Titik data di dalam jendela
- Interval waktu antara titik data
- Titik data terakhir sebelum jendela
- Titik data pertama setelah jendela (untuk beberapa algoritma)

Ketentuan:

- Titik data buruk — Setiap titik data dengan kualitas yang tidak baik atau nilai non-angka. Ini tidak dipertimbangkan dalam perhitungan hasil jendela.
- Interval buruk — Interval setelah titik data yang buruk. Interval sebelum titik data pertama yang diketahui juga dianggap sebagai interval yang buruk.
- Titik data yang baik — Setiap titik data dengan kualitas dan nilai numerik yang baik.

Note

- AWS IoT SiteWise hanya mengkonsumsi data GOOD berkualitas saat menghitung transformasi dan metrik. Ini mengabaikan UNCERTAIN dan titik BAD data.
- Interval sebelum titik data pertama yang diketahui dianggap sebagai interval yang buruk. Untuk informasi selengkapnya, lihat [the section called “Tutorial ekspresi rumus”](#).

Interval setelah titik data terakhir yang diketahui berlanjut tanpa batas, mempengaruhi semua jendela berikut. Ketika titik data baru tiba, fungsi menghitung ulang interval.

Mengikuti aturan di atas, hasil jendela agregat dihitung dan terbatas pada batas jendela. Secara default, fungsi hanya mengirimkan hasil jendela jika seluruh jendela adalah interval yang baik.

Jika interval jendela yang baik lebih kecil dari panjang jendela, fungsi tidak mengirim jendela.

Ketika titik data yang mempengaruhi hasil jendela berubah, fungsi menghitung ulang jendela, bahkan jika titik data berada di luar jendela.

Jika properti input memiliki setidaknya satu titik data dalam histori dan perhitungan telah dimulai, fungsi menghitung fungsi agregat berbobot waktu untuk setiap interval waktu.

Example Contoh skenario statetime

Pertimbangkan contoh di mana Anda memiliki aset dengan properti berikut:

- **Idle**— Pengukuran yang 0 atau 1. Ketika nilainya 1, mesin menganggur.
- **Idle Time**— Metrik yang menggunakan rumus `statetime(Idle)` untuk menghitung jumlah waktu dalam detik di mana mesin dalam keadaan idle, per interval 1 menit.

IdleProperti ini memiliki titik data berikut.

Stempel waktu	14:00:00	14:00:30	14:01:15	14:02:45	14:04:00
Idle	0	1	1	0	0

AWS IoT SiteWise menghitung Idle Time properti setiap menit dari nilai Idle. Setelah perhitungan ini selesai, Idle Time properti memiliki titik data berikut.

Stempel waktu	14:00:00	14:01:00PM	14:02:00PM	14:03:00	14:04:00
Idle Time	N/A	30	60	45	0

AWS IoT SiteWise melakukan perhitungan berikut untuk Idle Time pada akhir setiap menit.

- Pukul 14:00 (untuk 13:59 hingga 14:00)
 - Tidak ada data Idle sebelum pukul 14:00, jadi tidak ada titik data yang dihitung.
- Pukul 14:01 (untuk 2:00 PM hingga 14:01 PM)
 - Pada pukul 14:00:00, mesin aktif (`Idle`sedang0).
 - Pada 14:00:30, mesin dalam keadaan idle (`Idle`is). 1
 - Idle tidak berubah lagi sebelum akhir interval pada 2:01:00 PM, begitu Idle Time juga 30 detik.
- Pukul 14:02 (untuk 14:01 PM hingga 14:02 PM)
 - Pada pukul 14:01:00, mesin dalam keadaan idle (per titik data terakhir pada pukul 14:00:30).

- Pukul 14:01:15, mesin masih menganggur.
- Idle tidak berubah lagi sebelum akhir interval pada 2:02:00 PM, begitu Idle Time juga 60 detik.
- Pukul 14:03 (untuk 14:02 PM hingga 14:03 PM)
 - Pada pukul 14:02:00, mesin dalam keadaan idle (per titik data terakhir pada 2:01:15 PM).
 - Pada pukul 14:02:45, mesin aktif.
 - Idle tidak berubah lagi sebelum akhir interval pada 2:03:00 PM, begitu Idle Time juga 45 detik.
- Pukul 14:04 (untuk 14:03 PM hingga 14:04 PM)
 - Pada 2:03:00 PM, mesin aktif (per titik data terakhir pada 2:02:45 PM).
 - Idle tidak berubah lagi sebelum akhir interval pada 2:04:00 PM, begitu Idle Time juga 0 detik.

Example Contoh TimeWeightedAvg dan TimeWeightedStDev skenario

Tabel berikut menyediakan contoh input dan output untuk metrik jendela satu menit ini:

`Avg(x)`, `TimeWeightedAvg(x)`, `TimeWeightedAvg(x, "linear")`, `stDev(x)`,
`timeWeightedStDev(x)`, `timeWeightedStDev(x, 'p')`

Masukan sampel untuk jendela agregat satu menit:

Note

Semua titik data ini memiliki GOOD kualitas.

03:00:00	4.0
03:01:00	2.0
03:01:10	8.0
03:01:50	20.0
03:02:00	14.0
03:02:05	10.0

03:02:10	3.0
03:02:30	20.0
03:03:30	0.0

Output hasil agregat:

Note

Tidak ada - Hasil tidak diproduksi untuk jendela ini.

Waktu	Avg(x)	TimeWeightedAvg(x)	TimeWeightedAvg(X, "linear")	stDev(X)	timeWeightedStDev(x)	timeWeightedStDev(x, 'p')
3:00:00	4	Tidak ada	Tidak ada	0	Tidak ada	Tidak ada
3:01:00	2	4	3	0	0	0
3:02:00	14	9	13	6	5.430610041581775	5.385164807134504
3:03:00	11	13	12.875	8.54400374531753	7.724054437220943	7.659416862050705
3:04:00	0	10	2.5	0	10.084389681792215	10
3:05:00	Tidak ada	0	0	Tidak ada	0	0

Menggunakan fungsi temporal dalam transformasi

Dalam [transformasi](#) saja, Anda dapat menggunakan `pretrigger()` fungsi untuk mengambil nilai GOOD kualitas untuk variabel sebelum pembaruan properti yang memulai perhitungan transformasi saat ini.

Pertimbangkan contoh di mana produsen menggunakan AWS IoT SiteWise untuk memantau status mesin. Pabrik menggunakan pengukuran dan transformasi berikut untuk mewakili proses:

- Pengukuran, `current_state`, yang bisa 0 atau 1.
 - Jika mesin dalam keadaan pembersihan, `current_state` sama dengan 1.
 - Jika mesin dalam keadaan manufaktur, `current_state` sama dengan 0.
- Sebuah transformasi, `cleaning_state_duration`, itu sama.


```
if(pretrigger(current_state) == 1, timestamp(current_state) -
timestamp(pretrigger(current_state)), none)
```

 Transformasi ini mengembalikan berapa lama mesin berada dalam keadaan pembersihan dalam hitungan detik, dalam format epoch Unix. Untuk informasi selengkapnya, lihat [Menggunakan fungsi kondisional dalam ekspresi rumus](#) dan fungsi [timestamp \(\)](#).

Jika mesin tetap dalam kondisi pembersihan lebih lama dari yang diharapkan, pabrik mungkin menyelidiki mesin.

Anda juga dapat menggunakan `pretrigger()` fungsi dalam transformasi multivariat. Misalnya, Anda memiliki dua pengukuran bernama `x` dan `y`, dan transformasi, `z`, yang sama. $x + y + \text{pretrigger}(y)$ Tabel berikut menunjukkan nilai untuk `x`, `y`, dan `z` dari 9:00 AM sampai 9:15 AM.

Note

- Contoh ini mengasumsikan bahwa nilai untuk pengukuran tiba secara kronologis. Misalnya, nilai `x` untuk 09:00 AM tiba sebelum nilai `x` untuk 09:05 AM.
- Jika titik data untuk 9:05 tiba sebelum titik data untuk 9:00 AM, `z` tidak dihitung pada 9:05 AM.
- Jika nilai `x` untuk 9:05 tiba sebelum nilai `x` untuk 09:00 AM dan nilai `y` tiba secara kronologis, `z` sama $22 = 20 + 1 + 1$ dengan pukul 9:05 pagi.

	09:00AM	09:05AM	09:10AM	09:15AM
<code>x</code>	10	20		30
<code>y</code>	1	2	3	

	09:00AM	09:05AM	09:10AM	09:15AM
$z = x + y$ $+ \text{pretrigge}$ $r(y)$	ytidak menerima titik data apa pun sebelum pukul 09:00. Oleh karena itu, z tidak dihitung pada pukul 09:00.	$23 = 20 + 2 + 1$ pretrigge $r(y)$ sama dengan 1.	$25 = 20 + 3 + 2$ xtidak menerima titik data baru. pretrigge $r(y)$ sama dengan 2.	$36 = 30 + 3 + 3$ ytidak menerima titik data baru. Oleh karena itu, pretrigge $r(y)$ sama dengan 3 pada 09:15 AM.

Menggunakan fungsi tanggal dan waktu dalam ekspresi rumus

Dalam [transformasi](#) dan [metrik](#), Anda dapat menggunakan fungsi tanggal dan waktu dengan cara berikut:

- Ambil stempel waktu saat ini dari titik data di UTC atau di zona waktu lokal.
- Membangun stempel waktu dengan argumen, seperti, `year`, `month` dan `day_of_month`
- Ekstrak periode waktu seperti satu tahun atau bulan dengan `unix_time` argumen.

Fungsi	Deskripsi
<code>now()</code>	Mengembalikan tanggal dan waktu saat ini, dalam detik, dalam format epoch Unix.
<code>timestamp()</code>	<ul style="list-style-type: none"> • Dalam transformasi, fungsi mengembalikan stempel waktu, dalam hitungan detik, dari pesan input dalam format epoch Unix. <p>Hanya dalam transformasi, Anda dapat melakukan salah satu dari yang berikut:</p> <ul style="list-style-type: none"> • Berikan variabel sebagai argumen untuk fungsi. <code>timestamp(<i>variable-name</i>)</code> Fungsi mengembalikan stempel waktu, dalam hitungan detik, dari nilai

Fungsi	Deskripsi
	<p>GOOD kualitas terbaru untuk variabel yang ditentukan dalam format epoch Unix.</p> <p>Misalnya, jika aset Anda memiliki properti transform bernama <code>Temperature_F</code> yang menggunakan $9/5 * \text{Temperature_C}$ rumus untuk mengonversi setiap titik data suhu dari Celcius ke Fahrenheit, Anda dapat menggunakan <code>timestamp (Temperature_F)</code> fungsi tersebut untuk mendapatkan stempel waktu nilai kualitas terbaru untuk properti tersebut.</p> <p><code>GOOD Temperature_F</code></p> <ul style="list-style-type: none"> Gunakan <code>pretrigger()</code> fungsi sebagai argumen untuk fungsi. <code>timestamp (pretrigger(<i>variable-name</i>))</code> Fungsi mengembalikan stempel waktu, dalam hitungan detik, dari nilai GOOD kualitas untuk variabel tertentu sebelum pembaruan properti yang memulai perhitungan transformasi saat ini dalam format epoch Unix. Untuk informasi selengkapnya, lihat Menggunakan fungsi temporal dalam transformasi. Dalam metrik, fungsi mengembalikan stempel waktu yang diambil di akhir jendela saat ini, dalam hitungan detik, dalam format epoch Unix.

Fungsi	Deskripsi
<pre>mktime(time_zone, year, month, day_of_month, hour, minute, second)</pre>	<p>Mengembalikan waktu input dalam hitungan detik, dalam format epoch Unix.</p> <p>Persyaratan berikut berlaku untuk menggunakan fungsi ini:</p> <ul style="list-style-type: none">• Argumen zona waktu harus berupa string yang dikutip (' UTC '). Jika tidak ditentukan, zona waktu default adalah UTC. <p>Argumen zona waktu dapat menjadi argumen pertama atau terakhir.</p> <ul style="list-style-type: none">• Argumen tahun, bulan, hari bulan, jam, menit, dan kedua harus berurutan.• Argumen tahun, bulan, dan tanggal diperlukan. <p>Batasan berikut berlaku untuk menggunakan fungsi ini:</p> <ul style="list-style-type: none">• <code>year</code>- Nilai yang valid adalah antara 1970 dan 2250.• <code>month</code>- Nilai yang valid adalah antara 1 dan 12.• <code>day-of-month</code> - Nilai yang valid adalah antara 1 - 31.• <code>hour</code>- Nilai yang valid adalah antara 0 dan 23.• <code>minute</code>- Nilai yang valid adalah antara 0 dan 59.• <code>second</code>- Nilai yang valid adalah antara 0 dan 60. Ini bisa menjadi nomor floating point. <p>Contoh:</p>

Fungsi	Deskripsi
	<ul style="list-style-type: none">• <code>mktime(2020, 2, 29)</code>• <code>mktime('UTC+3', 2021, 12, 31, 22)</code>• <code>mktime(2022, 10, 13, 2, 55, 13.68, 'PST')</code>

Fungsi	Deskripsi
<code>localtime(unix_time, time_zone)</code>	<p>Mengembalikan tahun, hari dalam sebulan, hari dalam seminggu, hari dalam setahun, jam, menit, atau yang kedua di zona waktu yang ditentukan dari waktu Unix.</p> <p>Persyaratan berikut berlaku untuk menggunakan fungsi ini:</p> <ul style="list-style-type: none">• Argumen zona waktu harus berupa string yang diketip ('UTC'). Jika tidak ditentukan, zona waktu default adalah UTC.• Argumen waktu Unix adalah waktu dalam detik, dalam format epoch Unix. Kisaran valid adalah antara 1-31556889864403199. Ini bisa menjadi nomor floating point. <p>Contoh respon: <code>2007-12-03T10:15:30+01:00[Europe/Paris]</code></p> <p><code>localtime(unix_time, time_zone)</code> bukan fungsi mandiri. <code>sec()</code> Fungsi <code>year()</code> <code>mon()</code>, <code>mday</code>, <code>wday()</code>, <code>yday()</code>, <code>hour()</code> <code>minute()</code>, dan diambil <code>localtime(unix_time, time_zone)</code> sebagai argumen.</p> <p>Contoh:</p> <ul style="list-style-type: none">• <code>year(localtime('GMT', 1605898608.8113723))</code>• <code>now().localtime().year()</code>• <code>timestamp().localtime('PST').year()</code>

Fungsi	Deskripsi
	<ul style="list-style-type: none"> <code>localtime(1605289736, 'Europe/London').year()</code>
<code>year(localtime(unix_time, time_zone))</code>	Mengembalikan tahun dari <code>localtime(unix_time, time_zone)</code> .
<code>mon(localtime(unix_time, time_zone))</code>	Mengembalikan bulan dari <code>localtime(unix_time, time_zone)</code> .
<code>mday(localtime(unix_time, time_zone))</code>	Mengembalikan hari dalam sebulan dari <code>localtime(unix_time, time_zone)</code> .
<code>wday(localtime(unix_time, time_zone))</code>	Mengembalikan hari dalam seminggu dari <code>localtime(unix_time, time_zone)</code> .
<code>yday(localtime(unix_time, time_zone))</code>	Mengembalikan hari dalam setahun dari <code>localtime(unix_time, time_zone)</code> .
<code>hour(localtime(unix_time, time_zone))</code>	Mengembalikan jam dari <code>localtime(unix_time, time_zone)</code> .
<code>minute(localtime(unix_time, time_zone))</code>	Mengembalikan menit dari <code>localtime(unix_time, time_zone)</code> .
<code>sec(localtime(unix_time, time_zone))</code>	Mengembalikan yang kedua dari <code>localtime(unix_time, time_zone)</code> .

Format zona waktu yang didukung

Anda dapat menentukan argumen zona waktu dengan cara berikut:

- Offset zona waktu - Tentukan 'Z' untuk UTC atau offset ('+2' atau) '-5'
- ID Offset - Gabungkan singkatan zona waktu dan offset. Misalnya, 'GMT+2' dan 'UTC-01:00'. Singkatan zona waktu harus berisi hanya tiga huruf.

- ID berbasis wilayah - Misalnya, 'Etc/GMT+12' dan 'Pacific/Pago_Pago'.

Singkatan zona waktu yang didukung

Fungsi tanggal dan waktu mendukung singkatan zona waktu tiga huruf berikut:

- EST - - 05:00
- HST - - 10:00
- MST - - 07:00
- ACT - Australia/Darwin
- AET - Australia/Sydney
- AGT - Amerika/Argentina/Buenos_Aires
- ART - Afrika/Kairo
- AST - Amerika/Anchorage
- TARUHAN - Amerika/Sao_Paulo
- BST - Asia/Dhaka
- CAT - Afrika/Harare
- CET - Eropa/Paris
- CNT - Amerika/St_Johns
- CST - Amerika/Chicago
- CTT - Asia/Shanghai
- MAKAN - Afrika/Addis_Ababa
- IET - Amerika/Indiana/Indianapolis
- IST - Asia/Kolkata
- JST - Asia/Tokyo
- MIT - Pasifik/Apia
- NET - Asia/Yerevan
- NST - Pasifik/Auckland
- PLT - Asia/Karachi
- PRT - Amerika/Puerto_Rico

- PST - Amerika/Los_Angeles
- SST - Pasifik/Guadalkanal
- VST - Asia/Ho_Chi_Minh

ID berbasis wilayah yang didukung

Fungsi tanggal dan waktu mendukung ID berbasis Wilayah berikut, yang diatur berdasarkan hubungannya dengan UTC+ 00:00:

- DII/GMT+12 (UTC- 12:00)
- Pasifik/Pago_Pago (UTC- 11:00)
- Pasifik/Samoa (UTC- 11:00)
- Pasifik/Niue (UTC- 11:00)
- AS/Samoa (UTC- 11:00)
- DII/GMT+11 (UTC- 11:00)
- Pasifik/Midway (UTC- 11:00)
- Pasifik/Honolulu (UTC- 10:00)
- Pasifik/Rarotonga (UTC- 10:00)
- Pasifik/Tahiti (UTC- 10:00)
- Pasifik/Johnston (UTC- 10:00)
- AS/Hawaii (UTC- 10:00)
- SistemV/HST10 (UTC- 10:00)
- DII/GMT+10 (UTC- 10:00)
- Pasifik/Marquesas (UTC- 09:30)
- DII/GMT+9 (UTC- 09:00)
- Pasifik/Gambier (UTC- 09:00)
- Amerika/Atka (UTC- 09:00)
- SystemV/YST9 (UTC- 09:00)
- Amerika/Adak (UTC- 09:00)
- AS/Aleutian (UTC- 09:00)

- DII/GMT+8 (UTC- 08:00)
- AS/Alaska (UTC- 08:00)
- Amerika/Juni (UTC- 08:00)
- Amerika/Metlakatla (UTC- 08:00)
- Amerika/Yakutat (UTC- 08:00)
- Pasifik/Pitcairn (UTC- 08:00)
- Amerika/Sitka (UTC- 08:00)
- Amerika/Anchorage (UTC- 08:00)
- SistemV/PST8 (UTC- 08:00)
- Amerika>Nama (UTC- 08:00)
- SystemV/YST9YDT (UTC- 08:00)
- Kanada/Yukon (UTC- 07:00)
- AS/Pasifik-Baru (UTC- 07:00)
- DII/GMT+7 (UTC- 07:00)
- Amerika Serikat/Arizona (UTC- 07:00)
- Amerika/Dawson_Creek (UTC- 07:00)
- Kanada/Pasifik (UTC- 07:00)
- PST8PDT (UTC- 07:00)
- SistemV/MST7 (UTC- 07:00)
- Amerika/Dawson (UTC- 07:00)
- Meksiko/ BajaNorte (UTC- 07:00)
- Amerika/Tijuana (UTC- 07:00)
- Amerika/Creston (UTC- 07:00)
- Amerika/Hermosillo (UTC- 07:00)
- Amerika/Santa_Isabel (UTC- 07:00)
- Amerika/Vancouver (UTC- 07:00)
- Amerika/Ensenada (UTC- 07:00)
- Amerika/Phoenix (UTC- 07:00)

- Amerika/Whitehorse (UTC- 07:00)
- Amerika/Fort_Nelson (UTC- 07:00)
- SistemV/PST8PDT (UTC- 07:00)
- Amerika/Los_Angeles (UTC- 07:00)
- AS/Pasifik (UTC- 07:00)
- Amerika/El_Salvador (UTC- 06:00)
- Amerika/Guatemala (UTC- 06:00)
- Amerika/Belize (UTC- 06:00)
- Amerika/Managua (UTC- 06:00)
- Amerika/Tegucigalpa (UTC- 06:00)
- DII/GMT+6 (UTC- 06:00)
- Pasifik/Paskah (UTC- 06:00)
- Meksiko/ BajaSur (UTC- 06:00)
- Amerika/Regina (UTC- 06:00)
- Amerika/Denver (UTC- 06:00)
- Pasifik/Galapagos (UTC- 06:00)
- Amerika/Yellowknife (UTC- 06:00)
- Amerika/Swift_Current (UTC- 06:00)
- Amerika/Inuvik (UTC- 06:00)
- Amerika/Mazatlan (UTC- 06:00)
- Amerika/Boise (UTC- 06:00)
- Amerika/Costa_Rica (UTC- 06:00)
- MST7MDT (UTC-06:00)
- SistemV/CST6 (UTC- 06:00)
- Amerika/Chihuahua (UTC- 06:00)
- Amerika/Ojinaga (UTC- 06:00)
- Chili/ EasterIsland (UTC- 06:00)
- AS/Gunung (UTC- 06:00)
- Amerika/Edmonton (UTC- 06:00)

- Kanada/Gunung (UTC- 06:00)
- Amerika/Cambridge_Bay (UTC- 06:00)
- Navajo (UTC- 06:00)
- SistemV/MST7MDT (UTC- 06:00)
- Kanada/Saskatchewan (UTC- 06:00)
- Amerika/Shiprock (UTC- 06:00)
- Amerika/Panama (UTC- 05:00)
- Amerika/Chicago (UTC- 05:00)
- Amerika/Eirunepe (UTC- 05:00)
- DII/GMT+5 (UTC- 05:00)
- Meksiko/Umum (UTC- 05:00)
- Amerika/Porto_Acre (UTC- 05:00)
- Amerika/Guayaquil (UTC- 05:00)
- Amerika/Rankin_Inlet (UTC- 05:00)
- AS/Tengah (UTC- 05:00)
- Amerika/Rainy_River (UTC- 05:00)
- Amerika/Indiana/Knox (UTC- 05:00)
- Amerika/North_Dakota/Beulah (UTC- 05:00)
- Amerika/Monterrey (UTC- 05:00)
- Amerika/Jamaika (UTC- 05:00)
- Amerika/Atikokan (UTC- 05:00)
- Amerika/Coral_Harbour (UTC- 05:00)
- Amerika/North_Dakota/Pusat (UTC- 05:00)
- Amerika/Cayman (UTC- 05:00)
- Amerika/Indiana/Tell_City (UTC- 05:00)
- Amerika/Meksiko_Kota (UTC- 05:00)
- Amerika/Matamoros (UTC- 05:00)
- CST6CDT (UTC- 05:00)
- Amerika/Knox_in (UTC- 05:00)

- Amerika/Bogota (UTC- 05:00)
- Amerika/Menominey (UTC- 05:00)
- Amerika/Tegas (UTC- 05:00)
- SistemV/EST5 (UTC- 05:00)
- Kanada/Pusat (UTC- 05:00)
- Brasil/Acre (UTC- 05:00)
- Amerika/Cancun (UTC- 05:00)
- Amerika/Lima (UTC- 05:00)
- Amerika/Bahia_Banderas (UTC- 05:00)
- Amerika Serikat/Indiana (UTC- 05:00)
- Amerika/Rio_Branco (UTC- 05:00)
- SistemV/CST6CDT (UTC- 05:00)
- Jamaika (UTC- 05:00)
- Amerika/Merida (UTC- 05:00)
- Amerika/North_Dakota/New_Salem (UTC- 05:00)
- Amerika/Winnipeg (UTC- 05:00)
- Amerika/Cuiaba (UTC- 04:00)
- Amerika/Marigot (UTC- 04:00)
- Amerika/Indiana/Petersburg (UTC- 04:00)
- Chili/Kontinental (UTC- 04:00)
- Amerika/Grand_Turk (UTC- 04:00)
- Kuba (UTC- 04:00)
- DII/GMT+4 (UTC- 04:00)
- Amerika/Manaus (UTC- 04:00)
- Amerika/Fort_Wayne (UTC- 04:00)
- Amerika/St_Thomas (UTC- 04:00)
- Amerika/Anguilla (UTC- 04:00)
- Amerika/Havana (UTC- 04:00)
- AS/Michigan (UTC- 04:00)

- Amerika/Barbados (UTC- 04:00)
- Amerika/Louisville (UTC- 04:00)
- Amerika/Curacao (UTC- 04:00)
- Amerika/Guyana (UTC- 04:00)
- Amerika/Martinik (UTC- 04:00)
- Amerika/Puerto_Rico (UTC- 04:00)
- Amerika/PORT_OF_Spanyol (UTC- 04:00)
- SistemV/AST4 (UTC- 04:00)
- Amerika/Indiana/Vevay (UTC- 04:00)
- Amerika/Indiana/Vincennes (UTC- 04:00)
- Amerika/Kralendijk (UTC- 04:00)
- Amerika/Antigua (UTC- 04:00)
- Amerika/Indianapolis (UTC- 04:00)
- Amerika/Iqaluit (UTC- 04:00)
- Amerika/St_Vincent (UTC- 04:00)
- Amerika/Kentucky/Louisville (UTC- 04:00)
- Amerika/Dominika (UTC- 04:00)
- Amerika/Asuncion (UTC- 04:00)
- EST5EDT (UTC- 04:00)
- Amerika/Nassau (UTC- 04:00)
- Amerika/Kentucky/Monticello (UTC- 04:00)
- Brasil/Barat (UTC- 04:00)
- Amerika/Aruba (UTC- 04:00)
- Amerika/Indiana/Indianapolis (UTC- 04:00)
- Amerika/Santiago (UTC- 04:00)
- Amerika/La_Paz (UTC- 04:00)
- Amerika/Thunder_Bay (UTC- 04:00)
- Amerika/Indiana/Marengo (UTC- 04:00)
- Amerika/Blanc-Sablon (UTC- 04:00)

- Amerika/Santo_Domingo (UTC- 04:00)
- AS/Timur (UTC- 04:00)
- Kanada/Timur (UTC- 04:00)
- Amerika/Port-au-Prince (UTC- 04:00)
- Amerika/St_Barthelemy (UTC- 04:00)
- Amerika/Nipigon (UTC- 04:00)
- Amerika Serikat/Indiana Timur (UTC- 04:00)
- Amerika/St_Lucia (UTC- 04:00)
- Amerika/Montserrat (UTC- 04:00)
- Amerika/Lower_Princes (UTC- 04:00)
- Amerika/Detroit (UTC- 04:00)
- Amerika/Tortola (UTC- 04:00)
- Amerika/Porto_Velho (UTC- 04:00)
- Amerika/Campo_Grande (UTC- 04:00)
- Amerika/Perawan (UTC- 04:00)
- Amerika/Pangnirtung (UTC- 04:00)
- Amerika/Montreal (UTC- 04:00)
- Amerika/Indiana/Winamac (UTC- 04:00)
- Amerika/Boa_Vista (UTC- 04:00)
- Amerika/Grenada (UTC- 04:00)
- Amerika/New_York (UTC- 04:00)
- Amerika/St_Kitts (UTC- 04:00)
- Amerika/Caracas (UTC- 04:00)
- Amerika/Guadeloupe (UTC- 04:00)
- Amerika/Toronto (UTC- 04:00)
- SystemV/EST5EDT (UTC- 04:00)
- Amerika/Argentina/Catamarca (UTC- 03:00)
- Kanada/Atlantik (UTC- 03:00)
- Amerika/Argentina/Cordoba (UTC- 03:00)

- Amerika/Araguaina (UTC- 03:00)
- Amerika/Argentina/Salta (UTC- 03:00)
- DII/GMT+3 (UTC- 03:00)
- Amerika/Montevideo (UTC- 03:00)
- Brasil/Timur (UTC- 03:00)
- Amerika/Argentina/Mendoza (UTC- 03:00)
- Amerika/Argentina/Rio_Gallegos (UTC- 03:00)
- Amerika/Catamarca (UTC- 03:00)
- Amerika/Cordoba (UTC- 03:00)
- Amerika/Sao_Paulo (UTC- 03:00)
- Amerika/Argentina/Jujuy (UTC- 03:00)
- Amerika/Cayenne (UTC- 03:00)
- Amerika/Recife (UTC- 03:00)
- Amerika/Buenos_Aires (UTC- 03:00)
- Amerika/Paramaribo (UTC- 03:00)
- Amerika/Moncton (UTC- 03:00)
- Amerika/Mendoza (UTC- 03:00)
- Amerika/Santarem (UTC- 03:00)
- Atlantik/Bermuda (UTC- 03:00)
- Amerika/Maceio (UTC- 03:00)
- Atlantik/Stanley (UTC- 03:00)
- Amerika/Halifax (UTC- 03:00)
- Antartika/Rothera (UTC- 03:00)
- Amerika/Argentina/San_Luis (UTC- 03:00)
- Amerika/Argentina/Ushuaia (UTC- 03:00)
- Antartika/Palmer (UTC- 03:00)
- Amerika/Punta_Arenas (UTC- 03:00)
- Amerika/Glace_Bay (UTC- 03:00)
- Amerika/Fortaleza (UTC- 03:00)

- Amerika/Thule (UTC- 03:00)
- Amerika/Argentina/La_Rioja (UTC- 03:00)
- Amerika/Belem (UTC- 03:00)
- Amerika/Jujuy (UTC- 03:00)
- Amerika/Bahia (UTC- 03:00)
- Amerika/Goose_Bay (UTC- 03:00)
- Amerika/Argentina/San_Juan (UTC- 03:00)
- Amerika/Argentina/ ComodRivadavia (UTC- 03:00)
- Amerika/Argentina/Tucumán (UTC- 03:00)
- Amerika/Rosario (UTC- 03:00)
- SystemV/AST4ADT (UTC- 03:00)
- Amerika/Argentina/Buenos_Aires (UTC- 03:00)
- Amerika/St_Johns (UTC- 02:30)
- Kanada/Newfoundland (UTC- 02:30)
- Amerika/Miquelon (UTC- 02:00)
- DII/GMT+2 (UTC- 02:00)
- Amerika/Godthab (UTC- 02:00)
- Amerika/Noronha (UTC- 02:00)
- Brasil/ DeNoronha (UTC- 02:00)
- Atlantik/South_Georgia (UTC- 02:00)
- DII/GMT+1 (UTC- 01:00)
- Atlantik/Cape_Verde (UTC- 01:00)
- Pasifik/Kiritimati (UTC+ 14:00)
- DII/GMT-14 (UTC+ 14:00)
- Pasifik/Fakaofu (UTC+ 13:00)
- Pasifik/Enderbury (UTC+ 13:00)
- Pasifik/Apia (UTC+ 13:00)
- Pasifik/Tongatapu (UTC+ 13:00)
- DII/GMT-13 (UTC+13:00)

- NZ-OBROLAN (UTC+ 12:45)
- Pasifik/Chatham (UTC+ 12:45)
- Pasifik/Kwajalein (UTC+ 12:00)
- Antartika/ (UTC+ 12:00) McMurdo
- Pasifik/Wallis (UTC+ 12:00)
- Pasifik/Fiji (UTC+ 12:00)
- Pasifik/Funafuti (UTC+ 12:00)
- Pasifik/Nauru (UTC+ 12:00)
- Kwajalein (UTC+ 12:00)
- NZ (UTC+ 12:00)
- Pasifik/Bangun (UTC+ 12:00)
- Antartika/South_Pole (UTC+ 12:00)
- Pasifik/Tarawa (UTC+ 12:00)
- Pasifik/Auckland (UTC+ 12:00)
- Asia/Kamchatka (UTC+ 12:00)
- DII/GMT-12 (UTC+12:00)
- Asia/Anadyr (UTC+ 12:00)
- Pasifik/Majuro (UTC+ 12:00)
- Pasifik/Ponape (UTC+ 11:00)
- Pasifik/Bougainville (UTC+ 11:00)
- Antartika/Macquarie (UTC+ 11:00)
- Pasifik/Pohnpei (UTC+ 11:00)
- Pasifik/Efate (UTC+ 11:00)
- Pasifik/Norfolk (UTC+ 11:00)
- Asia/Magadan (UTC+ 11:00)
- Pasifik/Kosrae (UTC+ 11:00)
- Asia/Sakhalin (UTC+ 11:00)
- Pasifik/Noumea (UTC+ 11:00)
- DII/GMT-11 (UTC+ 11:00)

- Asia/Srednekolymusk (UTC+ 11:00)
- Pasifik/Guadalkanal (UTC+ 11:00)
- Australia/Lord_Howe (UTC+ 10:30)
- Australia/LHI (UTC+ 10:30)
- Australia/Hobart (UTC+ 10:00)
- Pasifik/Yap (UTC+ 10:00)
- Australia/Tasmania (UTC+ 10:00)
- Pasifik/Port_Moresby (UTC+ 10:00)
- Australia/ACT (UTC+10:00)
- Australia/Victoria (UTC+ 10:00)
- Pasifik/Chuuk (UTC+ 10:00)
- Australia/Queensland (UTC+ 10:00)
- Australia/Canberra (UTC+ 10:00)
- Australia/Currie (UTC+ 10:00)
- Pasifik/Guam (UTC+ 10:00)
- Pasifik/Truk (UTC+ 10:00)
- Australia/NSW (UTC+ 10:00)
- Asia/Vladivostok (UTC+ 10:00)
- Pasifik/Saipan (UTC+ 10:00)
- Antartika/Dumontdurville (UTC+ 10:00)
- Australia/Sydney (UTC+ 10:00)
- Australia/Brisbane (UTC+ 10:00)
- DII/GMT-10 (UTC+10:00)
- Asia/Ust-Nera (UTC+ 10:00)
- Australia/Melbourne (UTC+ 10:00)
- Australia/Lindeman (UTC+ 10:00)
- Australia/Utara (UTC+ 09:30)
- Australia/Yancowinna (UTC+ 09:30)
- Australia/Adelaide (UTC+ 09:30)

- Australia/Broken_Hill (UTC+ 09:30)
- Australia/Selatan (UTC+ 09:30)
- Australia/Darwin (UTC+ 09:30)
- DII/GMT-9 (UTC+ 09:00)
- Pasifik/Palau (UTC+ 09:00)
- Asia/Chita (UTC+ 09:00)
- Asia/Dili (UTC+ 09:00)
- Asia/Jayapura (UTC+ 09:00)
- Asia/Yakutsk (UTC+ 09:00)
- Asia/Pyongyang (UTC+ 09:00)
- ROK (UTC+ 09:00)
- Asia/Seoul (UTC+ 09:00)
- Asia/Khandyga (UTC+ 09:00)
- Jepang (UTC+ 09:00)
- Asia/Tokyo (UTC+ 09:00)
- Australia/Eucla (UTC+ 08:45)
- Asia/Kuching (UTC+ 08:00)
- Asia/Chungking (UTC+ 08:00)
- DII/GMT-8 (UTC+08:00)
- Australia/Perth (UTC+ 08:00)
- Asia/Makau (UTC+ 08:00)
- Asia/Makau (UTC+ 08:00)
- Asia/Choibalsan (UTC+ 08:00)
- Asia/Shanghai (UTC+ 08:00)
- Antartika/Casey (UTC+ 08:00)
- Asia/Ulan_Bator (UTC+ 08:00)
- Asia/Chongqing (UTC+ 08:00)
- Asia/Ulaanbaatar (UTC+ 08:00)
- Asia/Taipei (UTC+ 08:00)

- Asia/Manila (UTC+ 08:00)
- PRC (UTC+ 08:00)
- Asia/Ujung_Pandang (UTC+ 08:00)
- Asia/Harbin (UTC+ 08:00)
- Singapura (UTC+ 08:00)
- Asia/Brunei (UTC+ 08:00)
- Australia/Barat (UTC+ 08:00)
- Asia/Hong Kong (UTC+ 08:00)
- Asia/Makassar (UTC+ 08:00)
- Hong Kong (UTC+ 08:00)
- Asia/Kuala_Lumpur (UTC+ 08:00)
- Asia/Irkutsk (UTC+ 08:00)
- Asia/Singapura (UTC+ 08:00)
- Asia/Pontianak (UTC+ 07:00)
- DII/GMT-7 (UTC+ 07:00)
- Asia/Phnom_Penh (UTC+ 07:00)
- Asia/Novosibirsk (UTC+ 07:00)
- Antartika/Davis (UTC+ 07:00)
- Asia/Tomsk (UTC+ 07:00)
- Asia/Jakarta (UTC+ 07:00)
- Asia/Barnaul (UTC+ 07:00)
- India/Natal (UTC+ 07:00)
- Asia/Ho_Chi_Minh (UTC+ 07:00)
- Asia/Hovd (UTC+ 07:00)
- Asia/Bangkok (UTC+ 07:00)
- Asia/Vientiane (UTC+ 07:00)
- Asia/Novokuznetsk (UTC+ 07:00)
- Asia/Krasnoyarsk (UTC+ 07:00)
- Asia/Saigon (UTC+ 07:00)

- Asia/Yangon (UTC+ 06:30)
- Asia/Rangun (UTC+ 06:30)
- India/Cocos (UTC+ 06:30)
- Asia/Kashgar (UTC+ 06:00)
- DII/GMT-6 (UTC+ 06:00)
- Asia/Almaty (UTC+ 06:00)
- Asia/Dacca (UTC+ 06:00)
- Asia/Omsk (UTC+ 06:00)
- Asia/Dhaka (UTC+ 06:00)
- India/Chagos (UTC+ 06:00)
- Asia/Qyzylorda (UTC+ 06:00)
- Asia/Bishkek (UTC+ 06:00)
- Antartika/Vostok (UTC+ 06:00)
- Asia/Urumqi (UTC+ 06:00)
- Asia/Thimbu (UTC+ 06:00)
- Asia/Thimphu (UTC+ 06:00)
- Asia/Kathmandu (UTC+ 05:45)
- Asia/Katmandu (UTC+ 05:45)
- Asia/Kolkata (UTC+ 05:30)
- Asia/Kolombo (UTC+ 05:30)
- Asia/Kalkuta (UTC+ 05:30)
- Asia/Aqtau (UTC+ 05:00)
- DII/GMT-5 (UTC+ 05:00)
- Asia/Samarkand (UTC+ 05:00)
- Asia/Karachi (UTC+ 05:00)
- Asia/Yekaterinburg (UTC+ 05:00)
- Asia/Dushanbe (UTC+ 05:00)
- India/Maladewa (UTC+ 05:00)
- Asia/Lisan (UTC+ 05:00)

- Asia/Tashkent (UTC+ 05:00)
- Antartika/Mawson (UTC+ 05:00)
- Asia/Aqtobe (UTC+ 05:00)
- Asia/Ashkhabad (UTC+ 05:00)
- Asia/Ashgabat (UTC+ 05:00)
- Asia/Atyrau (UTC+ 05:00)
- India/Kerguelen (UTC+ 05:00)
- Iran (UTC+ 04:30)
- Asia/Teheran (UTC+ 04:30)
- Asia/Kabul (UTC+ 04:30)
- Asia/Yerevan (UTC+ 04:00)
- DII/GMT-4 (UTC+ 04:00)
- DII/GMT-4 (UTC+ 04:00)
- Asia/Dubai (UTC+ 04:00)
- India/Reuni (UTC+ 04:00)
- Eropa/Saratov (UTC+ 04:00)
- Eropa/Samara (UTC+ 04:00)
- India/Mahe (UTC+ 04:00)
- Asia/Baku (UTC+ 04:00)
- Asia/Muscat (UTC+ 04:00)
- Eropa/Volgograd (UTC+ 04:00)
- Eropa/Astrakhan (UTC+ 04:00)
- Asia/Tbilisi (UTC+ 04:00)
- Eropa/Ulyanovsk (UTC+ 04:00)
- Asia/Aden (UTC+ 03:00)
- Afrika/Nairobi (UTC+ 03:00)
- Eropa/Istanbul (UTC+ 03:00)
- DII/GMT-3 (UTC+ 03:00)
- Eropa/Zaporozhye (UTC+ 03:00)

- Israel (UTC+ 03:00)
- India/Comoro (UTC+ 03:00)
- Antartika/Syowa (UTC+ 03:00)
- Afrika/Mogadishu (UTC+ 03:00)
- Eropa/Bukares (UTC+ 03:00)
- Afrika/Asmera (UTC+ 03:00)
- Eropa/Mariehamn (UTC+ 03:00)
- Asia/Istanbul (UTC+ 03:00)
- Eropa/Tiraspol (UTC+ 03:00)
- Eropa/Moskow (UTC+ 03:00)
- Eropa/Chisinau (UTC+ 03:00)
- Eropa/Helsinki (UTC+ 03:00)
- Asia/Beirut (UTC+ 03:00)
- Asia/Tel_Aviv (UTC+ 03:00)
- Afrika/Djibouti (UTC+ 03:00)
- Eropa/Simferopol (UTC+ 03:00)
- Eropa/Sofia (UTC+ 03:00)
- Asia/Gaza (UTC+ 03:00)
- Afrika/Asmara (UTC+ 03:00)
- Eropa/Riga (UTC+ 03:00)
- Asia/Baghdad (UTC+ 03:00)
- Asia/Damaskus (UTC+ 03:00)
- Afrika/Dar_es_Salaam (UTC+ 03:00)
- Afrika/Addis_Ababa (UTC+ 03:00)
- Eropa/Uzhgorod (UTC+ 03:00)
- Asia/Yerusalem (UTC+ 03:00)
- Asia/Riyadh (UTC+ 03:00)
- Asia/Kuwait (UTC+ 03:00)
- Eropa/Kirov (UTC+ 03:00)

- Afrika/Kampala (UTC+ 03:00)
- Eropa/Minsk (UTC+ 03:00)
- Asia/Qatar (UTC+ 03:00)
- Eropa/Kiev (UTC+ 03:00)
- Asia/Bahrain (UTC+ 03:00)
- Eropa/Vilnius (UTC+ 03:00)
- India/Antananarivo (UTC+ 03:00)
- India/Mayotte (UTC+ 03:00)
- Eropa/Tallinn (UTC+ 03:00)
- Turki (UTC+ 03:00)
- Afrika/Juba (UTC+ 03:00)
- Asia/Nikosia (UTC+ 03:00)
- Asia/Keluarga (UTC+ 03:00)
- W-SU (UTC+ 03:00)
- MALAM (UTC+ 03:00)
- Asia/Hebron (UTC+ 03:00)
- Asia/Amman (UTC+ 03:00)
- Eropa/Nikosia (UTC+ 03:00)
- Eropa/Athina (UTC+ 03:00)
- Afrika/Kairo (UTC+ 02:00)
- Afrika/Mbabane (UTC+ 02:00)
- Eropa/Brussel (UTC+ 02:00)
- Eropa/Warsawa (UTC+ 02:00)
- CET (UTC+ 02:00)
- Eropa/Luksemburg (UTC+ 02:00)
- DII/GMT-2 (UTC+ 02:00)
- Libya (UTC+ 02:00)
- Afrika/Kigali (UTC+ 02:00)
- Afrika/Tripoli (UTC+ 02:00)

- Eropa/Kaliningrad (UTC+ 02:00)
- Afrika/Windhoek (UTC+ 02:00)
- Eropa/Malta (UTC+ 02:00)
- Eropa/Busingen (UTC+ 02:00)
-
- Eropa/Skopje (UTC+ 02:00)
- Eropa/Sarajevo (UTC+ 02:00)
- Eropa/Roma (UTC+ 02:00)
- Eropa/Zürich (UTC+ 02:00)
- Eropa/Gibraltar (UTC+ 02:00)
- Afrika/Lubumbashi (UTC+ 02:00)
- Eropa/Vaduz (UTC+ 02:00)
- Eropa/Ljubljana (UTC+ 02:00)
- Eropa/Berlin (UTC+ 02:00)
- Eropa/Stockholm (UTC+ 02:00)
- Eropa/Budapest (UTC+ 02:00)
- Eropa/Zagreb (UTC+ 02:00)
- Eropa/Paris (UTC+ 02:00)
- Afrika/Ceuta (UTC+ 02:00)
- Eropa/Praha (UTC+ 02:00)
- Antartika/Troll (UTC+ 02:00)
- Afrika/Gaborone (UTC+ 02:00)
- Eropa/Kopenhagen (UTC+ 02:00)
- Eropa/Wina (UTC+ 02:00)
- Eropa/Tirane (UTC+ 02:00)
- BERTEMU (UTC+ 02:00)
- Eropa/Amsterdam (UTC+ 02:00)
- Afrika/Maputo (UTC+ 02:00)
- Eropa/San_Marino (UTC+ 02:00)

- Polandia (UTC+ 02:00)
- Eropa/Andorra (UTC+ 02:00)
- Eropa/Oslo (UTC+ 02:00)
- Eropa/Podgorica (UTC+ 02:00)
- Afrika/Bujumbura (UTC+ 02:00)
- Atlantik/Jan_Mayen (UTC+ 02:00)
- Afrika/Maseru (UTC+ 02:00)
- Eropa/Madrid (UTC+ 02:00)
- Afrika/Blantyre (UTC+ 02:00)
- Afrika/Lusaka (UTC+ 02:00)
- Afrika/Harare (UTC+ 02:00)
- Afrika/Khartoum (UTC+ 02:00)
- Afrika/Johannesburg (UTC+ 02:00)
- Eropa/Beograd (UTC+ 02:00)
- Eropa/Bratislava (UTC+ 02:00)
- Arktik/Longyearbyen (UTC+ 02:00)
- Mesir (UTC+ 02:00)
- Eropa/Vatikan (UTC+ 02:00)
- Eropa/Monako (UTC+ 02:00)
- Eropa/London (UTC+ 01:00)
- DII/GMT-1 (UTC+ 01:00)
- Eropa/Jersey (UTC+ 01:00)
- Eropa/Guernsey (UTC+ 01:00)
- Eropa/Isle_of_man (UTC+ 01:00)
- Afrika/Tunis (UTC+ 01:00)
- Afrika/Malabo (UTC+ 01:00)
- GB-Eire (UTC+ 01:00)
- Afrika/Lagos (UTC+ 01:00)
- Afrika/Aljir (UTC+ 01:00)

- GB (UTC+ 01:00)
- Portugal (UTC+ 01:00)
- Afrika/Sao_Tome (UTC+ 01:00)
- Afrika/Ndjamena (UTC+ 01:00)
- Atlantik/Faeroe (UTC+ 01:00)
- Eire (UTC+ 01:00)
- Atlantik/Faroe (UTC+ 01:00)
- Eropa/Dublin (UTC+ 01:00)
- Afrika/Libreville (UTC+ 01:00)
- Afrika/EI_Aaiun (UTC+ 01:00)
- Afrika/EI_Aaiun (UTC+ 01:00)
- Afrika/Douala (UTC+ 01:00)
- Afrika/Brazzaville (UTC+ 01:00)
- Afrika/Porto-Novo (UTC+ 01:00)
- Atlantik/Madeira (UTC+ 01:00)
- Eropa/Lisbon (UTC+ 01:00)
- Atlantik/Kenari (UTC+ 01:00)
- Afrika/Casablanca (UTC+ 01:00)
- Eropa/Belfast (UTC+ 01:00)
- Afrika/Luanda (UTC+ 01:00)
- Afrika/Kinshasa (UTC+ 01:00)
- Afrika/Bangui (UTC+ 01:00)
- BASAH (UTC+ 01:00)
- Afrika/Niamey (UTC+ 01:00)
- GMT (UTC+ 00:00)
- DII/GMT-0 (UTC+ 00:00)
- Atlantik/St_Helena (UTC+ 00:00)
- DII/GMT+0 (UTC+ 00:00)
- Afrika/Banjul (UTC+ 00:00)

- DII/GMT (UTC+00:00)
- Afrika/Freetown (UTC+ 00:00)
- Afrika/Bamako (UTC+ 00:00)
- Afrika/Conakry (UTC+ 00:00)
- Universal (UTC+ 00:00)
- Afrika/Nouakchott (UTC+ 00:00)
- UTC (UTC+ 00:00)
- DII/Universal (UTC+00:00)
- Atlantik/Azores (UTC+ 00:00)
- Afrika/Abidjan (UTC+ 00:00)
- Afrika/Accra (UTC+ 00:00)
- DII/UCT (UTC+00:00)
- GMT0 (UTC+ 00:00)
- Zulu (UTC+ 00:00) Zulu (UTC+ 00:00)
- Afrika/Ouagadougou (UTC+ 00:00)
- Atlantik/Reykjavik (UTC+ 00:00)
- DII/Zulu (UTC+ 00:00)
- Islandia (UTC+ 00:00)
- Afrika/Lome (UTC+ 00:00)
- Greenwich (UTC+ 00:00)
- DII/GMT0 (UTC+ 00:00)
- Amerika/Danmarkshavn (UTC+ 00:00)
- Afrika/Dakar (UTC+ 00:00)
- Afrika/Bissau (UTC+ 00:00)
- DII/Greenwich (UTC+ 00:00)
- Afrika/Timbuktu (UTC+ 00:00)
- UCT (UTC+ 00:00)
- Afrika/Monrovia (UTC+ 00:00)
- DII/UTC (UTC+ 00:00)

Tutorial ekspresi rumus

Anda dapat mengikuti tutorial ini untuk menggunakan ekspresi rumus di AWS IoT SiteWise.

Topik

- [Menggunakan string dalam rumus](#)
- [Memfilter titik data](#)
- [Menghitung titik data yang cocok dengan suatu kondisi](#)
- [Data terlambat dalam rumus](#)
- [Kualitas data dalam rumus](#)
- [Nilai tidak terdefinisi, tak terbatas, dan meluap](#)

Menggunakan string dalam rumus

Anda dapat beroperasi pada string dalam ekspresi rumus Anda. Anda juga dapat memasukkan string dari variabel yang mereferensikan atribut dan properti pengukuran.

Important

Ekspresi rumus hanya dapat menampilkan nilai ganda atau string. Ekspresi bersarang dapat menampilkan tipe data lain, seperti string, tetapi rumus secara keseluruhan harus mengevaluasi ke angka atau string. Anda dapat menggunakan [fungsi jp](#) untuk mengonversi string ke angka. Nilai Boolean harus 1 (true) atau 0 (false). Untuk informasi selengkapnya, lihat [Nilai tidak terdefinisi, tak terbatas, dan meluap](#).

AWS IoT SiteWise menyediakan fitur ekspresi rumus berikut yang dapat Anda gunakan untuk beroperasi pada string:

- [String literal](#)
- [Operator indeks](#) (`s[index]`)
- [Operator irisan](#) (`s[start:end:step]`)
- [Fungsi perbandingan](#), yang dapat Anda gunakan membandingkan string berdasarkan urutan leksikografis
- [Fungsi string](#), yang mencakup `jp` fungsi yang dapat mengurai objek JSON serial dan mengonversi string menjadi angka

Memfilter titik data

Anda dapat menggunakan [fungsi if](#) untuk menyaring titik data yang tidak memenuhi kondisi.

`if` Fungsi mengevaluasi kondisi dan mengembalikan nilai yang berbeda untuk `true` dan `false` hasil. Anda dapat menggunakan [konstanta none](#) sebagai output untuk satu kasus `if` fungsi untuk membuang titik data untuk kasus itu.

Untuk memfilter titik data yang cocok dengan kondisi

- Buat transformasi yang menggunakan `if` fungsi untuk menentukan kondisi yang memeriksa apakah suatu kondisi terpenuhi, dan kembali `none` sebagai `result_if_false` nilai `result_if_true` atau.

Example Contoh: Saring titik data di mana air tidak mendidih

Pertimbangkan skenario di mana Anda memiliki pengukuran, `temp_c`, yang menyediakan suhu (dalam Celcius) air dalam mesin. Anda dapat menentukan transformasi berikut untuk menyaring titik data di mana air tidak mendidih:

- Transform: `boiling_temps = if(gte(temp_c, 100), temp_c, none)` — Mengembalikan suhu jika lebih besar dari atau sama dengan 100 derajat Celcius, jika tidak mengembalikan tidak ada titik data.

Menghitung titik data yang cocok dengan suatu kondisi

Anda dapat menggunakan [fungsi perbandingan](#) dan [sum \(\)](#) untuk menghitung jumlah titik data yang kondisinya benar.

Untuk menghitung titik data yang cocok dengan suatu kondisi

1. Buat transformasi yang menggunakan fungsi perbandingan untuk menentukan kondisi filter pada properti lain.
2. Buat metrik yang menjumlahkan titik data di mana kondisi itu terpenuhi.

Example Contoh: Hitung jumlah titik data di mana air mendidih

Pertimbangkan skenario di mana Anda memiliki pengukuran, `temp_c`, yang menyediakan suhu (dalam Celcius) air dalam mesin. Anda dapat menentukan sifat transformasi dan metrik berikut untuk menghitung jumlah titik data tempat air mendidih:

- Transform: `is_boiling = gte(temp_c, 100)` — Kembali 1 jika suhu lebih besar dari atau sama dengan 100 derajat Celcius, jika tidak kembali 0.
- Metrik: `boiling_count = sum(is_boiling)` — Mengembalikan jumlah titik data di mana air mendidih.

Data terlambat dalam rumus

AWS IoT SiteWise mendukung keterlambatan konsumsi data yang berusia hingga 7 hari. Saat AWS IoT SiteWise menerima data terlambat, itu menghitung ulang nilai yang ada untuk metrik apa pun yang memasukkan data akhir di jendela sebelumnya. Penghitungan ulang ini menghasilkan biaya pemrosesan data.

Note

Saat AWS IoT SiteWise menghitung properti yang memasukkan data terlambat, ia menggunakan ekspresi rumus setiap properti saat ini.

Setelah AWS IoT SiteWise menghitung ulang jendela sebelumnya untuk metrik, itu menggantikan nilai sebelumnya untuk jendela itu. Jika Anda mengaktifkan notifikasi untuk metrik tersebut, AWS IoT SiteWise juga akan memancarkan notifikasi nilai properti. Ini berarti Anda dapat menerima pemberitahuan pembaruan nilai properti baru untuk properti dan stempel waktu yang sama yang sebelumnya Anda terima notifikasi. Jika aplikasi atau data lake Anda menggunakan notifikasi nilai properti, Anda harus memperbarui nilai sebelumnya dengan nilai baru sehingga datanya akurat.

Kualitas data dalam rumus

Di AWS IoT SiteWise, setiap titik data memiliki kode kualitas, yang dapat berupa salah satu dari yang berikut:

- GOOD— Data tidak terpengaruh oleh masalah apa pun.
- BAD— Data dipengaruhi oleh masalah seperti kegagalan sensor.
- UNCERTAIN— Data dipengaruhi oleh masalah seperti ketidakakuratan sensor.

AWS IoT SiteWise hanya mengkonsumsi data GOOD berkualitas ketika menghitung transformasi dan metrik. AWS IoT SiteWise hanya menghasilkan data GOOD berkualitas untuk perhitungan yang berhasil. Jika komputasi tidak berhasil, maka AWS IoT SiteWise tidak menampilkan titik data untuk

perhitungan itu. Hal ini dapat terjadi jika perhitungan menghasilkan nilai undefined, infinite, atau overflow.

Untuk informasi selengkapnya tentang cara menanyakan data dan memfilter berdasarkan kualitas data, lihat [Data kueri dari AWS IoT SiteWise](#).

Nilai tidak terdefinisi, tak terbatas, dan meluap

Beberapa ekspresi rumus (seperti $x / 0$, $\text{sqrt}(-1)$, atau $\text{log}(0)$) menghitung nilai yang tidak terdefinisi dalam sistem bilangan real, tak terbatas, atau di luar rentang yang didukung oleh AWS IoT SiteWise. Ketika ekspresi properti aset menghitung nilai undefined, infinite, atau overflow, AWS IoT SiteWise tidak menampilkan titik data untuk perhitungan tersebut.

AWS IoT SiteWise juga tidak menampilkan titik data jika menghitung nilai non-numerik sebagai hasil dari ekspresi rumus. Ini berarti bahwa jika Anda mendefinisikan rumus yang menghitung string, array, atau [konstanta none](#), maka AWS IoT SiteWise tidak menampilkan titik data untuk perhitungan itu.

Example Contoh-contoh

Masing-masing ekspresi rumus berikut menghasilkan nilai yang tidak AWS IoT SiteWise dapat mewakili sebagai angka. AWS IoT SiteWise tidak menampilkan titik data saat menghitung ekspresi rumus ini.

- $x / 0$ tidak terdefinisi.
- $\text{log}(0)$ tidak terdefinisi.
- $\text{sqrt}(-1)$ tidak terdefinisi dalam sistem bilangan real.
- "hello" + " world" adalah string.
- `jp({'values':[3,6,7]}, '$.values')` adalah sebuah array.
- `if(gte(temp, 300), temp, none)` temp adalah none kapan kurang dari 300.

Membuat model komposit khusus (Komponen)

Model komposit khusus, atau komponen jika Anda menggunakan konsol, menyediakan tingkat organisasi lain untuk model aset dan model komponen Anda. Anda dapat menggunakannya untuk menyusun model Anda dengan mengelompokkan properti atau mereferensikan model lain. Untuk informasi selengkapnya tentang bekerja dengan model komposit khusus, lihat [Model komposit khusus \(Komponen\)](#).

Anda membuat model komposit khusus dalam model aset atau model komponen yang ada. Ada dua jenis model komposit khusus. Untuk mengelompokkan properti terkait dalam model, Anda dapat membuat model komposit kustom sebaris. Untuk mereferensikan model komponen dalam model aset atau model komponen, Anda dapat membuat model komposit component-model-based khusus.

Bagian berikut menjelaskan cara menggunakan AWS IoT SiteWise API untuk membuat model komposit khusus.

Topik

- [Membuat komponen inline \(konsol\)](#)
- [Membuat model komposit kustom inline \(AWS CLI\)](#)
- [Membuat component-model-based komponen \(konsol\)](#)
- [Membuat model komposit component-model-based khusus \(AWS CLI\)](#)

Membuat komponen inline (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk membuat komponen inline yang mendefinisikan propertinya sendiri.

Note

Karena ini adalah komponen inline, properti ini hanya berlaku untuk model aset saat ini dan tidak dibagikan di tempat lain.

Jika Anda perlu menghasilkan model yang dapat digunakan kembali (misalnya, untuk berbagi di antara beberapa model aset, atau menyertakan beberapa instance dalam satu model aset), Anda harus membuat komponen berdasarkan model komponen sebagai gantinya. Lihat bagian berikut untuk detailnya.

Untuk membuat komponen (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model.
3. Pilih model aset yang ingin Anda tambahkan komponennya.
4. Pada tab Properties, pilih Components.
5. Pilih Buat komponen.

6. Pada halaman Create component, lakukan hal berikut:
 - a. Masukkan Nama untuk komponen, seperti **ServoMotor** atau **ServoMotor Model**. Nama ini harus unik di semua komponen di akun Anda di Wilayah ini.
 - b. (Opsional) Tambahkan definisi Atribut untuk model. Atribut mewakili informasi yang jarang berubah. Untuk informasi selengkapnya, lihat [Mendefinisikan data statis \(atribut\)](#).
 - c. (Opsional) Tambahkan definisi Pengukuran untuk model. Pengukuran mewakili aliran data dari peralatan Anda. Untuk informasi selengkapnya, lihat [Mendefinisikan aliran data dari peralatan \(pengukuran\)](#).
 - d. (Opsional) Tambahkan definisi Transform untuk model. Transformasi adalah rumus yang memetakan data dari satu bentuk ke bentuk lainnya. Untuk informasi selengkapnya, lihat [Mengubah data \(transformasi\)](#).
 - e. (Opsional) Tambahkan definisi Metrik untuk model. Metrik adalah rumus yang mengumpulkan data dari interval waktu. Metrik dapat memasukkan data dari aset terkait, sehingga Anda dapat menghitung nilai yang mewakili operasi Anda atau bagian dari operasi Anda. Untuk informasi selengkapnya, lihat [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#).
 - f. Pilih Buat komponen.

Membuat model komposit kustom inline (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk membuat model komposit kustom inline yang mendefinisikan propertinya sendiri.

Gunakan [CreateAssetModelCompositeModel](#) operasi untuk membuat model inline dengan properti. Operasi ini mengharuskan muatan dengan struktur berikut.

Note

Karena ini adalah model komposit sebaris, properti ini hanya berlaku untuk model aset saat ini dan tidak dibagikan di tempat lain. Apa yang membuatnya “sebaris” adalah bahwa ia tidak memberikan nilai untuk `composedAssetModelId` bidang tersebut.

Jika Anda perlu menghasilkan model yang dapat digunakan kembali (misalnya, untuk berbagi di antara beberapa model aset, atau menyertakan beberapa contoh dalam satu model aset), Anda harus membuat model component-model-based komposit sebagai gantinya. Lihat bagian berikut untuk detailnya.

```
{
  "assetModelCompositeModelName": "CNCLathe_ServoMotorA",
  "assetModelCompositeModelType": "CUSTOM",
  "assetModelCompositeModelProperties": [
    {
      "dataType": "DOUBLE",
      "name": "Servo Motor Temperature",
      "type": {
        "measurement": {}
      },
      "unit": "Celsius"
    },
    {
      "dataType": "DOUBLE",
      "name": "Spindle speed",
      "type": {
        "measurement": {}
      },
      "unit": "rpm"
    }
  ]
}
```

Membuat component-model-based komponen (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk membuat komponen berdasarkan model komponen.

Untuk membuat component-model-based komponen (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model.
3. Pilih model aset yang ingin Anda tambahkan komponennya.
4. Pada tab Properties, pilih Components.
5. Pilih Buat komponen.
6. Pada halaman Create component, lakukan hal berikut:
 - a. Pilih model komponen yang ingin Anda gunakan berdasarkan komponennya.
 - b. Masukkan Nama untuk komponen, seperti **ServoMotor** atau **ServoMotor Model**. Nama ini harus unik di semua komponen di akun Anda di Wilayah ini.

- c. Pilih Buat komponen.

Membuat model komposit component-model-based khusus (AWS CLI)

Anda dapat menggunakan AWS CLI untuk membuat model komposit component-model-based khusus dalam model aset Anda. Model komposit component-model-based kustom adalah referensi ke model komponen yang telah Anda definisikan di tempat lain.

Gunakan [CreateAssetModelCompositeModel](#) operasi untuk membuat model komposit component-model-based khusus. Operasi ini mengharapkan muatan dengan struktur berikut.

Note

Dalam contoh ini, nilai `composedAssetModelId` adalah ID model aset atau ID eksternal dari model komponen yang ada. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna. Untuk contoh cara membuat model komponen, lihat [Membuat model komponen \(AWS CLI\)](#).

```
{
  "assetModelCompositeModelName": "CNCLathe_ServoMotorA",
  "assetModelCompositeModelType": "CUSTOM",
  "composedAssetModelId": component model ID
}
```

Karena ini hanya referensi, model komposit component-model-based khusus tidak memiliki properti sendiri, selain nama.

Jika Anda ingin menambahkan beberapa contoh komponen yang sama ke model aset Anda (misalnya, mesin CNC yang memiliki beberapa motor servo), Anda dapat menambahkan beberapa model komposit component-model-based khusus yang masing-masing memiliki nama mereka sendiri tetapi yang semuanya mereferensikannya sama. `composedAssetModelId`

Anda dapat membuat sarang komponen di dalam komponen lain. Untuk melakukannya, Anda dapat menambahkan model component-model-based komposit, seperti yang ditunjukkan dalam contoh ini, ke salah satu model komponen Anda.

Menciptakan aset

Anda dapat membuat aset dari model aset. Anda harus memiliki model aset sebelum Anda dapat membuat aset. Jika Anda belum membuat model aset, lihat [Membuat model aset](#).

Note

Anda hanya dapat membuat aset dari ACTIVE model. Jika status model Anda tidak ACTIVE, Anda mungkin perlu menunggu hingga beberapa menit sebelum Anda dapat membuat aset dari model itu. Untuk informasi selengkapnya, lihat [Status aset dan model](#).

Topik

- [Membuat aset \(konsol\)](#)
- [Membuat aset \(AWS CLI\)](#)
- [Mengkonfigurasi aset baru](#)

Membuat aset (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk membuat aset.

Untuk membuat aset (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih Buat aset.
4. Pada halaman Buat aset, lakukan hal berikut:
 - a. Untuk Model, pilih model aset untuk membuat aset.

Note

Jika model Anda tidak AKTIF, Anda harus menunggu hingga aktif, atau menyelesaikan masalah jika gagal.

- b. Masukkan Nama untuk aset Anda.

- c. (Opsional) Tambahkan tag untuk aset Anda. Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda AWS IoT SiteWise](#).
- d. Pilih Buat aset.

Saat Anda membuat aset, AWS IoT SiteWise konsol akan menavigasi ke halaman aset baru. Di halaman ini, Anda dapat melihat Status aset, yang awalnya DIBUAT. Halaman ini diperbarui secara otomatis, sehingga Anda dapat menunggu status aset diperbarui.

Note

Proses pembuatan aset bisa memakan waktu hingga satu menit. Setelah Status AKTIF, Anda dapat melakukan operasi pembaruan pada aset Anda. Untuk informasi selengkapnya, lihat [Status aset dan model](#).

Setelah Anda membuat aset, lihat [Mengkonfigurasi aset baru](#).

Membuat aset (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk membuat aset dari model aset.

Anda harus memiliki `assetModelId` untuk membuat aset. Jika Anda membuat model aset, tetapi tidak mengetahuinya `assetModelId`, gunakan [ListAssetModels](#) API untuk melihat semua model aset Anda.

Untuk membuat aset dari model aset, gunakan [CreateAsset](#) API dengan parameter berikut:

- `assetName`— Nama aset baru. Beri nama aset Anda untuk membantu Anda mengidentifikasinya.
- `assetModelId`— ID aset. Ini adalah ID aktual dalam format UUID, atau `externalId:myExternalId` jika memilikinya. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

Untuk membuat aset (AWS CLI)

- Jalankan perintah berikut untuk membuat aset. Ganti *nama aset* dengan nama untuk aset dan *asset-model-id* dengan ID atau ID eksternal model aset.

```
aws iotsitewise create-asset \  
  --asset-name asset-name \  
  --asset-model-id asset-model-id
```

Operasi mengembalikan respons yang berisi detail dan status aset baru Anda dalam format berikut.

```
{  
  "assetId": "String",  
  "assetArn": "String",  
  "assetStatus": {  
    "state": "String",  
    "error": {  
      "code": "String",  
      "message": "String"  
    }  
  }  
}
```

Aset state adalah CREATING sampai aset dibuat.

Note

Proses pembuatan aset bisa memakan waktu hingga satu menit. Untuk memeriksa status aset Anda, gunakan [DescribeAsset](#) operasi dengan ID aset Anda sebagai `assetId` parameter. Setelah aset state selesai ACTIVE, Anda dapat melakukan operasi pembaruan pada aset Anda. Untuk informasi selengkapnya, lihat [Status aset dan model](#).

Setelah Anda membuat aset, lihat [Mengkonfigurasi aset baru](#).

Mengkonfigurasi aset baru

Selesai mengonfigurasi aset Anda dengan salah satu tindakan opsional berikut:

- [Memetakan aliran data industri ke properti aset](#) jika aset Anda memiliki properti pengukuran.
- [Memperbarui nilai atribut](#) jika aset Anda memiliki nilai atribut yang unik.

- [Mengasosiasikan dan memisahkan aset](#) jika aset Anda adalah aset induk.

Mencari aset

Gunakan fungsionalitas Konsol AWS IoT SiteWise pencarian untuk menemukan aset berdasarkan metadata dan filter nilai properti real-time.

Prasyarat

AWS IoT SiteWise memerlukan izin untuk berintegrasi dengan AWS IoT TwinMaker mengatur, dan memodelkan data industri dengan lebih baik. Jika Anda telah memberikan izin untuk AWS IoT SiteWise, gunakan [ExecuteQuery](#) API. Jika Anda belum memberikan izin untuk AWS IoT SiteWise, dan membutuhkan bantuan untuk memulai, lihat [Mengintegrasikan dan AWS IoT SiteWise AWS IoT TwinMaker](#).

Pencarian lanjutan di Konsol AWS IoT SiteWise

Pencarian metadata

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Pencarian lanjutan di bawah Aset.
3. Di bawah Pencarian lanjutan pilih opsi pencarian Metadata.
4. Isi parameternya. Isi kolom sebanyak mungkin untuk pencarian yang efisien.
 - a. Nama aset — Masukkan nama aset lengkap, atau sebagian nama untuk pencarian luas.
 - b. Nama properti — Masukkan nama properti lengkap, atau sebagian nama untuk pencarian luas.
 - c. Operator — Pilih operator dari:
 - =
 - <
 - >
 - <=
 - >=
 - d. Nilai properti — Nilai ini dibandingkan dengan nilai terbaru properti.
 - e. Jenis nilai properti — Tipe data properti. Pilih dari yang berikut ini:

- ganda
- bilangan bulat
- Tali
- Boolean

5. Pilih Cari.

6. Dari tabel Hasil pencarian, pilih aset dari kolom Nama. Ini membawa Anda ke halaman aset terperinci untuk aset itu.

The screenshot shows the AWS IoT SiteWise console interface. At the top, there's a navigation bar with the AWS logo, 'Services', a search bar, and a region dropdown set to 'N. Virginia'. Below this, the breadcrumb 'IoT SiteWise > Assets' is visible. The main heading is 'Assets', with a 'Create asset' button on the right. A descriptive paragraph explains that assets represent industrial devices and processes. Below this is the 'Advanced search' section, which includes a 'Metadata search' tab and a 'Query builder' tab. The query builder has five fields: 'Asset name' (Level-2), 'Property name' (power_max), 'Operator' (>), 'Property value' (20), and 'Property value type' (Double). A 'Search' button is highlighted. Below the search criteria, the 'Search results (2)' section shows a table with two rows:

Name	Asset id	Description
Level-2-asset-1	d0e9019b-9c38-4316-b574-38317aa38143	
Level-2-asset-2	b9c0d2fc-1527-42ce-8ba2-d1a4e8ff43de	Example description

Pencarian parsial

Semua parameter tidak perlu disediakan untuk pencarian aset. Berikut adalah beberapa contoh pencarian sebagian menggunakan opsi pencarian Metadata:

- Temukan aset dengan namanya:
 - Masukkan nilai di bidang Asset name saja.
 - Nama properti dan kolom nilai Properti kosong.
- Temukan aset yang berisi properti dengan nama tertentu:

- Masukkan nilai di bidang Nama properti saja.
- Bidang Asset name dan Property value kosong.
- Temukan aset berdasarkan nilai terbaru dari properti mereka:
 - Masukkan nilai di bidang Nama properti dan nilai Properti.
 - Pilih jenis nilai Operator dan Properti.

Pencarian pembuat kueri

1. Arahkan ke Konsol AWS IoT SiteWise.
2. Di panel navigasi, pilih Pencarian lanjutan di bawah Aset.
3. Di bawah Pencarian lanjutan pilih opsi Pembuat kueri.
4. Di panel Query builder, tulis kueri SQL Anda untuk mengambil, dan. `asset_name asset_id asset_description`
5. Pilih Cari.
6. Dari tabel Hasil pencarian, pilih aset dari kolom Nama. Ini membawa Anda ke halaman aset terperinci untuk aset itu.

The screenshot shows the AWS IoT SiteWise 'Assets' page. At the top, there's a navigation bar with the AWS logo, 'Services', a search bar, and a region dropdown set to 'N. Virginia'. Below the navigation, the page title is 'Assets' with a 'Create asset' button. A brief description states: 'Assets represent industrial devices and processes that send data streams to SiteWise. Models are structures that enforce a specific model of properties and hierarchies for all instances of each asset. You must create every asset from a model.'

The 'Advanced search' section is active, showing a 'Query builder' tab. The query entered is:


```
SELECT a.asset_id, a.asset_name, a.asset_description
FROM asset a, asset_property p, latest_value_time_series ts
WHERE a.asset_name LIKE '%asset-2%' AND a.property_name = 'temperature_f' AND ts.double_value > 50.0
```

 Below the query builder, there are 'Clear' and 'Search' buttons. The search results section shows 2 results in a table:

Name	Asset id	Description
Level-2a-asset-2	4fed596d-e903-4338-86db-34ca9301233a	Generator #3
Level-2b-asset-2	b4ac2b24-4fce-4a72-9fea-ef6d0f741e8d	Generator #2

Note

- SELECTKlausa dalam kueri SQL harus menyertakan `asset_id` bidang `asset_name` dan untuk memastikan aset yang valid dalam tabel hasil Penelusuran.
- Pembuat Query hanya menampilkan Name, Asset id, dan Description dalam tabel hasil. Menambahkan lebih banyak bidang ke SELECT klausa tidak menambahkan lebih banyak kolom ke tabel hasil

Memetakan aliran data industri ke properti aset

Anda dapat menentukan alias properti pada properti aset. Ini membantu Anda mengidentifikasi properti aset saat Anda menelan atau mengambil data aset. Jika aset Anda memiliki properti pengukuran, Anda dapat menentukan alias properti untuk memetakan aliran data Anda ke properti pengukuran tersebut.

Proses ini mengharuskan Anda mengetahui alias properti Anda.

- Jika Anda menyerap data dari server OPC-UA menggunakan [sumber data OPC-UA di gateway SiteWise Edge](#), alias properti Anda adalah jalur ke variabel di bawah node Objects, dimulai dengan. /

Example

Jika jalur ke variabel Anda adalah `company/windfarm/3/turbine/7/temperature`, maka alias properti Anda adalah `/company/windfarm/3/turbine/7/temperature`.

Untuk informasi selengkapnya tentang arsitektur informasi OPC-UA, lihat [Pemetaan Model Informasi dan Spasi Alamat](#) di Referensi Online OPC UA.

Catatan

- Jika Anda mengonfigurasi awalan aliran data untuk sumber OPC-UA Anda, Anda harus menyertakan awalan tersebut dalam alias properti untuk semua aliran data dari sumber tersebut.

Example

Jika `/RentonWA` adalah awalan, maka alias sebelumnya adalah `/RentonWA/company/windfarm/3/turbine/7/temperature`

- Alias properti dapat berisi hingga 1.000 byte. Jalur variabel OPC-UA dapat berisi hingga 4.096 byte. Saat ini, AWS IoT SiteWise tidak mendukung pengambilan data dari variabel OPC-UA dengan jalur panjang.

- Jika Anda menelan data dari server Modbus menggunakan [sumber data Modbus TCP di gateway SiteWise Edge](#), alias properti Anda adalah:

`Modbus register set tag name`

Gunakan nilai ini untuk mengirim data dari register ini yang disetel ke properti aset.

- Jika Anda menyerap data dari sumber lain, seperti menggunakan [AWS IoT aturan](#) atau [API](#), Anda harus menentukan alias properti Anda. Anda dapat menentukan sistem penamaan alias properti yang berlaku untuk konfigurasi perangkat Anda. Misalnya, jika Anda menelan data dari AWS IoT berbagai hal, Anda dapat menyertakan nama benda dalam alias properti untuk mengidentifikasi aliran data secara unik. Untuk informasi lebih lanjut tentang contoh ini, lihat tutorial [Menelan data dari AWS IoT hal-hal](#).

Alias properti harus unik dalam Wilayah dan AWS akun. AWS IoT SiteWise mengembalikan kesalahan jika Anda menetapkan alias properti ke salah satu yang sudah ada di properti aset lain.

Jika Anda memiliki beberapa sumber OPC-UA dengan jalur aliran data yang identik, tambahkan awalan ke setiap jalur sumber untuk membentuk alias unik. Untuk informasi selengkapnya, lihat [Mengkonfigurasi sumber data](#).

Note

Bagian ini menjelaskan cara mengatur alias properti untuk properti pengukuran. Untuk informasi selengkapnya tentang cara menyetel alias properti untuk properti status alarm eksternal, lihat [Memetakan aliran status alarm eksternal](#).

Topik

- [Menyetel alias properti \(konsol\)](#)
- [Mengatur alias properti \(AWS CLI\)](#)

Menyetel alias properti (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk menyetel alias untuk properti aset.

Untuk menyetel alias properti (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang ingin Anda tetapkan alias properti.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih Edit.
5. Temukan properti yang ingin Anda atur alias, lalu masukkan alias properti.

"Wind Speed"

/company/windfarm/3/turbine/7/speed

Must be less than 2048 characters.

Notification status

DISABLED

Notification will be published to topic \$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/assets/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE/properties/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE

6. Pilih Simpan.

Mengatur alias properti ()AWS CLI

Gunakan AWS Command Line Interface (AWS CLI) untuk menetapkan alias untuk properti aset.

Anda harus mengetahui aset `assetId` dan properti Anda `propertyId` untuk menyelesaikan prosedur ini. Anda juga dapat menggunakan ID eksternal. Jika Anda membuat aset dan tidak mengetahuinya `assetId`, gunakan [ListAssets](#) API untuk mencantumkan semua aset untuk model tertentu. Gunakan [DescribeAsset](#) operasi untuk melihat properti aset Anda termasuk ID properti.

Gunakan [UpdateAssetProperty](#) operasi, untuk memetakan aliran data ke properti aset Anda.

Tentukan parameter berikut:

- `assetId`— ID aset atau ID eksternal. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.
- `propertyId`— ID properti aset atau ID eksternal.
- `propertyAlias`— Jalur aliran data ke alias ke properti.
- `propertyNotificationState`— Status pemberitahuan nilai properti: ENABLED atau DISABLED. Tentukan status notifikasi properti yang ada saat Anda memperbarui alias properti. Anda dapat mengambil status notifikasi yang ada dengan [DescribeAssetProperty](#) operasi.

Jika Anda menghilangkan parameter ini, status notifikasi baru adalah DISABLED. Untuk informasi selengkapnya tentang pemberitahuan properti, lihat [Berinteraksi dengan layanan lain AWS](#).

Untuk menetapkan alias properti ()AWS CLI

1. Jalankan perintah berikut untuk mengambil status notifikasi properti saat ini. Ganti *asset-id* dan *property-id* dengan ID properti aset.

```
aws iotsitewise describe-asset-property \
  --asset-id asset-id \
```

```
--property-id property-id
```

Operasi mengembalikan respons yang berisi rincian properti aset dalam format berikut. Status pemberitahuan properti ada `assetProperty.notification.state` di objek JSON.

```
{
  "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
  "assetName": "Wind Turbine 7",
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
  "assetProperty": {
    "id": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
    "name": "Wind Speed",
    "notification": {
      "topic": "$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/
assets/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE/properties/a1b2c3d4-5678-90ab-
cdef-33333EXAMPLE",
      "state": "ENABLED"
    },
    "dataType": "DOUBLE",
    "unit": "m/s",
    "type": {
      "measurement": {}
    }
  }
}
```

2. Jalankan perintah berikut untuk mengatur alias properti aset. Ganti *alias properti dengan alias* properti dan status pemberitahuan dengan *status notifikasi, atau hilangkan* untuk menonaktifkan notifikasi. `--property-notification-state` Anda dapat memperbarui unit aset secara opsional dengan *unit* baru dan `--property-unit`.

```
aws iotsitewise update-asset-property \
  --asset-id asset-id \
  --property-id property-id \
  --property-alias property-alias \
  --property-notification-state notification-state \
  --property-unit unit
```

3. Untuk memverifikasi alias telah ditetapkan, jalankan perintah berikut untuk mengambil detail properti. Ganti *asset-id* dan *property-id dengan ID* properti aset.

```
aws iotsitewise describe-asset-property \
```

```
--asset-id asset-id \  
--property-id property-id
```

Operasi mengembalikan respons yang berisi rincian properti aset dalam format berikut. Alias properti ada `assetProperty.alias` di objek JSON dan diatur ke `myAlias` dalam contoh ini.

```
{  
  "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetName": "Wind Turbine 7",  
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
  "assetProperty": {  
    "alias": "myAlias",  
    "id": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",  
    "name": "Wind Speed",  
    "notification": {  
      "topic": "$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/  
assets/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE/properties/a1b2c3d4-5678-90ab-  
cdef-33333EXAMPLE",  
      "state": "ENABLED"  
    },  
    "dataType": "DOUBLE",  
    "unit": "m/s",  
    "type": {  
      "measurement": {}  
    }  
  }  
}
```

Memperbarui nilai atribut

Aset mewarisi atribut model aset mereka, termasuk nilai default atribut. Dalam beberapa kasus, Anda akan ingin menyimpan atribut default model aset, seperti untuk properti produsen aset. Dalam kasus lain, Anda akan ingin memperbarui atribut yang diwariskan, seperti untuk lintang dan bujur aset.

Updating an attribute value (console)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk memperbarui nilai properti aset atribut.

Untuk memperbarui nilai atribut (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.

2. Di panel navigasi, pilih Aset.
3. Pilih aset yang ingin Anda perbarui atributnya.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih Edit.
5. Temukan atribut yang akan diperbarui, lalu masukkan nilai barunya.

6. Pilih Simpan.

Updating an attribute value (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk memperbarui nilai atribut.

Anda harus mengetahui aset `assetId` dan properti Anda `propertyId` untuk menyelesaikan prosedur ini. Anda juga dapat menggunakan ID eksternal. Jika Anda membuat aset dan tidak mengetahuinya `assetId`, gunakan [ListAssets](#) API untuk mencantumkan semua aset untuk model tertentu. Gunakan [DescribeAsset](#) operasi untuk melihat properti aset Anda termasuk ID properti.

Gunakan [BatchPutAssetPropertyValue](#) operasi untuk menetapkan nilai atribut ke aset Anda. Anda dapat menggunakan operasi ini untuk mengatur beberapa atribut sekaligus. Payload operasi ini berisi daftar entri, dan setiap entri berisi ID aset, ID properti, dan nilai atribut.

Untuk memperbarui nilai atribut (AWS CLI)

1. Buat file bernama `batch-put-payload.json` dan salin objek JSON berikut ke dalam file. Contoh payload ini menunjukkan cara mengatur lintang dan bujur turbin angin. Perbarui ID, nilai, dan stempel waktu untuk mengubah payload untuk kasus penggunaan Anda.


```
{
  "entries": [
    {
      "entryId": "windfarm3-turbine7-latitude",
      "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "propertyId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "propertyValues": [
        {
          "value": {
            "doubleValue": 47.6204
          },
          "timestamp": {
            "timeInSeconds": 1575691200
          }
        }
      ]
    },
    {
      "entryId": "windfarm3-turbine7-longitude",
      "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "propertyId": "a1b2c3d4-5678-90ab-cdef-55555EXAMPLE",
      "propertyValues": [
        {
          "value": {
            "doubleValue": 122.3491
          },
          "timestamp": {
            "timeInSeconds": 1575691200
          }
        }
      ]
    }
  ]
}
```

- Setiap entri dalam payload berisi `entryId` yang dapat Anda definisikan sebagai string unik apa pun. Jika ada entri permintaan yang gagal, setiap kesalahan akan berisi permintaan `entryId` yang sesuai sehingga Anda tahu permintaan mana yang harus dicoba lagi.
- Untuk menetapkan nilai atribut, Anda dapat menyertakan satu `timestamp-quality-value` (TQV) struktur dalam daftar `propertyValues` untuk setiap properti atribut. Struktur ini harus berisi yang baru `value` dan saat `initimestamp`.

- `value`— Struktur yang berisi salah satu bidang berikut, tergantung pada jenis properti yang ditetapkan:
 - `booleanValue`
 - `doubleValue`
 - `integerValue`
 - `stringValue`
- `timestamp`— Struktur yang berisi waktu epoch Unix saat ini dalam hitungan detik, `timeInSeconds` AWS IoT SiteWise menolak setiap titik data dengan stempel waktu yang ada lebih dari 7 hari di masa lalu atau lebih baru dari 5 menit di masa depan.

Untuk informasi lebih lanjut tentang cara menyiapkan muatan [BatchPutAssetPropertyValue](#), lihat [Menyerap data menggunakan API AWS IoT SiteWise](#).

2. Jalankan perintah berikut untuk mengirim nilai atribut ke AWS IoT SiteWise:

```
aws iotsitewise batch-put-asset-property-value -\-cli-input-json file://batch-put-payload.json
```

Mengasosiasikan dan memisahkan aset

Jika model aset Anda menentukan hierarki model aset anak, Anda dapat mengaitkan aset anak ke aset Anda. Aset induk dapat mengakses dan mengumpulkan data dari aset terkait. Untuk informasi selengkapnya tentang model aset hierarkis, lihat [Mendefinisikan hierarki model aset](#).

Topik

- [Mengasosiasikan dan memisahkan aset \(konsol\)](#)
- [Mengasosiasikan dan memisahkan aset \(\)AWS CLI](#)

Mengasosiasikan dan memisahkan aset (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk mengaitkan dan memisahkan aset.

Untuk mengaitkan aset (konsol)

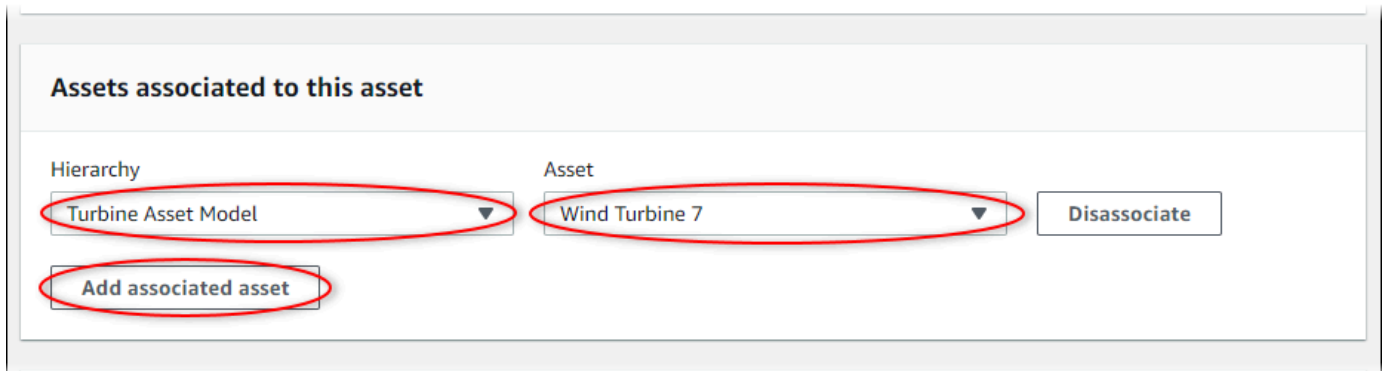
1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.

2. Di panel navigasi, pilih Aset.
3. Pilih aset induk yang ingin Anda kaitkan dengan aset anak.

i Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih Edit.
5. Di Aset yang terkait dengan aset ini, pilih Tambahkan aset terkait.



6. Untuk Hierarki, pilih hierarki yang mendefinisikan hubungan antara aset induk dan aset anak.
7. Untuk Aset, pilih aset anak untuk diasosiasikan.
8. Pilih Simpan.

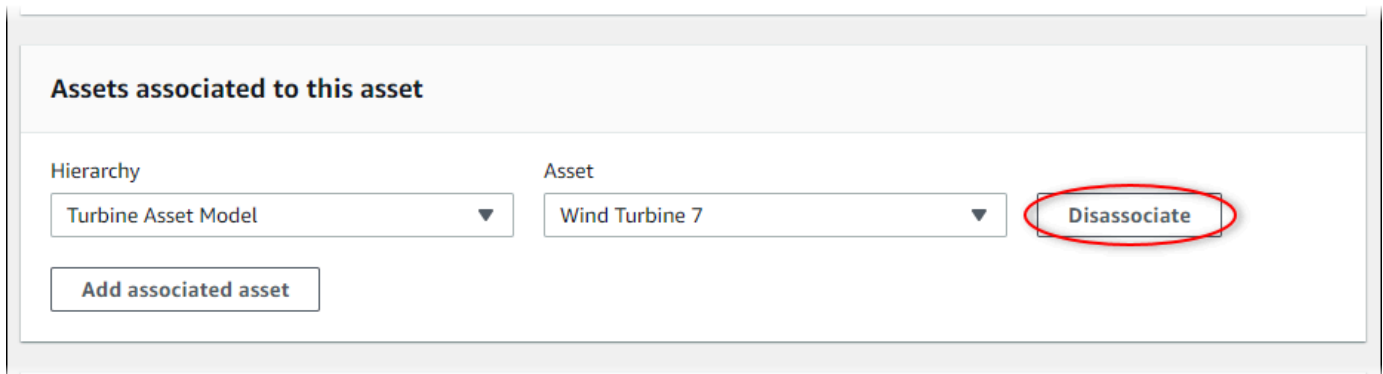
Untuk memisahkan aset (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset induk yang ingin Anda pisahkan aset anak.

i Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih Edit.
5. Dalam Aset yang terkait dengan aset ini, pilih Disassociate untuk aset tersebut.



6. Pilih Simpan.

Mengasosiasikan dan memisahkan aset (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk mengaitkan dan memisahkan aset.

Untuk prosedur ini, Anda harus mengetahui ID hierarki (`hierarchyId`) dalam model aset induk yang mendefinisikan hubungan dengan model aset anak. Gunakan [DescribeAsset](#) operasi untuk menemukan ID hierarki dalam respons.

Untuk menemukan ID hierarki

- Jalankan perintah berikut untuk menggambarkan aset induk. Ganti `parent-asset-id` dengan ID aset induk atau ID eksternal.

```
aws iotsitewise describe-asset --asset-id parent-asset-id
```

Operasi mengembalikan respons yang berisi detail aset. Respons berisi `assetHierarchies` daftar yang memiliki struktur berikut:

```
{
  ...
  "assetHierarchies": [
    {
      "id": "String",
      "name": "String"
    }
  ],
  ...
}
```

```
}
```

ID hierarki adalah id nilai untuk hierarki dalam daftar hierarki aset.

Setelah Anda memiliki ID hierarki, Anda dapat mengaitkan atau memisahkan aset dengan hierarki tersebut.

Untuk mengaitkan aset anak dengan aset induk, gunakan [AssociateAssets](#) operasi. Untuk memisahkan aset anak dari aset induk, gunakan [DisassociateAssets](#) operasi. Tentukan parameter berikut, yang sama untuk kedua operasi:

- `assetId`— ID aset induk atau ID eksternal.
- `hierarchyId`— ID hierarki atau ID eksternal dalam aset induk.
- `childAssetId`— ID aset anak atau ID eksternal.

Untuk mengaitkan aset (AWS CLI)

- Jalankan perintah berikut untuk mengaitkan aset anak ke aset induk. Ganti *parent-asset-id*, *hierarki-id*, dan *child-asset-id* dengan ID masing-masing:

```
aws iotsitewise associate-assets \  
  --asset-id parent-asset-id \  
  --hierarchy-id hierarki-id \  
  --child-asset-id child-asset-id
```

Untuk memisahkan aset (AWS CLI)

- Jalankan perintah berikut untuk memisahkan aset anak dari aset induk. Ganti *parent-asset-id*, *hierarki-id*, dan *child-asset-id* dengan ID masing-masing:

```
aws iotsitewise disassociate-assets \  
  --asset-id parent-asset-id \  
  --hierarchy-id hierarki-id \  
  --child-asset-id child-asset-id
```

Memperbarui aset dan model

Anda dapat memperbarui aset, model aset, dan model komponen AWS IoT SiteWise untuk mengubah nama dan definisinya. Operasi pembaruan ini asinkron dan membutuhkan waktu untuk menyebar. AWS IoT SiteWise Periksa status aset atau model sebelum Anda membuat perubahan tambahan. Anda harus menunggu hingga perubahan menyebar sebelum Anda dapat terus menggunakan aset atau model yang diperbarui.

Topik

- [Memperbarui aset](#)
- [Memperbarui model aset dan model komponen](#)
- [Memperbarui model komposit khusus \(Komponen\)](#)

Memperbarui aset

Anda dapat menggunakan AWS IoT SiteWise konsol atau API untuk memperbarui nama aset.

Saat Anda memperbarui aset, status aset adalah UPDATING hingga perubahan menyebar. Untuk informasi selengkapnya, lihat [Status aset dan model](#).

Topik

- [Memperbarui aset \(konsol\)](#)
- [Memperbarui aset \(AWS CLI\)](#)

Memperbarui aset (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk memperbarui detail aset.

Untuk memperbarui aset (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang akan diperbarui.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih Edit.
5. Perbarui Nama aset.
6. (Opsional) Di halaman ini, perbarui informasi lain untuk aset tersebut. Untuk informasi selengkapnya, lihat hal berikut:
 - [Memetakan aliran data industri ke properti aset](#)
 - [Memperbarui nilai atribut](#)
 - [Berinteraksi dengan layanan lain AWS](#)
7. Pilih Simpan.

Memperbarui aset (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk memperbarui nama aset.

Gunakan [UpdateAsset](#) operasi untuk memperbarui aset. Tentukan parameter berikut:

- `assetId`— ID aset. Ini adalah ID aktual dalam format UUID, atau `externalId:myExternalId` jika memilikinya. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.
- `assetName` Nama baru aset tersebut.

Untuk memperbarui nama aset (AWS CLI)

- Jalankan perintah berikut untuk memperbarui nama aset. Ganti `asset-id` dengan ID atau ID eksternal aset. Perbarui nama *aset dengan nama* baru untuk aset tersebut.

```
aws iotsitewise update-asset \  
  --asset-id asset-id \  
  --asset-name asset-name
```

Memperbarui model aset dan model komponen

Anda dapat menggunakan AWS IoT SiteWise konsol atau API untuk memperbarui model aset atau model komponen.

Anda tidak dapat mengubah tipe atau tipe data properti yang ada, atau jendela metrik yang ada. Anda juga tidak dapat mengubah jenis model dari model aset ke model komponen, atau sebaliknya.

Important

- Jika Anda menghapus properti dari model aset atau model komponen, AWS IoT SiteWise menghapus semua data sebelumnya untuk properti tersebut. Untuk model komponen, ini memengaruhi semua model aset yang menggunakan model komponen tersebut, jadi berhati-hatilah untuk memahami seberapa luas perubahan Anda dapat diterapkan.
- Jika Anda menghapus definisi hierarki dari model aset, AWS IoT SiteWise lepaskan semua aset dalam hierarki tersebut.

Saat Anda memperbarui model aset, setiap aset berdasarkan model tersebut mencerminkan perubahan apa pun yang Anda buat pada model yang mendasarinya. Sampai perubahan menyebar, setiap aset memiliki UPDATING negara. Anda harus menunggu sampai aset tersebut kembali ke ACTIVE negara sebelum Anda berinteraksi dengan mereka. Selama waktu ini, status model aset yang diperbarui akan menjadi PROPAGATING.

Saat Anda memperbarui model komponen, setiap model aset yang menggabungkan model komponen tersebut mencerminkan perubahan. Sampai perubahan model komponen menyebar, setiap model aset yang terpengaruh memiliki UPDATING status, diikuti oleh PROPAGATING saat memperbarui aset terkait, seperti yang dijelaskan dalam paragraf sebelumnya. Anda harus menunggu sampai model aset tersebut kembali ke ACTIVE keadaan sebelum Anda berinteraksi dengannya. Selama waktu ini, status model komponen yang diperbarui akan menjadi PROPAGATING.

Untuk informasi selengkapnya, lihat [Status aset dan model](#).

Topik

- [Memperbarui model aset atau komponen \(konsol\)](#)
- [Memperbarui aset atau model komponen \(AWS CLI\)](#)

Memperbarui model aset atau komponen (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk memperbarui model aset atau model komponen.

Untuk memperbarui model aset atau model komponen (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model.
3. Pilih model aset atau model komponen yang akan diperbarui.
4. Pilih Edit.
5. Pada halaman Edit model, lakukan salah satu hal berikut:
 - Dalam detail Model, ubah Nama model.
 - Ubah salah satu definisi Atribut. Anda tidak dapat mengubah tipe Data atribut yang ada. Untuk informasi selengkapnya, lihat [Mendefinisikan data statis \(atribut\)](#).
 - Ubah salah satu definisi Pengukuran. Anda tidak dapat mengubah tipe Data pengukuran yang ada. Untuk informasi selengkapnya, lihat [Mendefinisikan aliran data dari peralatan \(pengukuran\)](#).
 - Ubah salah satu definisi Transform. Untuk informasi selengkapnya, lihat [Mengubah data \(transformasi\)](#).
 - Ubah salah satu definisi Metrik. Anda tidak dapat mengubah Interval waktu metrik yang ada. Untuk informasi selengkapnya, lihat [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#).
 - (Hanya model aset) Ubah salah satu definisi Hierarki. Anda tidak dapat mengubah model Hierarki hierarki yang ada. Untuk informasi selengkapnya, lihat [Mendefinisikan hierarki model aset](#).
6. Pilih Simpan.

Memperbarui aset atau model komponen (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk memperbarui model aset atau model komponen.

Gunakan [UpdateAssetModelAPI](#) untuk memperbarui nama, deskripsi, dan properti model aset atau model komponen. Hanya untuk model aset, Anda dapat memperbarui hierarki. Tentukan parameter berikut:

- `assetModelId`— ID aset. Ini adalah ID aktual dalam format UUID, atau `externalId:myExternalId` jika memilikinya. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

Tentukan model yang diperbarui di payload. Untuk mempelajari tentang format yang diharapkan dari model aset atau model komponen, lihat [Membuat model aset](#).

Warning

[UpdateAssetModel](#) API menimpa model yang ada dengan model yang Anda berikan di payload. Untuk menghindari penghapusan properti atau hierarki model Anda, Anda harus menyertakan ID dan definisinya dalam payload model yang diperbarui. Untuk mempelajari cara menanyakan struktur model yang ada, lihat [DescribeAssetModel](#) operasinya.

Note

Prosedur berikut hanya dapat memperbarui model komposit tipe `AWS/ALARM`. Jika Anda ingin memperbarui model `CUSTOM` komposit, gunakan [UpdateAssetModelCompositeModel](#) sebagai gantinya. Untuk informasi selengkapnya, lihat [Memperbarui model komposit khusus \(Komponen\)](#).

Untuk memperbarui model aset atau model komponen (AWS CLI)

1. Jalankan perintah berikut untuk mengambil definisi model yang ada. Ganti `asset-model-id` dengan ID atau ID eksternal model aset atau model komponen untuk diperbarui.

```
aws iotsitewise describe-asset-model --asset-model-id asset-model-id
```

Operasi mengembalikan respons yang berisi detail model. Responsnya memiliki struktur sebagai berikut.

```
{
  "assetModelId": "String",
  "assetModelArn": "String",
  "assetModelName": "String",
  "assetModelDescription": "String",
```

```

"assetModelProperties": Array of AssetModelProperty,
"assetModelHierarchies": Array of AssetModelHierarchyDefinition,
"assetModelCompositeModels": Array of AssetModelCompositeModel,
"assetModelCompositeModelSummaries": Array of AssetModelCompositeModelSummary,
"assetModelCreationDate": "String",
"assetModelLastUpdateDate": "String",
"assetModelStatus": {
  "state": "String",
  "error": {
    "code": "String",
    "message": "String"
  },
"assetModelType": "String"
}
}

```

Untuk informasi lebih lanjut, lihat [DescribeAssetModel](#) operasi.

2. Buat file bernama `update-asset-model.json` dan salin respons perintah sebelumnya ke dalam file.
3. Hapus pasangan nilai kunci berikut dari objek JSON di `update-asset-model.json`:
 - `assetModelId`
 - `assetModelArn`
 - `assetModelCompositeModelSummaries`
 - `assetModelCreationDate`
 - `assetModelLastUpdateDate`
 - `assetModelStatus`
 - `assetModelType`

[UpdateAssetModel](#) Operasi mengharapkan muatan dengan struktur berikut:

```

{
  "assetModelName": "String",
  "assetModelDescription": "String",
  "assetModelProperties": Array of AssetModelProperty,
  "assetModelHierarchies": Array of AssetModelHierarchyDefinition,
  "assetModelCompositeModels": Array of AssetModelCompositeModel
}

```

4. Pada `update-asset-model.json`, lakukan langkah-langkah berikut:
 - Ubah nama model aset (`assetModelName`).
 - Mengubah, menambah, atau menghapus deskripsi model aset (`assetModelDescription`).
 - Mengubah, menambah, atau menghapus properti model aset (`assetModelProperties`). Anda tidak dapat mengubah properti `dataType` yang ada atau metrik `window` yang ada. Untuk informasi selengkapnya, lihat [Mendefinisikan properti data](#).
 - Mengubah, menambah, atau menghapus salah satu hierarki model aset (`assetModelHierarchies`). Anda tidak dapat mengubah hierarki `childAssetModelId` yang ada. Untuk informasi selengkapnya, lihat [Mendefinisikan hierarki model aset](#).
 - Mengubah, menambah, atau menghapus salah satu model komposit model aset tipe AWS/ALARM (`assetModelCompositeModels`). Alarm memantau properti lain sehingga Anda dapat mengidentifikasi kapan peralatan atau proses memerlukan perhatian. Setiap definisi alarm adalah model komposit yang menstandarisasi serangkaian properti yang digunakan alarm. Lihat informasi yang lebih lengkap di [Memantau data dengan alarm](#) dan [Mendefinisikan alarm pada model aset](#).
5. Jalankan perintah berikut untuk memperbarui model aset dengan definisi yang disimpan di `update-asset-model.json`. Ganti `asset-model-id` dengan ID model aset:

```
aws iotsitewise update-asset-model \  
  --asset-model-id asset-model-id \  
  --cli-input-json file://model-payload.json
```

Memperbarui model komposit khusus (Komponen)

Anda dapat menggunakan AWS IoT SiteWise API untuk memperbarui model komposit khusus, atau AWS IoT SiteWise konsol untuk memperbarui komponen.

Topik

- [Memperbarui komponen \(konsol\)](#)
- [Memperbarui model komposit khusus \(AWS CLI\)](#)

Memperbarui komponen (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk memperbarui komponen.

Untuk memperbarui komponen (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model.
3. Pilih model aset di mana komponennya berada.
4. Pada tab Properties, pilih Components.
5. Pilih komponen yang ingin Anda perbarui.
6. Pilih Edit.
7. Pada halaman Edit komponen, lakukan salah satu hal berikut:
 - Dalam detail Model, ubah Nama model.
 - Ubah salah satu definisi Atribut. Anda tidak dapat mengubah tipe Data atribut yang ada. Untuk informasi selengkapnya, lihat [Mendefinisikan data statis \(atribut\)](#).
 - Ubah salah satu definisi Pengukuran. Anda tidak dapat mengubah tipe Data pengukuran yang ada. Untuk informasi selengkapnya, lihat [Mendefinisikan aliran data dari peralatan \(pengukuran\)](#).
 - Ubah salah satu definisi Transform. Untuk informasi selengkapnya, lihat [Mengubah data \(transformasi\)](#).
 - Ubah salah satu definisi Metrik. Anda tidak dapat mengubah Interval waktu metrik yang ada. Untuk informasi selengkapnya, lihat [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#).
8. Pilih Simpan.

Memperbarui model komposit khusus (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk memperbarui model komposit kustom.

Untuk memperbarui nama atau deskripsi, gunakan [UpdateAssetModelCompositeModel](#) operasi. Hanya untuk model komposit khusus sebaris, Anda juga dapat memperbarui properti. Anda tidak dapat memperbarui properti model komposit component-model-based kustom, karena model komponen yang direferensikan menyediakan properti terkait.

⚠ Important

Jika Anda menghapus properti dari model komposit kustom, AWS IoT SiteWise menghapus semua data sebelumnya untuk properti tersebut. Anda tidak dapat mengubah tipe atau tipe data properti yang ada.

Untuk mengganti properti model komposit yang ada dengan yang baru dengan yang samaname, lakukan hal berikut:

1. Kirim `UpdateAssetModelCompositeModel` permintaan dengan seluruh properti yang ada dihapus.
2. Kirim `UpdateAssetModelCompositeModel` permintaan kedua yang mencakup properti baru. Properti aset baru akan name sama dengan yang sebelumnya dan AWS IoT SiteWise akan menghasilkan unik baruid.

Untuk memperbarui model komposit kustom (AWS CLI)

1. Untuk mengambil definisi model komposit yang ada, jalankan perintah berikut. Ganti `composite-model-id` dengan ID atau ID eksternal model komposit khusus untuk diperbarui, dan `asset-model-id` dengan model aset yang terkait dengan model komposit khusus. Untuk informasi selengkapnya, lihat di Panduan AWS IoT SiteWise Pengguna.

```
aws iotsitewise describe-asset-model-composite-model \  
--asset-model-composite-model-id composite-model-id \  
--asset-model-id asset-model-id
```

Untuk informasi lebih lanjut, lihat [DescribeAssetModelCompositeModel](#) operasi.

2. Buat file bernama `update-custom-composite-model.json`, lalu salin respons perintah sebelumnya ke dalam file.
3. Hapus setiap pasangan kunci-nilai dari objek JSON `update-custom-composite-model.json` kecuali untuk bidang berikut:
 - `assetModelCompositeModelName`
 - `assetModelCompositeModelDescription`(jika ada)
 - `assetModelCompositeModelProperties`(jika ada)
4. Pada `update-custom-composite-model.json`, lakukan langkah-langkah berikut:

- Ubah nilai `assetModelCompositeModelName`.
- Tambah atau hapus `assetModelCompositeModelDescription`, atau ubah nilainya.
- Hanya untuk model komposit kustom sebaris: Ubah, tambahkan, atau hapus properti model aset apa pun di `assetModelCompositeModelProperties`.

Untuk informasi selengkapnya tentang format yang diperlukan untuk file ini, lihat sintaks permintaan untuk [UpdateAssetModelCompositeModel](#).

5. Jalankan perintah berikut untuk memperbarui model komposit khusus dengan definisi yang disimpan di `update-custom-composite-model.json`. Ganti *composite-model-id* dengan ID model komposit, dan *asset-model-id* dengan ID model aset yang ada di dalamnya.

```
aws iotsitewise update-asset-model-composite-model \  
--asset-model-composite-model-id composite-model-id \  
--asset-model-id asset-model-id \  
--cli-input-json file://update-custom-composite-model.json
```

Menghapus aset dan model

Anda dapat menghapus aset dan model dari AWS IoT SiteWise saat Anda selesai menggunakannya. Operasi penghapusan tidak sinkron dan membutuhkan waktu untuk menyebar. AWS IoT SiteWise

Topik

- [Menghapus aset](#)
- [Menghapus model aset](#)

Menghapus aset

Anda dapat menggunakan AWS IoT SiteWise konsol atau API untuk menghapus aset.

Sebelum Anda dapat menghapus aset, Anda harus terlebih dahulu memisahkan aset turunannya dan memisahkannya dari aset induknya. Untuk informasi selengkapnya, lihat [Mengasosiasikan dan memisahkan aset](#). Jika Anda menggunakan AWS Command Line Interface (AWS CLI), Anda dapat menggunakan [ListAssociatedAssets](#) operasi untuk membuat daftar anak-anak aset.

Saat Anda menghapus aset, statusnya DELETING sampai perubahan menyebar. Untuk informasi selengkapnya, lihat [Status aset dan model](#). Setelah aset dihapus, Anda tidak dapat menanyakan aset tersebut. Jika Anda melakukannya, API mengembalikan respons HTTP 404.

 Important

AWS IoT SiteWise menghapus semua data properti untuk aset yang dihapus.

Topik

- [Menghapus aset \(konsol\)](#)
- [Menghapus aset \(\)AWS CLI](#)

Menghapus aset (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk menghapus aset.

Untuk menghapus aset (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang akan dihapus.

 Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Jika aset memiliki aset Terkait, hapus setiap aset. Anda dapat memilih nama aset untuk menavigasi ke halamannya, tempat Anda dapat menghapusnya.
5. Pada halaman aset, pilih Hapus.
6. Di kotak dialog Hapus aset, lakukan hal berikut:
 - a. Masukkan **Delete** untuk mengonfirmasi penghapusan.
 - b. Pilih Hapus.

Menghapus aset ()AWS CLI

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk menghapus aset.

Gunakan [DeleteAsset](#) operasi untuk menghapus aset. Tentukan parameter berikut:

- `assetId`— ID aset. Ini adalah ID aktual dalam format UUID, atau `externalId:myExternalId` jika memilikinya. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

Untuk menghapus aset (AWS CLI)

1. Jalankan perintah berikut untuk membuat daftar hierarki aset. Ganti `asset-id` dengan ID atau ID eksternal aset:

```
aws iotsitewise describe-asset --asset-id asset-id
```

Operasi mengembalikan respons yang berisi detail aset. Respons berisi `assetHierarchies` daftar yang memiliki struktur berikut:

```
{
  ...
  "assetHierarchies": [
    {
      "id": "String",
      "name": "String"
    }
  ],
  ...
}
```

Untuk informasi lebih lanjut, lihat [DescribeAsset](#) operasi.

2. Untuk setiap hierarki, jalankan perintah berikut untuk mencantumkan turunan aset yang terkait dengan hierarki tersebut. Ganti `asset-id` dengan ID atau ID eksternal aset dan `hierarki-id` dengan ID atau ID eksternal hierarki.

```
aws iotsitewise list-associated-assets \
  --asset-id asset-id \
  --hierarchy-id hierarchy-id
```

Untuk informasi lebih lanjut, lihat [ListAssociatedAssets](#) operasi.

3. Jalankan perintah berikut untuk menghapus setiap aset terkait dan kemudian menghapus aset tersebut. Ganti *asset-id* dengan ID atau ID eksternal aset.

```
aws iotsitewise delete-asset --asset-id asset-id
```

Menghapus model aset

Anda dapat menggunakan AWS IoT SiteWise konsol atau API untuk menghapus model aset.

Sebelum Anda dapat menghapus model aset, Anda harus terlebih dahulu menghapus semua aset yang dibuat dari model aset.

Saat Anda menghapus model aset, statusnya DELETING sampai perubahan menyebar. Untuk informasi selengkapnya, lihat [Status aset dan model](#). Setelah model aset dihapus, Anda tidak dapat menanyakan model aset tersebut. Jika Anda melakukannya, API mengembalikan respons HTTP 404.

Topik

- [Menghapus model aset \(konsol\)](#)
- [Menghapus model aset \(AWS CLI\)](#)

Menghapus model aset (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk menghapus model aset.

Untuk menghapus model aset (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model.
3. Pilih model aset yang akan dihapus.
4. Jika model memiliki Aset, hapus setiap aset. Pilih nama aset untuk menavigasi ke halamannya, tempat Anda dapat menghapusnya. Untuk informasi selengkapnya, lihat [Menghapus aset \(konsol\)](#).
5. Pada halaman model, pilih Hapus.
6. Dalam kotak dialog Delete model, lakukan hal berikut:

- a. Masukkan **Delete** untuk mengonfirmasi penghapusan.
- b. Pilih Hapus.

Menghapus model aset ()AWS CLI

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk menghapus model aset.

Gunakan [DeleteAssetModel](#) operasi untuk menghapus model aset. Tentukan parameter berikut:

- `assetModelId`— ID aset. Ini adalah ID aktual dalam format UUID, atau `externalId:myExternalId` jika memilikinya. Untuk informasi selengkapnya, lihat [Mereferensikan objek dengan ID eksternal](#) di AWS IoT SiteWise Panduan Pengguna.

Untuk menghapus model aset (AWS CLI)

1. Jalankan perintah berikut untuk mencantumkan semua aset yang dibuat dari model. Ganti *asset-model-id* dengan ID atau ID eksternal model aset.

```
aws iotsitewise list-assets --asset-model-id asset-model-id
```

Untuk informasi lebih lanjut, lihat [ListAssets](#) operasi.

2. Jika perintah sebelumnya mengembalikan aset apa pun dari model, hapus setiap aset. Untuk informasi selengkapnya, lihat [Menghapus aset \(\)AWS CLI](#).
3. Jalankan perintah berikut untuk menghapus model aset. Ganti *asset-model-id* dengan ID atau ID eksternal model aset.

```
aws iotsitewise delete-asset-model --asset-model-id asset-model-id
```

Operasi massal dengan aset dan model

Untuk bekerja dengan sejumlah besar aset atau model aset, gunakan operasi massal untuk mengimpor dan mengekspor sumber daya secara massal ke lokasi yang berbeda. Misalnya, Anda dapat membuat file data yang mendefinisikan aset atau model aset di bucket Amazon S3, dan menggunakan impor massal untuk membuat atau memperbaruinya. AWS IoT SiteWise Atau, jika

Anda memiliki sejumlah besar aset atau model aset AWS IoT SiteWise, Anda dapat mengekspornya ke Amazon S3.

Note

Anda melakukan operasi massal AWS IoT SiteWise dengan memanggil operasi di AWS IoT TwinMaker API. Anda dapat melakukan ini tanpa menyiapkan AWS IoT TwinMaker atau membuat AWS IoT TwinMaker ruang kerja. Yang Anda butuhkan hanyalah ember Amazon S3 tempat Anda dapat menempatkan konten Anda AWS IoT SiteWise .

Topik

- [Konsep dan terminologi kunci](#)
- [Fungsionalitas yang didukung](#)
- [Prasyarat operasi massal](#)
- [Menjalankan pekerjaan impor massal](#)
- [Menjalankan pekerjaan ekspor massal](#)
- [Pelacakan kemajuan pekerjaan dan penanganan kesalahan](#)
- [Impor contoh metadata](#)
- [Contoh metadata ekspor](#)
- [AWS IoT SiteWise skema pekerjaan transfer metadata](#)

Konsep dan terminologi kunci

AWS IoT SiteWise fitur impor dan ekspor massal bergantung pada konsep dan terminologi berikut:

- **Impor:** Tindakan memindahkan aset atau model aset dari file di bucket Amazon S3 ke file. AWS IoT SiteWise
- **Ekspor:** Tindakan memindahkan aset atau model aset dari AWS IoT SiteWise ke bucket Amazon S3.
- **Sumber:** Lokasi awal dari mana Anda ingin memindahkan konten.

Misalnya, bucket Amazon S3 adalah sumber impor, dan AWS IoT SiteWise merupakan sumber ekspor.

- Tujuan: Lokasi yang diinginkan di mana Anda ingin memindahkan konten Anda.

Misalnya, bucket Amazon S3 adalah tujuan ekspor, dan AWS IoT SiteWise merupakan tujuan impor.

- AWS IoT SiteWise Skema: Skema ini digunakan untuk mengimpor dan mengekspor metadata dari AWS IoT SiteWise
- Sumber daya tingkat atas: AWS IoT SiteWise Sumber daya yang dapat Anda buat atau perbarui secara individual, seperti model aset atau aset.
- Sub-sumber daya: Sumber AWS IoT SiteWise daya bersarang dalam sumber daya tingkat atas. Contohnya termasuk properti, hierarki, dan model komposit.
- Metadata: Informasi penting yang diperlukan untuk mengimpor atau mengekspor sumber daya dengan sukses. Contoh metadata adalah definisi aset dan model aset.
- metadataTransferJob: Objek yang dibuat saat Anda menjalankan `CreateMetadataTransferJob`.

Fungsionalitas yang didukung

Topik ini menjelaskan apa yang dapat Anda lakukan ketika Anda menjalankan operasi massal.

Operasi massal mendukung fungsionalitas berikut:

- Pembuatan sumber daya tingkat atas: Saat Anda mengimpor aset atau model aset yang tidak menentukan ID, atau ID yang tidak cocok dengan ID yang sudah ada, maka ID tersebut akan dibuat sebagai sumber daya baru.
- Penggantian sumber daya tingkat atas: Ketika Anda mengimpor aset atau model aset yang ID-nya cocok dengan yang sudah ada, maka itu akan menggantikan sumber daya yang ada.
- Pembuatan, penggantian, atau penghapusan sub-sumber daya: Ketika impor Anda menggantikan sumber daya tingkat atas seperti aset atau model aset, definisi baru akan menggantikan semua sub-sumber daya, seperti properti, hierarki, atau model komposit.

Misalnya, jika Anda memperbarui model aset selama impor massal, dan versi yang diperbarui mendefinisikan properti yang tidak ada pada aslinya, maka properti baru akan dibuat. Jika mendefinisikan properti yang sudah ada, maka properti yang ada akan diperbarui. Jika model aset yang diperbarui menghilangkan properti yang ada pada aslinya, maka properti tersebut akan dihapus.

- Tidak ada penghapusan sumber daya tingkat atas: Operasi massal tidak menghapus aset atau model aset. Operasi massal hanya membuat atau memperbaruinya.

Prasyarat operasi massal

Bagian ini menjelaskan prasyarat operasi massal, termasuk izin AWS Identity and Access Management (IAM) untuk bertukar sumber daya antara dan mesin lokal Anda. Layanan AWS Sebelum Anda memulai operasi massal, selesaikan prasyarat berikut:

- Buat bucket Amazon S3 untuk menyimpan sumber daya. Untuk informasi selengkapnya tentang menggunakan Amazon S3, lihat [Apa itu Amazon S3?](#)

Izin IAM

Untuk melakukan operasi massal, Anda harus membuat kebijakan AWS Identity and Access Management (IAM) dengan izin yang memungkinkan pertukaran AWS sumber daya antara Amazon S3 AWS IoT SiteWise, dan mesin lokal Anda. Untuk informasi selengkapnya tentang membuat kebijakan IAM, lihat [Membuat kebijakan IAM](#).

Untuk melakukan operasi massal, Anda memerlukan kebijakan berikut.

AWS IoT SiteWise kebijakan

Kebijakan ini memungkinkan akses ke tindakan AWS IoT SiteWise API yang diperlukan untuk operasi massal:

```
{
  "Sid": "SiteWiseApiAccess",
  "Effect": "Allow",
  "Action": [
    "iotsitewise:CreateAsset",
    "iotsitewise:CreateAssetModel",
    "iotsitewise:UpdateAsset",
    "iotsitewise:UpdateAssetModel",
    "iotsitewise:UpdateAssetProperty",
    "iotsitewise:ListAssets",
    "iotsitewise:ListAssetModels",
    "iotsitewise:ListAssetProperties",
    "iotsitewise:ListAssetModelProperties",
    "iotsitewise:ListAssociatedAssets",
    "iotsitewise:DescribeAsset",
    "iotsitewise:DescribeAssetModel",
    "iotsitewise:DescribeAssetProperty",
    "iotsitewise:AssociateAssets",
```

```

    "iotsitewise:DisassociateAssets",
    "iotsitewise:AssociateTimeSeriesToAssetProperty",
    "iotsitewise:DisassociateTimeSeriesFromAssetProperty",
    "iotsitewise:BatchPutAssetPropertyValue",
    "iotsitewise:BatchGetAssetPropertyValue",
    "iotsitewise:TagResource",
    "iotsitewise:UntagResource",
    "iotsitewise:ListTagsForResource",
    "iotsitewise>CreateAssetModelCompositeModel",
    "iotsitewise:UpdateAssetModelCompositeModel",
    "iotsitewise:DescribeAssetModelCompositeModel",
    "iotsitewise>DeleteAssetModelCompositeModel",
    "iotsitewise>ListAssetModelCompositeModels",
    "iotsitewise>ListCompositionRelationships",
    "iotsitewise:DescribeAssetCompositeModel"
  ],
  "Resource": "*"
}

```

AWS IoT TwinMaker kebijakan

Kebijakan ini memungkinkan akses ke operasi AWS IoT TwinMaker API yang Anda gunakan untuk bekerja dengan operasi massal:

```

{
  "Sid": "MetadataTransferJobApiAccess",
  "Effect": "Allow",
  "Action": [
    "iottwinmaker:CreateMetadataTransferJob",
    "iottwinmaker:CancelMetadataTransferJob",
    "iottwinmaker:GetMetadataTransferJob",
    "iottwinmaker:ListMetadataTransferJobs"
  ],
  "Resource": "*"
}

```

Kebijakan Amazon S3

Kebijakan ini menyediakan akses ke bucket Amazon S3 untuk mentransfer metadata untuk operasi massal.

For a specific Amazon S3 bucket

Jika Anda menggunakan satu bucket tertentu untuk bekerja dengan metadata operasi massal, kebijakan ini menyediakan akses ke bucket tersebut:

```
{
  "Effect": "Allow",
  "Action": [
    "s3:PutObject",
    "s3:GetObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:AbortMultipartUpload",
    "s3:ListBucketMultipartUploads",
    "s3:ListMultipartUploadParts"
  ],
  "Resource": [
    "arn:aws:s3::bucket name",
    "arn:aws:s3::bucket name/*"
  ]
}
```

To allow any Amazon S3 bucket

Jika Anda akan menggunakan banyak bucket berbeda untuk bekerja dengan metadata operasi massal, kebijakan ini menyediakan akses ke bucket apa pun:

```
{
  "Effect": "Allow",
  "Action": [
    "s3:PutObject",
    "s3:GetObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:AbortMultipartUpload",
    "s3:ListBucketMultipartUploads",
    "s3:ListMultipartUploadParts"
  ],
  "Resource": "*"
}
```


Untuk informasi tentang pemecahan masalah operasi impor dan ekspor, lihat. [Memecahkan masalah impor dan ekspor massal](#)

Menjalankan pekerjaan impor massal

Impor massal adalah tindakan memindahkan metadata ke ruang kerja. AWS IoT SiteWise Misalnya, impor massal dapat memindahkan metadata dari file lokal, atau file di bucket Amazon S3, ke ruang kerja. AWS IoT SiteWise

Langkah 1: Siapkan file untuk diimpor

Unduh file format AWS IoT SiteWise asli untuk mengimpor aset dan model aset. Lihat [AWS IoT SiteWise skema pekerjaan transfer metadata](#) untuk detail selengkapnya.

Langkah 2: Unggah file yang disiapkan ke Amazon S3

Unggah file ke Amazon S3. Lihat [Mengunggah file ke Amazon S3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon untuk detailnya.

Impor metadata (konsol)

Anda dapat menggunakan metadata impor massal. Konsol AWS IoT SiteWise Ikuti [Langkah 1: Siapkan file untuk diimpor](#) dan [Langkah 2: Unggah file yang disiapkan ke Amazon S3](#) siapkan file yang siap diimpor.

Impor data dari Amazon S3 ke Konsol AWS IoT SiteWise

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Pilih Operasi massal Baru dari panel navigasi.
3. Pilih Impor baru untuk memulai proses impor.
4. Pada halaman metadata Impor:
 - Pilih Jelajahi Amazon S3 untuk melihat bucket dan file Amazon S3.
 - Arahkan ke bucket Amazon S3 yang berisi file impor yang disiapkan.
 - Pilih file yang akan diimpor.
 - Tinjau file yang dipilih, dan pilih Impor.
5. Operasi massal pada halaman SiteWise metadata Konsol AWS IoT SiteWise menampilkan pekerjaan impor yang baru dibuat dalam tabel kemajuan Pekerjaan.

Impor metadata ()AWS CLI

Untuk melakukan tindakan impor, gunakan prosedur berikut:

Impor data dari Amazon S3 ke AWS CLI

1. Buat file metadata yang menentukan sumber daya yang ingin Anda impor, mengikuti [AWS IoT SiteWise skema pekerjaan transfer metadata](#). Simpan file ini di bucket Amazon S3 Anda.

Untuk contoh file metadata yang akan diimpor, lihat [Impor contoh metadata](#).

2. Sekarang buat file JSON dengan badan permintaan. Badan permintaan menentukan sumber dan tujuan untuk pekerjaan transfer. File ini terpisah dari file dari langkah sebelumnya. Pastikan untuk menentukan bucket Amazon S3 Anda sebagai sumber dan `iotsitewise` sebagai tujuan.

Contoh berikut menunjukkan badan permintaan:

```
{
  "metadataTransferJobId": "your-transfer-job-Id",
  "sources": [{
    "type": "s3",
    "s3Configuration": {
      "location": "arn:aws:s3::your-S3-bucket-name/  
your_import_metadata.json"
    }
  }],
  "destination": {
    "type": "iotsitewise"
  }
}
```

3. Memanggil `CreateMetadataTransferJob` dengan menjalankan AWS CLI perintah berikut. Dalam contoh ini, file isi permintaan dari langkah sebelumnya diberi nama `createMetadataTransferJobExport.json`.

```
aws iottwinmaker create-metadata-transfer-job --region us-east-1 \  
--cli-input-json file://createMetadataTransferJobImport.json
```

Ini akan membuat pekerjaan transfer metadata, dan memulai proses mentransfer sumber daya yang Anda pilih.

Menjalankan pekerjaan ekspor massal

Ekspor massal adalah tindakan memindahkan metadata dari AWS IoT SiteWise ruang kerja ke bucket Amazon S3.

Saat melakukan ekspor massal AWS IoT SiteWise konten ke Amazon S3, Anda dapat menentukan filter untuk membatasi model aset dan aset tertentu yang ingin Anda ekspor.

Filter harus ditentukan di `iotSiteWiseConfiguration` bagian dalam bagian sumber permintaan JSON Anda.

Note

Anda dapat menyertakan beberapa filter dalam permintaan Anda. Operasi massal akan mengekspor model aset dan aset yang cocok dengan filter mana pun.

Jika Anda tidak memberikan filter apa pun, operasi massal mengekspor semua model aset dan aset Anda.

Example minta badan dengan filter

```
{
  "metadataTransferJobId": "your-transfer-job-id",
  "sources": [
    {
      "type": "iotsitewise",
      "iotSiteWiseConfiguration": {
        "filters": [
          {
            "filterByAssetModel": {
              "assetModelId": "asset model ID"
            }
          },
          {
            "filterByAssetModel": {
              "assetModelId": "asset model ID",
              "includeAssets": true
            }
          },
          {
            "filterByAssetModel": {
```

```
        "assetModelId": "asset model ID",
        "includeOffspring": true
    }
}
]
}
],
"destination": {
    "type": "s3",
    "s3Configuration": {
        "location": "arn:aws:s3:::your-S3-bucket-location"
    }
}
}
```

Ekspor metadata (konsol)

Prosedur berikut menjelaskan tindakan ekspor konsol:

Buat pekerjaan ekspor di Konsol AWS IoT SiteWise

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Pilih Operasi massal Baru dari panel navigasi.
3. Pilih Ekspor baru untuk memulai proses ekspor.
4. Pada halaman metadata Ekspor:
 - Masukkan nama untuk pekerjaan ekspor. Ini adalah nama yang digunakan untuk file yang diekspor di bucket Amazon S3 Anda.
 - Pilih sumber daya Anda untuk diekspor, yang menetapkan filter untuk pekerjaan:
 - Ekspor semua aset dan model aset. Gunakan filter pada aset dan model aset.
 - Aset ekspor. Filter aset Anda.
 - Pilih aset yang akan digunakan untuk filter ekspor.
 - (Opsional) Tambahkan keturunan atau model aset terkait.
 - Model aset ekspor. Filter pada model aset Anda.
 - Pilih model aset yang akan digunakan untuk filter ekspor.
 - (Opsional) Tambahkan keturunan, atau aset terkait atau keduanya.
 - Pilih Berikutnya.

- Arahkan ke bucket Amazon S3:
 - Pilih Jelajahi Amazon S3 untuk melihat bucket dan file Amazon S3.
 - Arahkan ke bucket Amazon S3 tempat file harus ditempatkan.
 - Pilih Berikutnya.
 - Tinjau pekerjaan ekspor dan pilih Ekspor.
5. Operasi massal pada halaman SiteWise metadata Konsol AWS IoT SiteWise menampilkan pekerjaan impor yang baru dibuat dalam tabel kemajuan Pekerjaan.

Untuk berbagai cara menggunakan filter saat mengekspor metadata, lihat. [Contoh metadata ekspor](#)

Metadata ekspor ()AWS CLI

Prosedur berikut menjelaskan tindakan AWS CLI ekspor:

Ekspor data dari AWS IoT SiteWise ke Amazon S3

1. Buat file JSON dengan badan permintaan Anda. Badan permintaan menentukan sumber dan tujuan untuk pekerjaan transfer. Contoh berikut menunjukkan badan permintaan contoh:

```
{
  "metadataTransferJobId": "your-transfer-job-Id",
  "sources": [{
    "type": "iotsitewise"
  }],
  "destination": {
    "type": "s3",
    "s3Configuration": {
      "location": "arn:aws:s3::your-S3-bucket-location"
    }
  }
}
```

Pastikan untuk menentukan bucket Amazon S3 Anda sebagai tujuan pekerjaan transfer metadata.

Note

Contoh ini akan mengekspor semua model aset dan aset Anda. Untuk membatasi ekspor ke model aset atau aset tertentu, Anda dapat menyertakan filter di badan permintaan Anda. Untuk informasi selengkapnya tentang menerapkan filter ekspor, lihat [Contoh metadata ekspor](#).

2. Simpan file isi permintaan Anda untuk digunakan pada langkah berikutnya. Dalam contoh ini, file dinamai `createMetadataTransferJobExport.json`.
3. Memanggil `CreateMetadataTransferJob` dengan menjalankan AWS CLI perintah berikut:

```
aws iottwinmaker create-metadata-transfer-job --region us-east-1 \  
--cli-input-json file://createMetadataTransferJobExport.json
```

Ganti file JSON input `createMetadataTransferJobExport.json` dengan nama file transfer Anda sendiri.

Pelacakan kemajuan pekerjaan dan penanganan kesalahan

Pekerjaan proses massal membutuhkan waktu untuk diproses. Setiap pekerjaan diproses dalam urutan AWS IoT SiteWise menerima permintaan. Ini diproses one-at-a-time untuk setiap akun. Ketika pekerjaan selesai, antrian berikutnya secara otomatis mulai diproses. AWS IoT SiteWise menyelesaikan pekerjaan secara asinkron dan memperbarui status masing-masing saat berlangsung. Setiap pekerjaan memiliki bidang status yang berisi status sumber daya dan pesan kesalahan, jika berlaku.

Negara dapat menjadi salah satu dari nilai berikut:

- **VALIDATING**— Memvalidasi pekerjaan termasuk format file yang dikirimkan, dan isinya.
- **PENDING**— Pekerjaan dalam antrian. Anda dapat membatalkan pekerjaan dalam keadaan ini dari AWS IoT SiteWise konsol, tetapi semua negara bagian lain akan berlanjut hingga akhir.
- **RUNNING**— Memproses pekerjaan. Ini membuat dan memperbarui sumber daya seperti yang didefinisikan oleh file impor, atau mengekspor sumber daya berdasarkan filter pekerjaan ekspor yang dipilih. Jika dibatalkan, sumber daya apa pun yang diimpor oleh pekerjaan ini tidak akan dihapus. Untuk informasi selengkapnya, lihat [Tinjau kemajuan dan detail pekerjaan \(konsol\)](#).

- CANCELLING— Pekerjaan secara aktif dibatalkan.
- ERROR Satu atau lebih sumber daya gagal diproses. Periksa laporan pekerjaan terperinci untuk informasi lebih lanjut. Untuk informasi selengkapnya, lihat [Periksa detail kesalahan \(konsol\)](#).
- COMPLETED— Job selesai tanpa kesalahan.
- CANCELLED— Pekerjaan dibatalkan dan tidak antri. Jika Anda membatalkan RUNNING pekerjaan, sumber daya yang sudah diimpor oleh pekerjaan ini pada saat pembatalan tidak dihapus. AWS IoT SiteWise

Topik

- [Pelacakan kemajuan pekerjaan](#)
- [Periksa kesalahan](#)

Pelacakan kemajuan pekerjaan

Tinjau kemajuan dan detail pekerjaan (konsol)

Lihat [Impor metadata \(konsol\)](#) atau [Ekspor metadata \(konsol\)](#) untuk memulai pekerjaan massal.

Ikhtisar kemajuan pekerjaan di AWS IoT SiteWise konsol:

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Pilih Operasi massal Baru dari panel navigasi.
3. Tabel kemajuan Pekerjaan di AWS IoT SiteWise konsol, menampilkan daftar pekerjaan operasi massal.
4. Kolom tipe Job menjelaskan apakah itu pekerjaan ekspor atau impor. Kolom Tanggal yang diimpor menampilkan tanggal pekerjaan dimulai.
5. Kolom Status menampilkan status pekerjaan. Anda dapat memilih pekerjaan untuk melihat detail tentang pekerjaan itu.
6. Pekerjaan yang dipilih menunjukkan Sukses setelah berhasil, atau daftar kegagalan jika pekerjaan gagal. Deskripsi kesalahan juga ditampilkan dengan setiap jenis sumber daya.

Ikhtisar detail pekerjaan di AWS IoT SiteWise konsol:

Tabel kemajuan Pekerjaan di AWS IoT SiteWise konsol, menampilkan daftar pekerjaan operasi massal.

1. Pilih pekerjaan untuk melihat detail lebih lanjut.
2. Untuk pekerjaan impor, `Data source ARN` mewakili lokasi Amazon S3 dari file impor.
3. Untuk pekerjaan ekspor, `Data destination ARN` mewakili lokasi Amazon S3 file setelah ekspor.
4. `StatusDanStatus reason`, memberikan rincian tambahan tentang pekerjaan saat ini. Lihat [Pelacakan kemajuan pekerjaan dan penanganan kesalahan](#) untuk detail selengkapnya.
5. `Queued position`Ini mewakili posisi pekerjaan dalam antrian proses. Pekerjaan diproses satu per satu. Posisi antrian 1, menunjukkan bahwa pekerjaan akan diproses selanjutnya.
6. Halaman detail pekerjaan juga menampilkan jumlah kemajuan pekerjaan.
 - Jenis jumlah kemajuan pekerjaan adalah:
 - i. `Total resources`— Menunjukkan jumlah total aset dalam proses transfer.
 - ii. `Succeeded`— Menunjukkan jumlah aset yang berhasil ditransfer selama proses.
 - iii. `Failed`— Menunjukkan jumlah aset yang gagal selama proses.
 - iv. `Skipped`— Menunjukkan jumlah aset yang dilewati selama proses.
7. Status pekerjaan `PENDING` atau `VALIDATING`, menampilkan semua kemajuan pekerjaan dihitung sebagai-. Ini menunjukkan bahwa jumlah kemajuan pekerjaan sedang dievaluasi.
8. Status pekerjaan `RUNNING` menampilkan `Total resources` hitungan, pekerjaan yang diajukan untuk diproses. Hitungan terperinci (`Succeeded`, `Failed`, dan `Skipped`), berlaku untuk sumber daya yang diproses. Jumlah dari hitungan rinci lebih kecil dari `Total resources` hitungan, sampai status pekerjaan adalah `COMPLETED` atau `ERROR`
9. Jika status pekerjaan adalah `COMPLETED` atau `ERROR`, `Total resources` hitungannya sama dengan jumlah hitungan rinci (`Succeeded`, `Failed`, dan `Skipped`).
10. Jika status pekerjaan adalah `ERROR`, periksa tabel Kegagalan Job untuk rincian tentang kesalahan dan kegagalan tertentu. Lihat [Periksa detail kesalahan \(konsol\)](#) untuk detail selengkapnya.

Tinjau kemajuan dan detail pekerjaan (AWS CLI)

Setelah memulai operasi massal, Anda dapat memeriksa atau memperbarui statusnya menggunakan tindakan API berikut:

- Untuk mengambil informasi tentang pekerjaan tertentu, gunakan tindakan [GetMetadataTransferJob](#) API.

Mengambil informasi dengan **GetMetadataTransferJob** API:

1. Buat dan jalankan pekerjaan transfer. Panggil GetMetadataTransferJob API.

Example AWS CLI perintah:

```
aws iottwinmaker get-metadata-transfer-job \  
  --metadata-transfer-job-id your_metadata_transfer_job_id \  
  --region your_region
```

2. GetMetadataTransferJobAPI mengembalikan MetadataTransferJobProgress objek dengan parameter berikut:
 - SucceededCount - Menunjukkan jumlah aset yang berhasil ditransfer dalam proses.
 - FailedCount - Menunjukkan jumlah aset yang gagal selama proses.
 - SkippedCount - Menunjukkan jumlah aset yang dilewati selama proses.
 - TotalCount — Menunjukkan jumlah total aset dalam proses transfer.

Parameter ini menunjukkan status kemajuan pekerjaan. Jika statusnyaRUNNING, mereka membantu melacak jumlah sumber daya yang masih akan diproses.

Jika Anda menemukan kesalahan validasi skema, atau jika failedCount lebih besar dari atau sama dengan 1, status kemajuan pekerjaan berubah menjadi. ERROR Laporan kesalahan lengkap untuk pekerjaan ditempatkan di bucket Amazon S3 Anda. Lihat [Periksa kesalahan](#) untuk detail selengkapnya.

- Untuk membuat daftar pekerjaan saat ini, gunakan tindakan [ListMetadataTransferJobs](#)API.

Gunakan file JSON untuk memfilter pekerjaan yang dikembalikan berdasarkan statusnya saat ini. Lihat prosedur berikut:

1. Untuk menentukan filter yang ingin Anda gunakan, buat file JSON AWS CLI masukan. ingin menggunakan:

```
{  
  "sourceType": "s3",  
  "destinationType": "iottwinmaker",  
  "filters": [{  
    "state": "COMPLETED"  
  }]  
}
```

```
}
```

Untuk daftar state nilai yang valid, lihat [ListMetadataTransferJobsFilter](#) di Panduan Referensi AWS IoT TwinMaker API.

- Gunakan file JSON sebagai argumen dalam AWS CLI contoh perintah berikut:

```
aws iottwinmaker list-metadata-transfer-job --region your_region \  
--cli-input-json file://ListMetadataTransferJobsExample.json
```

- Untuk membatalkan pekerjaan, gunakan tindakan [CancelMetadataTransferJob](#) API. API ini membatalkan tugas transfer metadata tertentu, tanpa memengaruhi sumber daya apa pun yang sudah diekspor atau diimpor:

```
aws iottwinmaker cancel-metadata-transfer-job \  
--region your_region \  
--metadata-transfer-job-id job-to-cancel-id
```

Periksa kesalahan

Periksa detail kesalahan (konsol)

Detail kesalahan di AWS IoT SiteWise konsol:

- Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
- Lihat tabel kemajuan Pekerjaan Konsol AWS IoT SiteWise untuk daftar pekerjaan operasi massal.
- Pilih pekerjaan untuk melihat detail pekerjaan.
- Jika status pekerjaan adalah COMPLETED atau ERROR, Total resources hitungannya sama dengan jumlah hitungan rinci (Succeeded, Failed, dan Skipped).
- Jika status pekerjaan adalah ERROR, periksa tabel Kegagalan Job untuk rincian tentang kesalahan dan kegagalan tertentu.
- Tabel kegagalan Job menampilkan konten dari laporan pekerjaan. Resource type Bidang menunjukkan lokasi kesalahan atau kegagalan, seperti berikut ini:

- Misalnya, kesalahan validasi Bulk operations template di Resource type bidang menunjukkan bahwa templat impor dan format file skema metadata tidak cocok. Untuk informasi selengkapnya, lihat [AWS IoT SiteWise skema pekerjaan transfer metadata](#).
- Gagal Asset di Resource type lapangan menunjukkan bahwa aset tidak dibuat karena konflik dengan aset lain. Lihat [Kesalahan umum](#) untuk informasi tentang kesalahan dan konflik AWS IoT SiteWise sumber daya.

Periksa detail kesalahan ()AWS CLI

Untuk menangani dan mendiagnosis kesalahan yang dihasilkan selama tugas transfer, lihat prosedur berikut tentang penggunaan tindakan GetMetadataTransferJob API:

1. Setelah membuat dan menjalankan pekerjaan transfer, hubungi [GetMetadataTransferJob](#):

```
aws iottwinmaker get-metadata-transfer-job \  
  --metadata-transfer-job-id your_metadata_transfer_job_id \  
  --region us-east-1
```

2. Setelah Anda melihat status pekerjaan beralih keCOMPLETED, Anda dapat mulai memverifikasi hasil pekerjaan.
3. Ketika Anda memanggilGetMetadataTransferJob, ia mengembalikan objek yang dipanggil [MetadataTransferJobProgress](#).

MetadataTransferJobProgress Objek berisi parameter berikut:

- FailedCount: Menunjukkan jumlah aset yang gagal selama proses transfer.
 - SkippedCount: Menunjukkan jumlah aset yang dilewati selama proses transfer.
 - SucceededCount: Menunjukkan jumlah aset yang berhasil selama proses transfer.
 - TotalCount: Menunjukkan jumlah total aset yang terlibat dalam proses transfer.
4. Selain itu, panggilan API mengembalikan elemenreportUrl, yang berisi URL presigned. Jika pekerjaan transfer Anda memiliki masalah yang perlu Anda selidiki lebih lanjut, kunjungi url ini.

Impor contoh metadata

Bagian ini menunjukkan cara membuat file metadata untuk mengimpor model aset dan aset dengan operasi impor massal tunggal.

Contoh impor massal

Anda dapat mengimpor banyak model aset dan aset dengan satu operasi impor massal. Contoh berikut menunjukkan cara membuat file metadata untuk melakukan hal ini.

Dalam skenario contoh ini, Anda memiliki berbagai situs kerja yang berisi robot industri dalam sel kerja.

Contoh mendefinisikan dua model aset:

- `RobotModel1`: Model aset ini mewakili jenis robot tertentu yang Anda miliki di lokasi kerja Anda. Robot memiliki properti pengukuran, `Temperature`.
- `WorkCell1`: Model aset ini mewakili kumpulan robot dalam salah satu situs kerja Anda. Model aset mendefinisikan hierarki, `robotHierarchyOEM1`, untuk mewakili hubungan bahwa sel kerja berisi robot.

Contoh ini juga mendefinisikan beberapa aset:

- `WorkCell1`: sel kerja dalam situs Boston Anda
- `RobotArm123456`: robot di dalam sel kerja itu
- `RobotArm987654`: robot lain di dalam sel kerja itu

File metadata JSON berikut mendefinisikan model dan aset aset ini. Menjalankan impor massal dengan metadata ini menciptakan model aset dan aset di dalamnya AWS IoT SiteWise, termasuk hubungan hierarkisnya.

File metadata untuk impor

```
{
  "assetModels": [
    {
      "assetModelExternalId": "Robot.OEM1.3536",
      "assetModelName": "RobotModel1",
      "assetModelProperties": [
        {
          "dataType": "DOUBLE",
          "externalId": "Temperature",
          "name": "Temperature",
          "type": {
            "measurement": {
```

```

        "processingConfig": {
            "forwardingConfig": {
                "state": "ENABLED"
            }
        }
    },
    "unit": "fahrenheit"
}
]
},
{
    "assetModelExternalId": "ISA95.WorkCell",
    "assetModelName": "WorkCell",
    "assetModelProperties": [],
    "assetModelHierarchies": [
        {
            "externalId": "workCellHierarchyWithOEM1Robot",
            "name": "robotHierarchyOEM1",
            "childAssetModelExternalId": "Robot.OEM1.3536"
        }
    ]
}
],
"assets": [
    {
        "assetExternalId": "Robot.OEM1.3536.123456",
        "assetName": "RobotArm123456",
        "assetModelExternalId": "Robot.OEM1.3536"
    },
    {
        "assetExternalId": "Robot.OEM1.3536.987654",
        "assetName": "RobotArm987654",
        "assetModelExternalId": "Robot.OEM1.3536"
    },
    {
        "assetExternalId": "BostonSite.Area1.Line1.WorkCell1",
        "assetName": "WorkCell1",
        "assetModelExternalId": "ISA95.WorkCell",
        "assetHierarchies": [
            {
                "externalId": "workCellHierarchyWithOEM1Robot",
                "childAssetExternalId": "Robot.OEM1.3536.123456"
            }
        ]
    },

```

```

    {
      "externalId": "workCellHierarchyWithOEM1Robot",
      "childAssetExternalId": "Robot.OEM1.3536.987654"
    }
  ]
}
]
}

```

Contoh on-boarding awal model dan aset

Dalam skenario contoh ini, Anda memiliki berbagai situs kerja yang berisi robot industri di sebuah perusahaan.

Contoh mendefinisikan beberapa model aset:

- **Sample_Enterprise**— Model aset ini mewakili perusahaan yang menjadi bagian dari situs tersebut. Model aset mendefinisikan hierarki, *Enterprise to Site*, untuk mewakili hubungan situs dengan perusahaan.
- **Sample_Site**— Model aset ini mewakili lokasi manufaktur di dalam perusahaan. Model aset mendefinisikan hierarki, *Site to Line*, untuk mewakili hubungan garis ke situs.
- **Sample_Welding Line**— Model aset ini mewakili jalur perakitan di dalam lokasi kerja. Model aset mendefinisikan hierarki, *Line to Robot*, untuk mewakili hubungan robot dengan garis.
- **Sample_Welding Robot**— Model aset ini mewakili jenis robot tertentu di lokasi kerja Anda.

Contoh ini juga mendefinisikan aset berdasarkan model aset.

- **Sample_AnyCompany Motor** Aset ini dibuat dari model **Sample_Enterprise** aset.
- **Sample_Chicago** Aset ini dibuat dari model **Sample_Site** aset.
- **Sample_Welding Line 1** Aset ini dibuat dari model **Sample_Welding Line** aset.
- **Sample_Welding Robot 1** Aset ini dibuat dari model **Sample_Welding Robot** aset.
- **Sample_Welding Robot 2** Aset ini dibuat dari model **Sample_Welding Robot** aset.

File metadata JSON berikut mendefinisikan model dan aset aset ini. Menjalankan impor massal dengan metadata ini menciptakan model aset dan aset di dalamnya AWS IoT SiteWise, termasuk hubungan hierarkisnya.

File JSON ke aset dan model onboard untuk impor

```

{
  "assetModels": [
    {
      "assetModelExternalId": "External_Id_Welding_Robot",
      "assetModelName": "Sample_Welding Robot",
      "assetModelProperties": [
        {
          "dataType": "STRING",
          "externalId": "External_Id_Welding_Robot_Serial_Number",
          "name": "Serial Number",
          "type": {
            "attribute": {
              "defaultValue": "-"
            }
          },
          "unit": "-"
        },
        {
          "dataType": "DOUBLE",
          "externalId": "External_Id_Welding_Robot_Cycle_Count",
          "name": "CycleCount",
          "type": {
            "measurement": {}
          },
          "unit": "EA"
        },
        {
          "dataType": "DOUBLE",
          "externalId": "External_Id_Welding_Robot_Joint_1_Current",
          "name": "Joint 1 Current",
          "type": {
            "measurement": {}
          },
          "unit": "Amps"
        },
        {
          "dataType": "DOUBLE",
          "externalId": "External_Id_Welding_Robot_Joint_1_Max_Current",
          "name": "Max Joint 1 Current",
          "type": {
            "metric": {

```

```

        "expression": "max(joint1current)",
        "variables": [
            {
                "name": "joint1current",
                "value": {
                    "propertyExternalId":
"External_Id_Welding_Robot_Joint_1_Current"
                }
            }
        ],
        "window": {
            "tumbling": {
                "interval": "5m"
            }
        }
    },
    "unit": "Amps"
}
]
},
{
    "assetModelExternalId": "External_Id_Welding_Line",
    "assetModelName": "Sample_Welding Line",
    "assetModelProperties": [
        {
            "dataType": "DOUBLE",
            "externalId": "External_Id_Welding_Line_Availability",
            "name": "Availability",
            "type": {
                "measurement": {}
            },
            "unit": "%"
        }
    ],
    "assetModelHierarchies": [
        {
            "externalId": "External_Id_Welding_Line_T0_Robot",
            "name": "Line to Robot",
            "childAssetModelExternalId": "External_Id_Welding_Robot"
        }
    ]
},
{

```



```

    "assetModelExternalId": "External_Id_Site",
    "assetModelName": "Sample_Site",
    "assetModelProperties": [
      {
        "dataType": "STRING",
        "externalId": "External_Id_Site_Street_Address",
        "name": "Street Address",
        "type": {
          "attribute": {
            "defaultValue": "-"
          }
        },
        "unit": "-"
      }
    ],
    "assetModelHierarchies": [
      {
        "externalId": "External_Id_Site_T0_Line",
        "name": "Site to Line",
        "childAssetModelExternalId": "External_Id_Welding_Line"
      }
    ]
  },
  {
    "assetModelExternalId": "External_Id_Enterprise",
    "assetModelName": "Sample_Enterprise",
    "assetModelProperties": [
      {
        "dataType": "STRING",
        "name": "Company Name",
        "externalId": "External_Id_Enterprise_Company_Name",
        "type": {
          "attribute": {
            "defaultValue": "-"
          }
        },
        "unit": "-"
      }
    ],
    "assetModelHierarchies": [
      {
        "externalId": "External_Id_Enterprise_T0_Site",
        "name": "Enterprise to Site",
        "childAssetModelExternalId": "External_Id_Site"
      }
    ]
  }
]

```

```

    }
  ]
}
],
"assets": [
  {
    "assetExternalId": "External_Id_Welding_Robot_1",
    "assetName": "Sample_Welding Robot 1",
    "assetModelExternalId": "External_Id_Welding_Robot",
    "assetProperties": [
      {
        "externalId": "External_Id_Welding_Robot_Serial_Number",
        "attributeValue": "S1000"
      },
      {
        "externalId": "External_Id_Welding_Robot_Cycle_Count",
        "alias": "AnyCompany/Chicago/Welding Line/S1000/Count"
      },
      {
        "externalId": "External_Id_Welding_Robot_Joint_1_Current",
        "alias": "AnyCompany/Chicago/Welding Line/S1000/1/Current"
      }
    ]
  },
  {
    "assetExternalId": "External_Id_Welding_Robot_2",
    "assetName": "Sample_Welding Robot 2",
    "assetModelExternalId": "External_Id_Welding_Robot",
    "assetProperties": [
      {
        "externalId": "External_Id_Welding_Robot_Serial_Number",
        "attributeValue": "S2000"
      },
      {
        "externalId": "External_Id_Welding_Robot_Cycle_Count",
        "alias": "AnyCompany/Chicago/Welding Line/S2000/Count"
      },
      {
        "externalId": "External_Id_Welding_Robot_Joint_1_Current",
        "alias": "AnyCompany/Chicago/Welding Line/S2000/1/Current"
      }
    ]
  },
  {

```

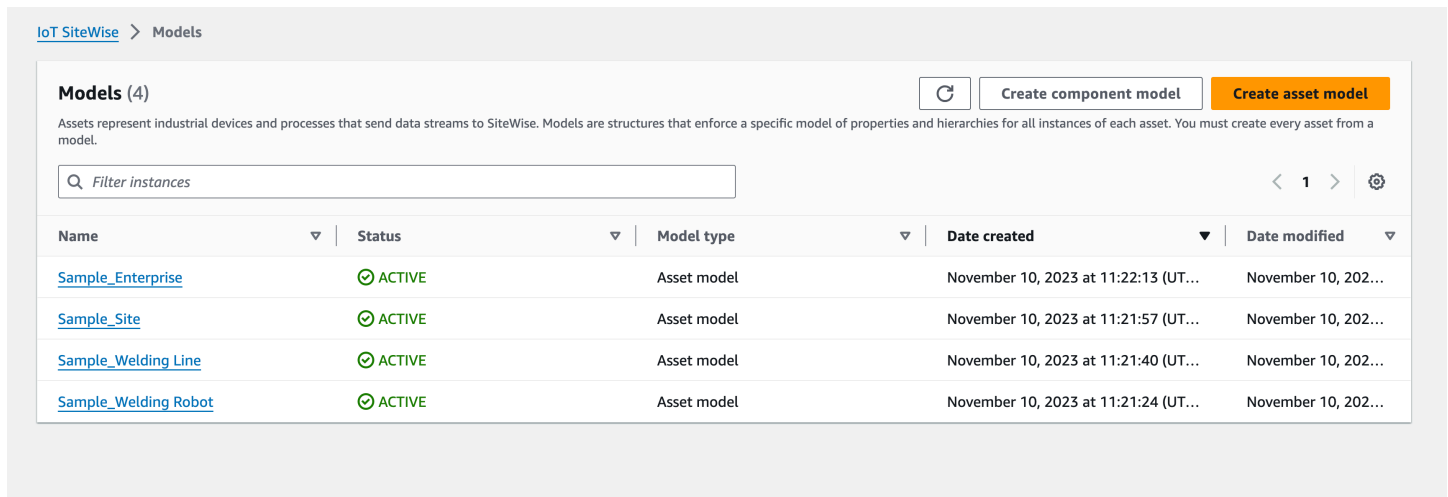
```

    "assetExternalId": "External_Id_Welding_Line_1",
    "assetName": "Sample_Welding Line 1",
    "assetModelExternalId": "External_Id_Welding_Line",
    "assetProperties": [
      {
        "externalId": "External_Id_Welding_Line_Availability",
        "alias": "AnyCompany/Chicago/Welding Line/Availability"
      }
    ],
    "assetHierarchies": [
      {
        "externalId": "External_Id_Welding_Line_T0_Robot",
        "childAssetExternalId": "External_Id_Welding_Robot_1"
      },
      {
        "externalId": "External_Id_Welding_Line_T0_Robot",
        "childAssetExternalId": "External_Id_Welding_Robot_2"
      }
    ]
  },
  {
    "assetExternalId": "External_Id_Site_Chicago",
    "assetName": "Sample_Chicago",
    "assetModelExternalId": "External_Id_Site",
    "assetHierarchies": [
      {
        "externalId": "External_Id_Site_T0_Line",
        "childAssetExternalId": "External_Id_Welding_Line_1"
      }
    ]
  },
  {
    "assetExternalId": "External_Id_Enterprise_AnyCompany",
    "assetName": "Sample_AnyEnterprise Motor",
    "assetModelExternalId": "External_Id_Enterprise",
    "assetHierarchies": [
      {
        "externalId": "External_Id_Enterprise_T0_Site",
        "childAssetExternalId": "External_Id_Site_Chicago"
      }
    ]
  }
]

```

}

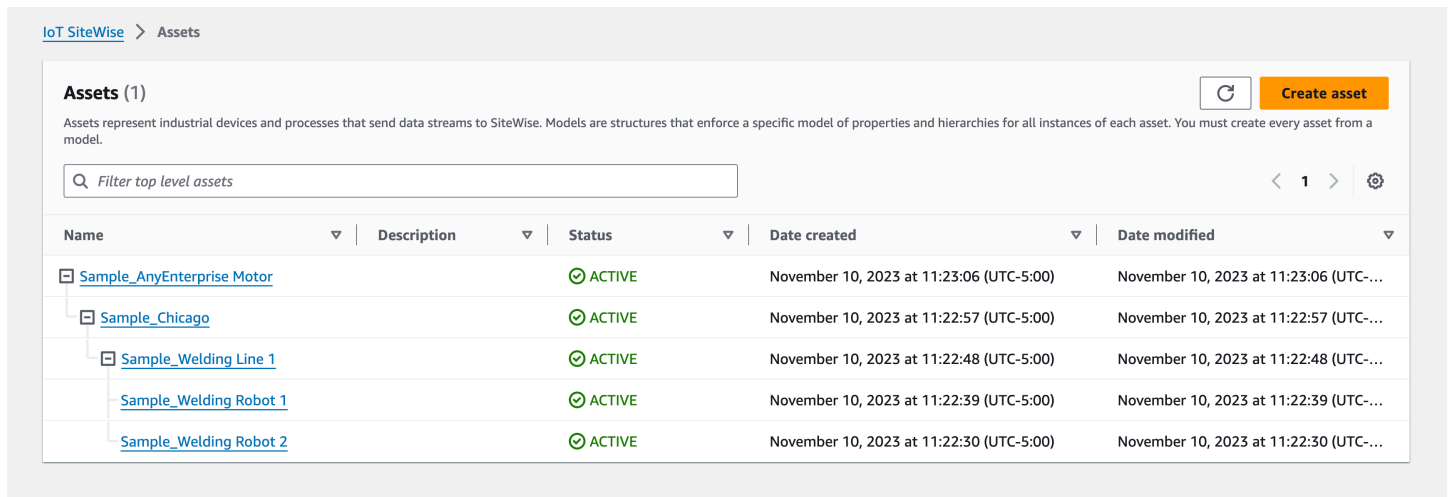
Screenshot berikut adalah model yang ditampilkan Konsol AWS IoT SiteWise setelah Anda menjalankan contoh kode sebelumnya.



The screenshot shows the 'Models' page in the AWS IoT SiteWise console. It displays a list of four models, all with a status of 'ACTIVE'. The models are: Sample_Enterprise, Sample_Site, Sample_Welding Line, and Sample_Welding Robot. Each model is an 'Asset model' type. The table includes columns for Name, Status, Model type, Date created, and Date modified.

Name	Status	Model type	Date created	Date modified
Sample_Enterprise	ACTIVE	Asset model	November 10, 2023 at 11:22:13 (UT...)	November 10, 202...
Sample_Site	ACTIVE	Asset model	November 10, 2023 at 11:21:57 (UT...)	November 10, 202...
Sample_Welding Line	ACTIVE	Asset model	November 10, 2023 at 11:21:40 (UT...)	November 10, 202...
Sample_Welding Robot	ACTIVE	Asset model	November 10, 2023 at 11:21:24 (UT...)	November 10, 202...

Screenshot berikut adalah model, aset, dan hierarki yang ditampilkan di Konsol AWS IoT SiteWise setelah Anda menjalankan contoh kode sebelumnya.



The screenshot shows the 'Assets' page in the AWS IoT SiteWise console. It displays a hierarchical structure of assets. The root asset is 'Sample_AnyEnterprise Motor', which has three child assets: 'Sample_Chicago', 'Sample_Welding Line 1', and 'Sample_Welding Robot 1'. The 'Sample_Welding Line 1' asset has two child assets: 'Sample_Welding Robot 1' and 'Sample_Welding Robot 2'. All assets have a status of 'ACTIVE'. The table includes columns for Name, Description, Status, Date created, and Date modified.

Name	Description	Status	Date created	Date modified
Sample_AnyEnterprise Motor		ACTIVE	November 10, 2023 at 11:23:06 (UTC-5:00)	November 10, 2023 at 11:23:06 (UTC-...
Sample_Chicago		ACTIVE	November 10, 2023 at 11:22:57 (UTC-5:00)	November 10, 2023 at 11:22:57 (UTC-...
Sample_Welding Line 1		ACTIVE	November 10, 2023 at 11:22:48 (UTC-5:00)	November 10, 2023 at 11:22:48 (UTC-...
Sample_Welding Robot 1		ACTIVE	November 10, 2023 at 11:22:39 (UTC-5:00)	November 10, 2023 at 11:22:39 (UTC-...
Sample_Welding Robot 2		ACTIVE	November 10, 2023 at 11:22:30 (UTC-5:00)	November 10, 2023 at 11:22:30 (UTC-...

Contoh orientasi aset tambahan

Contoh ini mendefinisikan aset tambahan untuk diimpor ke model aset yang ada di akun Anda:

- **Sample_Welding Line 2** Aset ini dibuat dari model **Sample_Welding Line** aset.
- **Sample_Welding Robot 3** Aset ini dibuat dari model **Sample_Welding Robot** aset.
- **Sample_Welding Robot 4** Aset ini dibuat dari model **Sample_Welding Robot** aset.

Untuk membuat aset awal untuk contoh ini, lihat [Contoh on-boarding awal model dan aset](#).

File metadata JSON berikut mendefinisikan model dan aset aset ini. Menjalankan impor massal dengan metadata ini menciptakan model aset dan aset di dalamnya AWS IoT SiteWise, termasuk hubungan hierarkisnya.

File JSON ke aset tambahan onboard

```
{
  "assets": [
    {
      "assetExternalId": "External_Id_Welding_Robot_3",
      "assetName": "Sample_Welding Robot 3",
      "assetModelExternalId": "External_Id_Welding_Robot",
      "assetProperties": [
        {
          "externalId": "External_Id_Welding_Robot_Serial_Number",
          "attributeValue": "S3000"
        },
        {
          "externalId": "External_Id_Welding_Robot_Cycle_Count",
          "alias": "AnyCompany/Chicago/Welding Line/S3000/Count"
        },
        {
          "externalId": "External_Id_Welding_Robot_Joint_1_Current",
          "alias": "AnyCompany/Chicago/Welding Line/S3000/1/Current"
        }
      ]
    },
    {
      "assetExternalId": "External_Id_Welding_Robot_4",
      "assetName": "Sample_Welding Robot 4",
      "assetModelExternalId": "External_Id_Welding_Robot",
      "assetProperties": [
        {
          "externalId": "External_Id_Welding_Robot_Serial_Number",
          "attributeValue": "S4000"
        },
        {
          "externalId": "External_Id_Welding_Robot_Cycle_Count",
          "alias": "AnyCompany/Chicago/Welding Line/S4000/Count"
        }
      ]
    }
  ]
}
```

```

        {
            "externalId": "External_Id_Welding_Robot_Joint_1_Current",
            "alias": "AnyCompany/Chicago/Welding Line/S4000/1/Current"
        }
    ],
},
{
    "assetExternalId": "External_Id_Welding_Line_1",
    "assetName": "Sample_Welding Line 1",
    "assetModelExternalId": "External_Id_Welding_Line",
    "assetHierarchies": [
        {
            "externalId": "External_Id_Welding_Line_T0_Robot",
            "childAssetExternalId": "External_Id_Welding_Robot_1"
        },
        {
            "externalId": "External_Id_Welding_Line_T0_Robot",
            "childAssetExternalId": "External_Id_Welding_Robot_2"
        },
        {
            "externalId": "External_Id_Welding_Line_T0_Robot",
            "childAssetExternalId": "External_Id_Welding_Robot_3"
        }
    ]
},
{
    "assetExternalId": "External_Id_Welding_Line_2",
    "assetName": "Sample_Welding Line 2",
    "assetModelExternalId": "External_Id_Welding_Line",
    "assetHierarchies": [
        {
            "externalId": "External_Id_Welding_Line_T0_Robot",
            "childAssetExternalId": "External_Id_Welding_Robot_4"
        }
    ]
},
{
    "assetExternalId": "External_Id_Site_Chicago",
    "assetName": "Sample_Chicago",
    "assetModelExternalId": "External_Id_Site",
    "assetHierarchies": [
        {
            "externalId": "External_Id_Site_T0_Line",
            "childAssetExternalId": "External_Id_Welding_Line_1"
        }
    ]
}

```

```

    },
    {
      "externalId": "External_Id_Site_T0_Line",
      "childAssetExternalId": "External_Id_Welding_Line_2"
    }
  ]
}
]
}
}

```

Screenshot berikut adalah model, aset, dan hierarki yang ditampilkan di Konsol AWS IoT SiteWise setelah Anda menjalankan contoh kode sebelumnya.

The screenshot shows the AWS IoT SiteWise Assets console. At the top, there is a search bar with the text "Filter top level assets" and a "Create asset" button. Below the search bar is a table with columns: Name, Description, Status, Date created, and Date modified. The table contains a list of assets, including "Sample_AnyCompany Motor", "Sample_Chicago", "Sample_Welding Line 1", "Sample_Welding Robot 2", "Sample_Welding Robot 3", "Sample_Welding Robot 1", "Sample_Welding Line 2", and "Sample_Welding Robot 4". All assets are listed as "ACTIVE".

Name	Description	Status	Date created	Date modified
Sample_AnyCompany Motor		ACTIVE	November 09, 2023 at 19:18:05 (UTC-5:00)	November 09, 2023 at 19:18:05 (UTC-5:00)
Sample_Chicago		ACTIVE	November 09, 2023 at 19:17:56 (UTC-5:00)	November 09, 2023 at 19:17:56 (UTC-5:00)
Sample_Welding Line 1		ACTIVE	November 09, 2023 at 19:17:48 (UTC-5:00)	November 09, 2023 at 19:17:48 (UTC-5:00)
Sample_Welding Robot 2		ACTIVE	November 09, 2023 at 19:17:39 (UTC-5:00)	November 09, 2023 at 19:51:05 (UTC-5:00)
Sample_Welding Robot 3		ACTIVE	November 09, 2023 at 20:40:02 (UTC-5:00)	November 09, 2023 at 20:40:02 (UTC-5:00)
Sample_Welding Robot 1		ACTIVE	November 09, 2023 at 19:17:30 (UTC-5:00)	November 09, 2023 at 19:51:05 (UTC-5:00)
Sample_Welding Line 2		ACTIVE	November 09, 2023 at 20:40:20 (UTC-5:00)	November 09, 2023 at 20:40:20 (UTC-5:00)
Sample_Welding Robot 4		ACTIVE	November 09, 2023 at 20:40:11 (UTC-5:00)	November 09, 2023 at 20:40:11 (UTC-5:00)

Contoh orientasi properti baru

Contoh ini mendefinisikan properti baru pada model aset yang ada. Lihat [Contoh orientasi aset tambahan](#) untuk memasukkan aset dan model tambahan.

- **Joint 1 Temperature**— Properti ini ditambahkan ke model `Sample_Welding Robot` aset. Properti baru ini juga akan menyebar ke setiap aset yang dibuat dari model `Sample_Welding Robot` aset.

Untuk menambahkan properti baru ke model aset yang ada, lihat contoh file metadata JSON berikut. Seperti yang ditunjukkan dalam JSON, seluruh definisi model `Sample_Welding Robot` aset yang

ada harus disediakan bersama dengan properti baru. Jika seluruh daftar properti dari definisi yang ada tidak disediakan, AWS IoT SiteWise menghapus properti yang dihilangkan.

File JSON ke properti baru onboard

Contoh ini menambahkan properti baru Joint 1 Temperature ke model aset.

```
{
  "assetModels": [
    {
      "assetModelExternalId": "External_Id_Welding_Robot",
      "assetModelName": "Sample_Welding_Robot",
      "assetModelProperties": [
        {
          "dataType": "STRING",
          "externalId": "External_Id_Welding_Robot_Serial_Number",
          "name": "Serial Number",
          "type": {
            "attribute": {
              "defaultValue": "-"
            }
          },
          "unit": "-"
        },
        {
          "dataType": "DOUBLE",
          "externalId": "External_Id_Welding_Robot_Cycle_Count",
          "name": "CycleCount",
          "type": {
            "measurement": {}
          },
          "unit": "EA"
        },
        {
          "dataType": "DOUBLE",
          "externalId": "External_Id_Welding_Robot_Joint_1_Current",
          "name": "Joint 1 Current",
          "type": {
            "measurement": {}
          },
          "unit": "Amps"
        },
        {

```



```

        "dataType": "DOUBLE",
        "externalId": "External_Id_Welding_Robot_Joint_1_Max_Current",
        "name": "Max Joint 1 Current",
        "type": {
            "metric": {
                "expression": "max(joint1current)",
                "variables": [
                    {
                        "name": "joint1current",
                        "value": {
                            "propertyExternalId":
"External_Id_Welding_Robot_Joint_1_Current"
                        }
                    }
                ],
                "window": {
                    "tumbling": {
                        "interval": "5m"
                    }
                }
            }
        },
        "unit": "Amps"
    },
    {
        "dataType": "DOUBLE",
        "externalId": "External_Id_Welding_Robot_Joint_1_Temperature",
        "name": "Joint 1 Temperature",
        "type": {
            "measurement": {}
        },
        "unit": "degC"
    }
]
}

```

Contoh metadata ekspor

Saat melakukan ekspor massal AWS IoT SiteWise konten ke Amazon S3, Anda dapat menentukan filter untuk membatasi model aset dan aset tertentu yang ingin Anda ekspor.

Anda menentukan filter di `iotSiteWiseConfiguration` bagian dalam `sources` bagian badan permintaan Anda.

Note

Anda dapat menyertakan beberapa filter. Operasi massal akan mengekspor model aset atau aset apa pun yang cocok dengan filter mana pun.

Jika Anda tidak memberikan filter apa pun, maka operasi akan mengekspor semua model aset dan aset Anda.

```
{
  "metadataTransferJobId": "your-transfer-job-id",
  "sources": [{
    "type": "iotsitewise",
    "iotSiteWiseConfiguration": {
      "filters": [{
        list of filters
      }]
    }
  ]],
  "destination": {
    "type": "s3",
    "s3Configuration": {
      "location": "arn:aws:s3:::your-S3-bucket-location"
    }
  }
}
```

Pemfilteran berdasarkan model aset

Anda dapat memfilter model aset tertentu. Anda juga dapat menyertakan semua aset menggunakan model tersebut, atau semua model aset dalam hierarki. Anda tidak dapat menyertakan aset dan hierarki.

Untuk informasi selengkapnya tentang hierarki, lihat [Mendefinisikan hierarki model aset](#).

Asset model

Filter ini mencakup model aset yang ditentukan:

```
"filterByAssetModel": {  
  "assetModelId": "asset model ID"  
}
```

Asset model and its assets

Filter ini mencakup model aset yang ditentukan, bersama dengan semua aset yang menggunakan model aset tersebut:

```
"filterByAssetModel": {  
  "assetModelId": "asset model ID",  
  "includeAssets": true  
}
```

Asset model and its hierarchy

Filter ini mencakup model aset yang ditentukan, bersama dengan semua model aset terkait dalam hierarki:

```
"filterByAssetModel": {  
  "assetModelId": "asset model ID",  
  "includeOffspring": true  
}
```

Pemfilteran berdasarkan aset

Anda dapat memfilter aset tertentu. Anda juga dapat menyertakan model asetnya, atau semua aset terkait dalam hierarki. Anda tidak dapat menyertakan model aset dan hierarki.

Untuk informasi selengkapnya tentang hierarki, lihat [Mendefinisikan hierarki model aset](#).

Asset

Filter ini mencakup aset yang ditentukan:

```
"filterByAsset": {  
  "assetId": "asset ID"  
}
```

Asset and its asset model

Filter ini mencakup aset yang ditentukan, bersama dengan model aset yang digunakannya:

```
"filterByAsset": {
  "assetId": "asset ID",
  "includeAssetModel": true
}
```

Asset and its hierarchy

Filter ini mencakup aset yang ditentukan, bersama dengan semua aset terkait dalam hierarki:

```
"filterByAsset": {
  "assetId": "asset ID",
  "includeOffspring": true
}
```

AWS IoT SiteWise skema pekerjaan transfer metadata

Gunakan skema pekerjaan transfer AWS IoT SiteWise metadata untuk referensi saat melakukan operasi impor dan ekspor massal Anda sendiri:

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "title": "IoTSiteWise",
  "description": "Metadata transfer job resource schema for IoTSiteWise",
  "definitions": {
    "Name": {
      "type": "string",
      "minLength": 1,
      "maxLength": 256,
      "pattern": "[^\\u0000-\\u001F\\u007F]+"
    },
    "Description": {
      "type": "string",
      "minLength": 1,
      "maxLength": 2048,
      "pattern": "[^\\u0000-\\u001F\\u007F]+"
    },
    "ID": {
      "type": "string",
```

```

    "minLength": 36,
    "maxLength": 36,
    "pattern": "^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$"
  },
  "ExternalId": {
    "type": "string",
    "minLength": 2,
    "maxLength": 128,
    "pattern": "[a-zA-Z0-9_][a-zA-Z_\\-0-9.:]*[a-zA-Z0-9_]+"
  },
  "AttributeValue": {
    "description": "The value of the property attribute.",
    "type": "string",
    "minLength": 1,
    "maxLength": 1024,
    "pattern": "[^\\u0000-\\u001F\\u007F]+"
  },
  "PropertyUnit": {
    "description": "The unit of measure (such as Newtons or RPM) of the asset
property.",
    "type": "string",
    "minLength": 1,
    "maxLength": 256,
    "pattern": "[^\\u0000-\\u001F\\u007F]+"
  },
  "PropertyAlias": {
    "description": "The property alias that identifies the property.",
    "type": "string",
    "minLength": 1,
    "maxLength": 1000,
    "pattern": "[^\\u0000-\\u001F\\u007F]+"
  },
  "AssetProperty": {
    "description": "The asset property's definition, alias, unit, and notification
state.",
    "type": "object",
    "additionalProperties": false,
    "anyOf": [
      {
        "required": [
          "id"
        ]
      }
    ],
  },

```

```

    "required": [
      "externalId"
    ]
  },
],
"properties": {
  "id": {
    "description": "The ID of the asset property.",
    "$ref": "#/definitions/ID"
  },
  "externalId": {
    "description": "The ExternalID of the asset property.",
    "$ref": "#/definitions/ExternalId"
  },
  "alias": {
    "$ref": "#/definitions/PropertyAlias"
  },
  "unit": {
    "$ref": "#/definitions/PropertyUnit"
  },
  "attributeValue": {
    "$ref": "#/definitions/AttributeValue"
  },
  "retainDataOnAliasChange": {
    "type": "string",
    "default": "TRUE",
    "enum": [
      "TRUE",
      "FALSE"
    ]
  },
  "propertyNotificationState": {
    "description": "The MQTT notification state (ENABLED or DISABLED) for this
asset property.",
    "type": "string",
    "enum": [
      "ENABLED",
      "DISABLED"
    ]
  }
}
},
"AssetHierarchy": {
  "description": "A hierarchy specifies allowed parent/child asset relationships.",

```

```
"type": "object",
"additionalProperties": false,
"anyOf": [
  {
    "required": [
      "id",
      "childAssetId"
    ]
  },
  {
    "required": [
      "externalId",
      "childAssetId"
    ]
  },
  {
    "required": [
      "id",
      "childAssetExternalId"
    ]
  },
  {
    "required": [
      "externalId",
      "childAssetExternalId"
    ]
  }
],
"properties": {
  "id": {
    "description": "The ID of a hierarchy in the parent asset's model.",
    "$ref": "#/definitions/ID"
  },
  "externalId": {
    "description": "The ExternalID of a hierarchy in the parent asset's model.",
    "$ref": "#/definitions/ExternalID"
  },
  "childAssetId": {
    "description": "The ID of the child asset to be associated.",
    "$ref": "#/definitions/ID"
  },
  "childAssetExternalId": {
    "description": "The ExternalID of the child asset to be associated.",
    "$ref": "#/definitions/ExternalID"
  }
}
```

```
    }
  }
},
"Tag": {
  "type": "object",
  "additionalProperties": false,
  "required": [
    "key",
    "value"
  ],
  "properties": {
    "key": {
      "type": "string"
    },
    "value": {
      "type": "string"
    }
  }
},
"AssetModelType": {
  "type": "string",
  "default": null,
  "enum": [
    "ASSET_MODEL",
    "COMPONENT_MODEL"
  ]
},
"AssetModelCompositeModel": {
  "description": "Contains a composite model definition in an asset model. This composite model definition is applied to all assets created from the asset model.",
  "type": "object",
  "additionalProperties": false,
  "anyOf": [
    {
      "required": [
        "id"
      ]
    },
    {
      "required": [
        "externalId"
      ]
    }
  ]
},
],
```



```
"required": [
  "name",
  "type"
],
"properties": {
  "id": {
    "description": "The ID of the asset model composite model.",
    "$ref": "#/definitions/ID"
  },
  "externalId": {
    "description": "The ExternalID of the asset model composite model.",
    "$ref": "#/definitions/ExternalId"
  },
  "parentId": {
    "description": "The ID of the parent asset model composite model.",
    "$ref": "#/definitions/ID"
  },
  "parentExternalId": {
    "description": "The ExternalID of the parent asset model composite model.",
    "$ref": "#/definitions/ExternalId"
  },
  "composedAssetModelId": {
    "description": "The ID of the composed asset model.",
    "$ref": "#/definitions/ID"
  },
  "composedAssetModelExternalId": {
    "description": "The ExternalID of the composed asset model.",
    "$ref": "#/definitions/ExternalId"
  },
  "description": {
    "description": "A description for the asset composite model.",
    "$ref": "#/definitions/Description"
  },
  "name": {
    "description": "A unique, friendly name for the asset composite model.",
    "$ref": "#/definitions/Name"
  },
  "type": {
    "description": "The type of the composite model. For alarm composite models,
this type is AWS/ALARM.",
    "$ref": "#/definitions/Name"
  },
  "properties": {
    "description": "The property definitions of the asset model.",
```

```
        "type": "array",
        "items": {
            "$ref": "#/definitions/AssetModelProperty"
        }
    }
},
"AssetModelProperty": {
    "description": "Contains information about an asset model property.",
    "type": "object",
    "additionalProperties": false,
    "anyOf": [
        {
            "required": [
                "id"
            ]
        },
        {
            "required": [
                "externalId"
            ]
        }
    ],
    "required": [
        "name",
        "dataType",
        "type"
    ],
    "properties": {
        "id": {
            "description": "The ID of the asset model property.",
            "$ref": "#/definitions/ID"
        },
        "externalId": {
            "description": "The ExternalID of the asset model property.",
            "$ref": "#/definitions/ExternalId"
        },
        "name": {
            "description": "The name of the asset model property.",
            "$ref": "#/definitions/Name"
        },
        "dataType": {
            "description": "The data type of the asset model property.",
            "$ref": "#/definitions/DataType"
        }
    }
}
```

```

    },
    "dataKeySpec": {
      "description": "The data type of the structure for this property.",
      "$ref": "#/definitions/Name"
    },
    },
    "unit": {
      "description": "The unit of the asset model property, such as Newtons or
RPM.",
      "type": "string",
      "minLength": 1,
      "maxLength": 256,
      "pattern": "[^\\u0000-\\u001F\\u007F]+"
    },
    },
    "type": {
      "description": "The property type",
      "$ref": "#/definitions/PropertyType"
    }
  }
},
"DataType": {
  "type": "string",
  "enum": [
    "STRING",
    "INTEGER",
    "DOUBLE",
    "BOOLEAN",
    "STRUCT"
  ]
},
"PropertyType": {
  "description": "Contains a property type, which can be one of attribute,
measurement, metric, or transform.",
  "type": "object",
  "additionalProperties": false,
  "properties": {
    "attribute": {
      "$ref": "#/definitions/Attribute"
    },
    },
    "transform": {
      "$ref": "#/definitions/Transform"
    },
    },
    "metric": {
      "$ref": "#/definitions/Metric"
    }
  },
},

```

```

    "measurement": {
      "$ref": "#/definitions/Measurement"
    }
  },
  "Attribute": {
    "type": "object",
    "additionalProperties": false,
    "properties": {
      "defaultValue": {
        "type": "string",
        "minLength": 1,
        "maxLength": 1024,
        "pattern": "[^\\u0000-\\u001F\\u007F]+"
      }
    }
  },
  "Transform": {
    "type": "object",
    "additionalProperties": false,
    "required": [
      "expression",
      "variables"
    ],
    "properties": {
      "expression": {
        "description": "The mathematical expression that defines the transformation function.",
        "type": "string",
        "minLength": 1,
        "maxLength": 1024
      },
      "variables": {
        "description": "The list of variables used in the expression.",
        "type": "array",
        "items": {
          "$ref": "#/definitions/ExpressionVariable"
        }
      }
    },
    "processingConfig": {
      "$ref": "#/definitions/TransformProcessingConfig"
    }
  }
},

```

```
"TransformProcessingConfig": {
  "description": "The processing configuration for the given transform property.",
  "type": "object",
  "additionalProperties": false,
  "required": [
    "computeLocation"
  ],
  "properties": {
    "computeLocation": {
      "description": "The compute location for the given transform property.",
      "$ref": "#/definitions/ComputeLocation"
    },
    "forwardingConfig": {
      "description": "The forwarding configuration for a given property.",
      "$ref": "#/definitions/ForwardingConfig"
    }
  }
},
"Metric": {
  "type": "object",
  "additionalProperties": false,
  "required": [
    "expression",
    "variables",
    "window"
  ],
  "properties": {
    "expression": {
      "description": "The mathematical expression that defines the metric aggregation function.",
      "type": "string",
      "minLength": 1,
      "maxLength": 1024
    },
    "variables": {
      "description": "The list of variables used in the expression.",
      "type": "array",
      "items": {
        "$ref": "#/definitions/ExpressionVariable"
      }
    },
    "window": {
      "description": "The window (time interval) over which AWS IoT SiteWise computes the metric's aggregation expression",

```

```
    "$ref": "#/definitions/MetricWindow"
  },
  "processingConfig": {
    "$ref": "#/definitions/MetricProcessingConfig"
  }
},
"MetricProcessingConfig": {
  "description": "The processing configuration for the metric.",
  "type": "object",
  "additionalProperties": false,
  "required": [
    "computeLocation"
  ],
  "properties": {
    "computeLocation": {
      "description": "The compute location for the given metric property.",
      "$ref": "#/definitions/ComputeLocation"
    }
  }
},
"ComputeLocation": {
  "type": "string",
  "enum": [
    "EDGE",
    "CLOUD"
  ]
},
"ForwardingConfig": {
  "type": "object",
  "additionalProperties": false,
  "required": [
    "state"
  ],
  "properties": {
    "state": {
      "type": "string",
      "enum": [
        "ENABLED",
        "DISABLED"
      ]
    }
  }
},
},
```

```
"MetricWindow": {
  "description": "Contains a time interval window used for data aggregate
computations (for example, average, sum, count, and so on).",
  "type": "object",
  "additionalProperties": false,
  "properties": {
    "tumbling": {
      "description": "The tumbling time interval window.",
      "type": "object",
      "additionalProperties": false,
      "required": [
        "interval"
      ],
      "properties": {
        "interval": {
          "description": "The time interval for the tumbling window.",
          "type": "string",
          "minLength": 2,
          "maxLength": 23
        },
        "offset": {
          "description": "The offset for the tumbling window.",
          "type": "string",
          "minLength": 2,
          "maxLength": 25
        }
      }
    }
  }
},
"ExpressionVariable": {
  "type": "object",
  "additionalProperties": false,
  "required": [
    "name",
    "value"
  ],
  "properties": {
    "name": {
      "description": "The friendly name of the variable to be used in the
expression.",
      "type": "string",
      "minLength": 1,
      "maxLength": 64,
```

```

    "pattern": "^[a-z][a-z0-9_]*$"
  },
  "value": {
    "description": "The variable that identifies an asset property from which to
use values.",
    "$ref": "#/definitions/VariableValue"
  }
},
"VariableValue": {
  "type": "object",
  "additionalProperties": false,
  "anyOf": [
    {
      "required": [
        "propertyId"
      ]
    },
    {
      "required": [
        "propertyExternalId"
      ]
    }
  ],
  "properties": {
    "propertyId": {
      "$ref": "#/definitions/ID"
    },
    "propertyExternalId": {
      "$ref": "#/definitions/ExternalId"
    },
    "hierarchyId": {
      "$ref": "#/definitions/ID"
    },
    "hierarchyExternalId": {
      "$ref": "#/definitions/ExternalId"
    }
  }
},
"Measurement": {
  "type": "object",
  "additionalProperties": false,
  "properties": {
    "processingConfig": {

```



```
        "$ref": "#/definitions/MeasurementProcessingConfig"
      }
    }
  },
  "MeasurementProcessingConfig": {
    "type": "object",
    "additionalProperties": false,
    "required": [
      "forwardingConfig"
    ],
    "properties": {
      "forwardingConfig": {
        "description": "The forwarding configuration for the given measurement property.",
        "$ref": "#/definitions/ForwardingConfig"
      }
    }
  },
  "AssetModelHierarchy": {
    "description": "Contains information about an asset model hierarchy.",
    "type": "object",
    "additionalProperties": false,
    "anyOf": [
      {
        "required": [
          "id",
          "childAssetModelId"
        ]
      },
      {
        "required": [
          "id",
          "childAssetModelExternalId"
        ]
      },
      {
        "required": [
          "externalId",
          "childAssetModelId"
        ]
      },
      {
        "required": [
          "externalId",
```

```

        "childAssetModelExternalId"
      ]
    }
  ],
  "required": [
    "name"
  ],
  "properties": {
    "id": {
      "description": "The ID of the asset model hierarchy.",
      "$ref": "#/definitions/ID"
    },
    "externalId": {
      "description": "The ExternalID of the asset model hierarchy.",
      "$ref": "#/definitions/ExternalId"
    },
    "name": {
      "description": "The name of the asset model hierarchy.",
      "$ref": "#/definitions/Name"
    },
    "childAssetModelId": {
      "description": "The ID of the asset model. All assets in this hierarchy must
be instances of the child AssetModelId asset model.",
      "$ref": "#/definitions/ID"
    },
    "childAssetModelExternalId": {
      "description": "The ExternalID of the asset model. All assets in this
hierarchy must be instances of the child AssetModelId asset model.",
      "$ref": "#/definitions/ExternalId"
    }
  }
},
"AssetModel": {
  "type": "object",
  "additionalProperties": false,
  "anyOf": [
    {
      "required": [
        "assetModelId"
      ]
    },
    {
      "required": [
        "assetModelExternalId"
      ]
    }
  ]
}

```

```

    ]
  }
],
"required": [
  "assetModelName"
],
"properties": {
  "assetModelId": {
    "description": "The ID of the asset model.",
    "$ref": "#/definitions/ID"
  },
  "assetModelExternalId": {
    "description": "The ID of the asset model.",
    "$ref": "#/definitions/ExternalId"
  },
  "assetModelName": {
    "description": "A unique, friendly name for the asset model.",
    "$ref": "#/definitions/Name"
  },
  "assetModelDescription": {
    "description": "A description for the asset model.",
    "$ref": "#/definitions/Description"
  },
  "assetModelType": {
    "description": "The type of the asset model.",
    "$ref": "#/definitions/AssetModelType"
  },
  "assetModelProperties": {
    "description": "The property definitions of the asset model.",
    "type": "array",
    "items": {
      "$ref": "#/definitions/AssetModelProperty"
    }
  },
  "assetModelCompositeModels": {
    "description": "The composite asset models that are part of this asset model. Composite asset models are asset models that contain specific properties.",
    "type": "array",
    "items": {
      "$ref": "#/definitions/AssetModelCompositeModel"
    }
  },
  "assetModelHierarchies": {

```

```
    "description": "The hierarchy definitions of the asset model. Each hierarchy specifies an asset model whose assets can be children of any other assets created from this asset model.",
    "type": "array",
    "items": {
      "$ref": "#/definitions/AssetModelHierarchy"
    }
  },
  "tags": {
    "description": "A list of key-value pairs that contain metadata for the asset model.",
    "type": "array",
    "items": {
      "$ref": "#/definitions/Tag"
    }
  }
},
"Asset": {
  "type": "object",
  "additionalProperties": false,
  "anyOf": [
    {
      "required": [
        "assetId",
        "assetModelId"
      ]
    },
    {
      "required": [
        "assetExternalId",
        "assetModelId"
      ]
    },
    {
      "required": [
        "assetId",
        "assetModelExternalId"
      ]
    },
    {
      "required": [
        "assetExternalId",
        "assetModelExternalId"
      ]
    }
  ]
}
```

```

    ]
  }
],
"required": [
  "assetName"
],
"properties": {
  "assetId": {
    "description": "The ID of the asset",
    "$ref": "#/definitions/ID"
  },
  "assetExternalId": {
    "description": "The external ID of the asset",
    "$ref": "#/definitions/ExternalId"
  },
  "assetModelId": {
    "description": "The ID of the asset model from which to create the asset.",
    "$ref": "#/definitions/ID"
  },
  "assetModelExternalId": {
    "description": "The ExternalID of the asset model from which to create the
asset.",
    "$ref": "#/definitions/ExternalId"
  },
  "assetName": {
    "description": "A unique, friendly name for the asset.",
    "$ref": "#/definitions/Name"
  },
  "assetDescription": {
    "description": "A description for the asset",
    "$ref": "#/definitions/Description"
  },
  "assetProperties": {
    "type": "array",
    "items": {
      "$ref": "#/definitions/AssetProperty"
    }
  },
  "assetHierarchies": {
    "type": "array",
    "items": {
      "$ref": "#/definitions/AssetHierarchy"
    }
  }
},

```

```
    "tags": {
      "description": "A list of key-value pairs that contain metadata for the
asset.",
      "type": "array",
      "uniqueItems": false,
      "items": {
        "$ref": "#/definitions/Tag"
      }
    }
  },
  "additionalProperties": false,
  "properties": {
    "assetModels": {
      "type": "array",
      "uniqueItems": false,
      "items": {
        "$ref": "#/definitions/AssetModel"
      }
    },
    "assets": {
      "type": "array",
      "uniqueItems": false,
      "items": {
        "$ref": "#/definitions/Asset"
      }
    }
  }
}
```

Memantau data dengan alarm

Anda dapat mengonfigurasi alarm untuk data Anda untuk mengingatkan tim Anda saat peralatan atau proses berjalan secara sub-optimal. Kinerja optimal mesin atau proses berarti bahwa nilai untuk metrik tertentu harus berada dalam kisaran batas tinggi dan rendah. Ketika metrik ini berada di luar jangkauan operasinya, operator peralatan harus diberi tahu sehingga mereka dapat memperbaiki masalah. Gunakan alarm untuk mengidentifikasi masalah dengan cepat dan memberi tahu operator untuk memaksimalkan kinerja peralatan dan proses Anda.

Topik

- [Jenis alarm](#)
- [Status alarm](#)
- [Properti status alarm](#)
- [Mendefinisikan alarm pada model aset](#)
- [Mengkonfigurasi alarm pada aset](#)
- [Menanggapi alarm](#)
- [Menelan status alarm eksternal](#)

Jenis alarm

Anda dapat menentukan alarm yang mendeteksi di AWS Cloud dan alarm yang Anda deteksi dengan proses eksternal. AWS IoT SiteWise mendukung jenis alarm berikut:

- AWS IoT Events alarm

AWS IoT Events Alarm adalah alarm yang mendeteksi. AWS IoT Events AWS IoT SiteWise mengirimkan nilai properti aset ke model alarm di AWS IoT Events. Kemudian, AWS IoT Events kirim status alarm ke AWS IoT SiteWise. Anda dapat mengonfigurasi opsi seperti saat alarm mendeteksi dan siapa yang harus diberi tahu saat status alarm berubah. Anda juga dapat menentukan [AWS IoT Events tindakan](#) yang terjadi ketika status alarm berubah.

Alarm AWS IoT Events dalam contoh model alarm. Model alarm menentukan ambang batas dan tingkat keparahan alarm, apa yang harus dilakukan ketika status alarm berubah, dan banyak lagi. Saat mengonfigurasi setiap sifat model alarm, Anda menentukan properti atribut dari model aset yang dipantau alarm. Semua aset berdasarkan model aset menggunakan nilai atribut saat AWS

IoT Events mengevaluasi sifat alarm tersebut. Untuk informasi selengkapnya, lihat [Menggunakan alarm](#) di Panduan AWS IoT Events Pengembang.

Anda dapat merespons AWS IoT Events alarm saat berubah status. Misalnya, Anda dapat mengenali atau menunda alarm saat alarm menjadi aktif. Anda juga dapat mengaktifkan, menonaktifkan, dan mengatur ulang alarm.

SiteWise Pengguna monitor dapat memvisualisasikan, mengonfigurasi, dan merespons AWS IoT Events alarm di portal SiteWise Monitor. Untuk informasi selengkapnya, lihat [Memantau dengan alarm](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.

Note

AWS IoT Events biaya berlaku untuk mengevaluasi alarm ini dan mentransfer data antara AWS IoT SiteWise dan AWS IoT Events. Untuk informasi selengkapnya, lihat [harga AWS IoT Events](#).

- Alarm eksternal

Alarm eksternal adalah alarm yang Anda evaluasi di luar. AWS IoT SiteWise Gunakan alarm eksternal jika Anda memiliki sumber data yang melaporkan status alarm. Alarm eksternal berisi properti pengukuran tempat Anda menelan data status alarm.

Anda tidak dapat mengakui atau menunda alarm eksternal saat berubah status.

SiteWise Pengguna monitor dapat melihat status alarm eksternal di portal SiteWise Monitor, tetapi mereka tidak dapat mengonfigurasi atau merespons alarm ini.

AWS IoT SiteWise tidak mengevaluasi keadaan alarm eksternal.

Status alarm

Alarm industri mencakup informasi tentang keadaan peralatan atau proses yang mereka pantau dan (opsional) informasi tentang respons operator terhadap status alarm.

Saat Anda menentukan AWS IoT Events alarm, Anda menentukan apakah akan mengaktifkan alur pengakuan atau tidak. Alur pengakuan diaktifkan secara default. Saat Anda mengaktifkan opsi ini, operator dapat mengenali alarm dan meninggalkan catatan dengan detail tentang alarm atau tindakan yang mereka ambil untuk mengatasinya. Jika operator tidak mengakui alarm aktif sebelum

menjadi tidak aktif, alarm menjadi terkunci. Status terkunci menunjukkan bahwa alarm menjadi aktif dan tidak dikenali, sehingga operator perlu memeriksa peralatan atau proses dan mengakui alarm yang terkunci.

Alarm memiliki status berikut:

- Normal (Normal) - Alarm diaktifkan tetapi tidak aktif. Proses atau peralatan industri beroperasi seperti yang diharapkan.
- Aktif (Active) — Alarm aktif. Proses atau peralatan industri berada di luar jangkauan operasinya dan perlu diperhatikan.
- Diakui (Acknowledged) — Operator mengakui status alarm.

Status ini hanya berlaku untuk alarm tempat Anda mengaktifkan aliran pengakuan.

- Latched (Latched) — Alarm kembali normal tetapi aktif dan tidak ada operator yang mengakuinya. Proses atau peralatan industri membutuhkan perhatian dari operator untuk mengatur ulang alarm ke normal.

Status ini hanya berlaku untuk alarm tempat Anda mengaktifkan aliran pengakuan.

- Snoozed (SnoozeDisabled) — Alarm dinonaktifkan karena operator menunda alarm. Operator menentukan durasi alarm tunda. Setelah durasi itu, alarm kembali ke keadaan normal.
- Disabled (Disabled) - Alarm dinonaktifkan dan tidak akan mendeteksi.

Properti status alarm

AWS IoT SiteWise menyimpan data status alarm sebagai objek JSON yang diserialisasikan ke string. Objek ini berisi status dan informasi tambahan tentang alarm, seperti tindakan respons operator dan aturan yang dievaluasi alarm.

Anda mengidentifikasi properti status alarm dengan nama dan jenis strukturnya, `AWS/ALARM_STATE`. Untuk informasi selengkapnya, lihat [Mendefinisikan alarm pada model aset](#).

Objek data status alarm berisi informasi berikut:

`stateName`

Status alarm. Untuk informasi selengkapnya, lihat [Status alarm](#).

Tipe data: STRING

customerAction

(Opsional) Objek yang berisi informasi tentang respons operator terhadap alarm. Operator dapat mengaktifkan, menonaktifkan, mengakui, dan menunda alarm. Ketika mereka melakukannya, data status alarm mencakup respons mereka dan catatan yang dapat mereka tinggalkan saat mereka merespons. Objek ini berisi informasi berikut:

actionName

Nama tindakan yang diambil operator untuk menanggapi alarm. Nilai ini berisi salah satu string berikut:

- ENABLE
- DISABLE
- SNOOZE
- ACKNOWLEDGE
- RESET

Tipe data: STRING

enable

(Opsional) Objek yang ada `customerAction` saat operator mengaktifkan alarm. Saat operator mengaktifkan alarm, status alarm berubah menjadi `Normal`. Objek ini berisi informasi berikut:

note

(Opsional) Catatan bahwa pelanggan pergi ketika mereka mengaktifkan alarm.

Tipe data: STRING

Panjang maksimal: 128 karakter

disable

(Opsional) Objek yang ada `customerAction` saat operator menonaktifkan alarm. Saat operator mengaktifkan alarm, status alarm berubah menjadi `Disabled`. Objek ini berisi informasi berikut:

note

(Opsional) Catatan bahwa pelanggan pergi ketika mereka menonaktifkan alarm.

Tipe data: STRING

Panjang maksimal: 128 karakter

acknowledge

(Opsional) Objek yang ada di `customerAction` saat operator mengakui alarm. Saat operator mengaktifkan alarm, status alarm berubah menjadi `Acknowledged`. Objek ini berisi informasi berikut:

note

(Opsional) Catatan bahwa pelanggan meninggalkan ketika mereka mengakui alarm.

Tipe data: STRING

Panjang maksimal: 128 karakter

snooze

(Opsional) Objek yang ada `customerAction` saat operator menunda alarm. Saat operator mengaktifkan alarm, status alarm berubah menjadi `SnoozeDisabled`. Objek ini berisi informasi berikut:

snoozeDuration

Durasi dalam hitungan detik operator menunda alarm. Alarm berubah menjadi `Normal` status setelah durasi ini.

Tipe data: INTEGER

note

(Opsional) Catatan bahwa pelanggan pergi ketika mereka menunda alarm.

Tipe data: STRING

Panjang maksimal: 128 karakter

ruleEvaluation

(Opsional) Objek yang berisi informasi tentang aturan yang mengevaluasi alarm. Objek ini berisi informasi berikut:

simpleRule

Objek yang berisi informasi tentang aturan sederhana, yang membandingkan nilai properti dengan nilai ambang batas dengan operator perbandingan. Objek ini berisi informasi berikut:

`inputProperty`

Nilai properti yang dievaluasi alarm ini.

Tipe data: DOUBLE

`operator`

Operator perbandingan yang digunakan alarm ini untuk membandingkan properti dengan ambang batas. Nilai ini berisi salah satu string berikut:

- `<`- Kurang dari
- `<=`- Kurang dari atau sama
- `=`— Sama
- `!=`- Tidak sama
- `>=`— Lebih besar dari atau sama
- `>`— Lebih besar dari

Tipe data: STRING

`threshold`

Nilai ambang batas yang dibandingkan dengan nilai properti alarm ini.

Tipe data: DOUBLE

Mendefinisikan alarm pada model aset

Model aset mendorong standardisasi data industri dan alarm Anda. Anda dapat menentukan definisi alarm pada model aset untuk membakukan alarm untuk semua aset berdasarkan model aset.

Anda menggunakan model aset komposit untuk menentukan alarm pada model aset. Model aset komposit adalah model aset yang menstandarisasi serangkaian properti tertentu pada model aset lain. Model aset komposit memastikan bahwa properti tertentu hadir pada model aset. Alarm memiliki tipe, status, dan properti sumber (opsional), sehingga model komposit alarm memberlakukan bahwa properti ini ada.

Setiap model aset komposit memiliki tipe yang mendefinisikan properti untuk model komposit tersebut. Model komposit alarm menentukan properti untuk jenis alarm, status alarm, dan sumber

alarm (opsional). Saat Anda membuat aset dari model aset dengan model komposit, aset tersebut menyertakan properti dari model komposit bersama properti yang Anda tentukan dalam model aset.

Setiap properti dalam model komposit harus memiliki nama yang mengidentifikasinya untuk jenis model komposit. Properti model komposit mendukung properti dengan tipe data yang kompleks. Properti ini memiliki tipe STRUCT data dan dataTypeSpec sifat yang menentukan tipe data kompleks properti. Properti tipe data yang kompleks berisi data JSON yang diserialkan sebagai string.

Model komposit alarm memiliki sifat berikut. Setiap properti harus memiliki nama yang mengidentifikasinya untuk jenis model komposit ini.

Jenis alarm

Jenis alarm. Tentukan satu dari yang berikut ini:

- **IOT_EVENTS**— AWS IoT Events Alarm. AWS IoT SiteWise mengirimkan data AWS IoT Events ke untuk mengevaluasi keadaan alarm ini. Anda harus menentukan properti sumber alarm untuk menentukan model AWS IoT Events alarm untuk definisi alarm ini.
- **EXTERNAL**— Alarm eksternal. Anda menelan keadaan alarm sebagai pengukuran.

Nama properti: `AWS/ALARM_TYPE`

Jenis properti: [atribut](#)

Tipe data: `STRING`

Status alarm

Data deret waktu untuk keadaan alarm. Ini adalah objek serial sebagai string yang berisi status dan informasi lain tentang alarm. Untuk informasi selengkapnya, lihat [Properti status alarm](#).

Nama properti: `AWS/ALARM_STATE`

Jenis properti: [pengukuran](#)

Tipe data: `STRUCT`

Jenis struktur data: `AWS/ALARM_STATE`

Sumber alarm

(Opsional) Nama Sumber Daya Amazon (ARN) dari sumber daya yang mengevaluasi status alarm. Untuk AWS IoT Events alarm, ini adalah ARN dari model alarm.

Nama properti: `AWS/ALARM_SOURCE`

Jenis properti: [atribut](#)

Tipe data: `STRING`

Example Contoh model komposit alarm

Model aset berikut mewakili boiler yang memiliki alarm untuk memantau suhunya. AWS IoT SiteWise mengirimkan data suhu AWS IoT Events untuk mendeteksi alarm.

```
{
  "assetModelName": "Boiler",
  "assetModelDescription": "A boiler that alarms when its temperature exceeds its
limit.",
  "assetModelProperties": [
    {
      "name": "Temperature",
      "dataType": "DOUBLE",
      "unit": "Celsius",
      "type": {
        "measurement": {}
      }
    },
    {
      "name": "High Temperature",
      "dataType": "DOUBLE",
      "unit": "Celsius",
      "type": {
        "attribute": {
          "defaultValue": "105.0"
        }
      }
    }
  ],
  "assetModelCompositeModels": [
    {
      "name": "BoilerTemperatureHighAlarm",
      "type": "AWS/ALARM",
      "properties": [
        {
          "name": "AWS/ALARM_TYPE",
          "dataType": "STRING",
```

```
    "type": {
      "attribute": {
        "defaultValue": "IOT_EVENTS"
      }
    },
    {
      "name": "AWS/ALARM_STATE",
      "dataType": "STRUCT",
      "dataTypeSpec": "AWS/ALARM_STATE",
      "type": {
        "measurement": {}
      }
    },
    {
      "name": "AWS/ALARM_SOURCE",
      "dataType": "STRING",
      "type": {
        "attribute": {}
      }
    }
  ]
}
```

Topik

- [Mendefinisikan alarm AWS IoT Events](#)
- [Mendefinisikan alarm eksternal](#)

Mendefinisikan alarm AWS IoT Events

Saat Anda membuat AWS IoT Events alarm, AWS IoT SiteWise kirimkan nilai properti aset AWS IoT Events ke untuk mengevaluasi status alarm. AWS IoT Events definisi alarm bergantung pada model alarm yang Anda tentukan AWS IoT Events. Untuk menentukan AWS IoT Events alarm pada model aset, Anda menentukan model komposit alarm yang menentukan model AWS IoT Events alarm sebagai properti sumber alarmnya.

AWS IoT Events alarm tergantung pada input seperti ambang alarm dan pengaturan pemberitahuan alarm. Anda mendefinisikan input ini sebagai atribut pada model aset. Anda kemudian dapat

menyesuaikan input ini pada setiap aset berdasarkan model. AWS IoT SiteWise Konsol dapat membuat atribut ini untuk Anda. Jika Anda mendefinisikan alarm dengan API AWS CLI atau, Anda harus menentukan atribut ini secara manual pada model aset.

Anda juga dapat menentukan tindakan lain yang terjadi ketika alarm Anda mendeteksi, seperti tindakan pemberitahuan alarm kustom. Misalnya, Anda dapat mengonfigurasi tindakan yang mengirimkan pemberitahuan push ke topik Amazon SNS. Untuk informasi selengkapnya tentang tindakan yang dapat Anda tentukan, lihat [Bekerja dengan AWS layanan lain](#) di Panduan AWS IoT Events Pengembang.

Saat Anda memperbarui atau menghapus model aset, AWS IoT SiteWise dapat memeriksa apakah model alarm di AWS IoT Events memantau properti aset yang terkait dengan model aset ini. Ini mencegah Anda menghapus properti aset yang saat ini digunakan AWS IoT Events alarm. Untuk mengaktifkan fitur ini AWS IoT SiteWise, Anda harus memiliki `iotevents:ListInputRoutings` izin. Izin ini memungkinkan AWS IoT SiteWise untuk melakukan panggilan ke operasi [ListInputRoutings](#) API yang didukung oleh AWS IoT Events. Untuk informasi selengkapnya, lihat [ListInputRoutings Izin \(Opsional\)](#).

Note

Fitur pemberitahuan alarm tidak tersedia di Wilayah China (Beijing).

Topik

- [Persyaratan untuk pemberitahuan alarm](#)
- [Mendefinisikan AWS IoT Events alarm \(AWS IoT SiteWise konsol\)](#)
- [Mendefinisikan AWS IoT Events alarm \(AWS IoT Events konsol\)](#)
- [Mendefinisikan AWS IoT Events alarm \(AWS CLI\)](#)

Persyaratan untuk pemberitahuan alarm

AWS IoT Events menggunakan AWS Lambda fungsi di AWS akun Anda untuk mengirim pemberitahuan alarm. Anda harus membuat fungsi Lambda ini di AWS Wilayah yang sama dengan alarm Anda untuk mengaktifkan pemberitahuan alarm. Fungsi Lambda ini menggunakan [Amazon Simple Notification Service \(Amazon SNS\)](#) untuk mengirim notifikasi teks dan [Amazon Simple Email Service \(Amazon SES\)](#) untuk mengirim pemberitahuan email. Saat Anda membuat AWS IoT

Events alarm, Anda mengonfigurasi protokol dan pengaturan yang digunakan alarm untuk mengirim pemberitahuan.

AWS IoT Events menyediakan template AWS CloudFormation tumpukan yang dapat Anda gunakan untuk membuat fungsi Lambda ini di akun Anda. Untuk informasi selengkapnya, lihat [Fungsi Lambda pemberitahuan alarm](#) di Panduan AWS IoT Events Pengembang.

Mendefinisikan AWS IoT Events alarm (AWS IoT SiteWise konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk menentukan AWS IoT Events alarm pada model aset yang ada. Untuk menentukan AWS IoT Events alarm pada model aset baru, buat model aset, lalu selesaikan langkah-langkah ini. Untuk informasi selengkapnya, lihat [Membuat model aset](#).

Important


Setiap alarm memerlukan atribut yang menentukan nilai ambang untuk dibandingkan dengan alarm. Anda harus menentukan atribut nilai ambang pada model aset sebelum Anda dapat menentukan alarm.

Pertimbangkan contoh di mana Anda ingin menentukan alarm yang mendeteksi ketika turbin angin melebihi peringkat kecepatan angin maksimum 50 mph. Sebelum Anda menentukan alarm, Anda harus menentukan atribut (Kecepatan angin maksimum) dengan nilai default 50.

Untuk menentukan AWS IoT Events alarm pada model aset


1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model.
3. Pilih model aset untuk menentukan alarm.
4. Pilih tab Alarm.
5. Pilih Tambahkan alarm.
6. Di bagian Opsi jenis alarm, pilih AWS IoT Events alarm.
7. Di bagian Detail alarm, lakukan hal berikut:
 - a. Masukkan nama untuk alarm Anda.
 - b. (Opsional) Masukkan deskripsi untuk alarm Anda.
8. Di bagian Definisi ambang batas, Anda menentukan kapan alarm mendeteksi dan tingkat keparahan alarm. Lakukan hal-hal berikut:

- a. Pilih Properti tempat alarm mendeteksi. Setiap kali properti ini menerima nilai baru, AWS IoT SiteWise mengirimkan nilai AWS IoT Events ke untuk mengevaluasi keadaan alarm.
 - b. Pilih Operator yang akan digunakan untuk membandingkan properti dengan nilai ambang batas. Pilih dari salah satu pilihan berikut:
 - < kurang dari
 - <= kurang dari atau sama
 - == sama
 - != tidak sama
 - >= lebih besar dari atau sama
 - > lebih besar dari
 - c. Untuk Nilai, pilih properti atribut yang akan digunakan sebagai nilai ambang batas. AWS IoT Events membandingkan nilai properti dengan nilai atribut ini.
 - d. Masukkan tingkat keparahan alarm. Gunakan nomor yang dipahami tim Anda untuk mencerminkan tingkat keparahan alarm ini.
9. (Opsional) Di pengaturan Pemberitahuan - bagian opsional, lakukan hal berikut:
- a. Pilih Aktif.

 Note

Jika Anda memilih Tidak Aktif, Anda dan tim Anda tidak akan menerima pemberitahuan alarm apa pun.


- b. Untuk Penerima, pilih penerima.

 Important

Anda dapat mengirim pemberitahuan alarm ke AWS IAM Identity Center pengguna. Untuk menggunakan fitur ini, Anda harus mengaktifkan IAM Identity Center. Anda hanya dapat mengaktifkan Pusat Identitas IAM di satu AWS Wilayah pada satu waktu. Ini berarti Anda dapat menentukan pemberitahuan alarm hanya di Wilayah tempat Anda mengaktifkan Pusat Identitas IAM. Untuk informasi selengkapnya, lihat [Memulai](#) di Panduan AWS IAM Identity Center Pengguna.


- c. Untuk Protokol, pilih dari opsi berikut:

- Email & teks - Alarm memberi tahu pengguna IAM Identity Center dengan pesan SMS dan pesan email.
 - Email — Alarm memberi tahu pengguna IAM Identity Center dengan pesan email.
 - Teks — Alarm memberi tahu pengguna IAM Identity Center dengan pesan SMS.
- d. Untuk Pengirim, pilih pengirim.

 Important


Anda harus memverifikasi alamat email pengirim di Amazon Simple Email Service (Amazon SES). Untuk informasi selengkapnya, lihat [Memverifikasi alamat email di Amazon SES](#), di Panduan Pengembang Layanan Email Sederhana Amazon.

10. Di bagian Status aset default, Anda dapat mengatur status default untuk alarm yang dibuat dari model aset ini.

 Note

Anda mengaktifkan atau menonaktifkan alarm ini untuk aset yang Anda buat dari model aset ini di langkah selanjutnya.

11. Di bagian Pengaturan lanjutan, Anda dapat mengonfigurasi izin, pengaturan pemberitahuan tambahan, tindakan status alarm, model alarm di SiteWise Monitor, dan aliran pengakuan.

 Note

AWS IoT Events alarm memerlukan peran layanan berikut:

- Peran yang AWS IoT Events mengasumsikan untuk mengirim nilai status alarm ke AWS IoT SiteWise.
- Peran yang AWS IoT Events mengasumsikan untuk mengirim data ke Lambda. Anda hanya perlu peran ini jika alarm Anda mengirimkan notifikasi.

Di bagian Izin, lakukan hal berikut:

- a. Untuk AWS IoT Events peran, gunakan peran yang ada atau buat peran dengan izin yang diperlukan. Peran ini membutuhkan `iotsitewise:BatchPutAssetPropertyValue`

izin dan hubungan kepercayaan yang memungkinkan `iotevents.amazonaws.com` untuk mengambil peran tersebut.

- b. Untuk peran AWS IoT Events Lambda, gunakan peran yang ada atau buat peran dengan izin yang diperlukan. Peran ini membutuhkan `lambda:InvokeFunction` dan `ss-directory:DescribeUser` izin dan hubungan kepercayaan yang memungkinkan `iotevents.amazonaws.com` untuk mengambil peran.

12. (Opsional) Di bagian Pengaturan pemberitahuan tambahan, lakukan hal berikut:

- a. Untuk atribut Penerima, Anda menentukan atribut yang nilainya menentukan penerima pemberitahuan. Anda dapat memilih pengguna IAM Identity Center sebagai penerima.

Anda dapat membuat atribut atau menggunakan atribut yang ada pada model aset.

- Jika Anda memilih Buat atribut penerima baru, tentukan nama atribut Penerima dan nilai default Penerima - opsional untuk atribut.
- Jika Anda memilih Gunakan atribut penerima yang ada, pilih atribut di Nama atribut Penerima. Alarm menggunakan nilai default atribut yang Anda pilih.

Anda dapat mengganti nilai default pada setiap aset yang Anda buat dari model aset ini.

- b. Untuk atribut pesan kustom, Anda menentukan atribut yang nilainya menentukan pesan kustom yang akan dikirim selain pesan perubahan status default. Misalnya, Anda dapat menentukan pesan yang membantu tim Anda memahami cara mengatasi alarm ini.

Anda dapat memilih untuk membuat atribut atau menggunakan atribut yang ada pada model aset.

- Jika Anda memilih untuk Membuat atribut pesan kustom baru, tentukan nama atribut pesan kustom dan Nilai default pesan kustom - opsional untuk atribut.
- Jika Anda memilih Gunakan atribut pesan kustom yang ada, pilih atribut di Nama atribut pesan kustom. Alarm menggunakan nilai default atribut yang Anda pilih.

Anda dapat mengganti nilai default pada setiap aset yang Anda buat dari model aset ini.

- c. Untuk Kelola fungsi Lambda Anda, lakukan salah satu hal berikut:

- Untuk AWS IoT SiteWise membuat fungsi Lambda baru, pilih Buat lambda baru dari templat yang dikelola AWS.

- Untuk menggunakan fungsi Lambda yang ada, pilih Gunakan lambda yang ada dan pilih nama fungsi.

Untuk informasi selengkapnya, lihat [Mengelola notifikasi alarm](#) di Panduan AWS IoT Events Pengembang.

13. (Opsional) Di bagian Set state action, lakukan hal berikut:

- a. Pilih Edit tindakan.
- b. Di bawah Tambahkan tindakan status alarm, tambahkan tindakan. dan pilih Simpan.

Anda dapat menambahkan hingga 10 tindakan.

AWS IoT Events dapat melakukan tindakan saat alarm aktif. Anda dapat menentukan tindakan bawaan untuk menggunakan timer atau mengatur variabel, atau mengirim data ke AWS sumber daya lain. Untuk informasi selengkapnya, lihat [Tindakan yang didukung](#) di Panduan AWS IoT Events Pengembang.

14. (Opsional) Di bawah Kelola model alarm di SiteWise Monitor - opsional, pilih Aktif atau Tidak Aktif.

Gunakan opsi ini sehingga Anda dapat memperbarui model alarm di SiteWise Monitorss. Pengaturan ini diaktifkan secara default.

15. Di bawah Akui aliran, pilih Aktif atau Tidak Aktif. Untuk informasi lebih lanjut tentang aliran pengakuan, lihat [Status alarm](#).

16. Pilih Tambahkan alarm.

Note

AWS IoT SiteWise Konsol membuat beberapa permintaan API untuk menambahkan alarm ke model aset. Saat Anda memilih Tambah alarm, konsol akan membuka kotak dialog yang menunjukkan kemajuan permintaan API ini. Tetap di halaman ini sampai setiap permintaan API berhasil atau sampai permintaan API gagal. Jika permintaan gagal, tutup kotak dialog, perbaiki masalah, dan pilih Tambahkan alarm untuk mencoba lagi.

Mendefinisikan AWS IoT Events alarm (AWS IoT Events konsol)

Anda dapat menggunakan AWS IoT Events konsol untuk menentukan AWS IoT Events alarm pada model aset yang ada. Untuk menentukan AWS IoT Events alarm pada model aset baru, buat model aset, lalu selesaikan langkah-langkah ini. Untuk informasi selengkapnya, lihat [Membuat model aset](#).


Important

Setiap alarm memerlukan atribut yang menentukan nilai ambang untuk dibandingkan dengan alarm. Anda harus menentukan atribut nilai ambang pada model aset sebelum Anda dapat menentukan alarm.

Pertimbangkan contoh di mana Anda ingin menentukan alarm yang mendeteksi ketika turbin angin melebihi peringkat kecepatan angin maksimum 50 mph. Sebelum Anda menentukan alarm, Anda harus menentukan atribut (Kecepatan angin maksimum) dengan nilai default 50.

Untuk menentukan AWS IoT Events alarm pada model aset

1. Navigasikan ke [konsol AWS IoT Events](#) tersebut.
2. Di panel navigasi, pilih Model alarm.
3. Pilih Buat model alarm.
4. Masukkan nama untuk alarm Anda.
5. (Opsional) Masukkan deskripsi untuk alarm Anda.
6. Di bagian Target alarm, lakukan hal berikut:
 - a. Untuk opsi Target, pilih properti AWS IoT SiteWise aset.
 - b. Pilih model aset yang ingin Anda tambahkan alarmnya.
7. Di bagian Definisi ambang batas, Anda menentukan kapan alarm mendeteksi dan tingkat keparahan alarm. Lakukan hal-hal berikut:
 - a. Pilih Properti tempat alarm mendeteksi. Setiap kali properti ini menerima nilai baru, AWS IoT SiteWise mengirimkan nilai AWS IoT Events ke untuk mengevaluasi keadaan alarm.
 - b. Pilih Operator yang akan digunakan untuk membandingkan properti dengan nilai ambang batas. Pilih dari salah satu pilihan berikut:
 - < kurang dari

- <= kurang dari atau sama
 - == sama
 - != tidak sama
 - >= lebih besar dari atau sama
 - > lebih besar dari
- c. Untuk Nilai, pilih properti atribut yang akan digunakan sebagai nilai ambang batas. AWS IoT Events membandingkan nilai properti dengan nilai atribut ini.
 - d. Masukkan tingkat keparahan alarm. Gunakan nomor yang dipahami tim Anda untuk mencerminkan tingkat keparahan alarm ini.
8. (Opsional) Di pengaturan Pemberitahuan - bagian opsional, lakukan hal berikut:
- a. Untuk Protokol, pilih dari opsi berikut:
 - Email & teks - Alarm memberi tahu pengguna IAM Identity Center dengan pesan SMS dan pesan email.
 - Email — Alarm memberi tahu pengguna IAM Identity Center dengan pesan email.
 - Teks — Alarm memberi tahu pengguna IAM Identity Center dengan pesan SMS.
 - b. Untuk Pengirim, pilih pengirim.
-  **Important**

Anda harus memverifikasi alamat email pengirim di Amazon Simple Email Service (Amazon SES). Untuk informasi selengkapnya, lihat [Memverifikasi alamat email di Amazon SES](#), di Panduan Pengembang Layanan Email Sederhana Amazon.
- c. Pilih atribut di atribut Penerima - opsional. Alarm menggunakan nilai default atribut yang Anda pilih.
 - d. Pilih atribut di atribut pesan kustom - opsional. Alarm menggunakan nilai default atribut yang Anda pilih.
9. Di bagian Instance, tentukan status Default untuk alarm ini. Anda dapat mengaktifkan atau menonaktifkan alarm ini untuk semua aset yang Anda buat dari model aset ini di langkah selanjutnya.
10. Dalam pengaturan Pengaturan lanjutan, Anda dapat mengonfigurasi izin, pengaturan pemberitahuan tambahan, tindakan status alarm, model alarm di SiteWise Monitor, dan aliran pengakuan.

Note

AWS IoT Events alarm memerlukan peran layanan berikut:

- Peran yang AWS IoT Events mengasumsikan untuk mengirim nilai status alarm ke AWS IoT SiteWise.
- Peran yang AWS IoT Events mengasumsikan untuk mengirim data ke Lambda. Anda hanya perlu peran ini jika alarm Anda mengirimkan notifikasi.

- Di bagian Akui alur, pilih Diaktifkan atau Dinonaktifkan. Untuk informasi lebih lanjut tentang aliran pengakuan, lihat [Status alarm](#).
- Di bagian Izin, lakukan hal berikut:
 - Untuk AWS IoT Events peran, gunakan peran yang ada atau buat peran dengan izin yang diperlukan. Peran ini membutuhkan `iotsitewise:BatchPutAssetPropertyValue` izin dan hubungan kepercayaan yang memungkinkan `iotevents.amazonaws.com` untuk mengambil peran tersebut.
 - Untuk peran Lambda, gunakan peran yang ada atau buat peran dengan izin yang diperlukan. Peran ini membutuhkan `lambda:InvokeFunction` dan `sso-directory:DescribeUser` izin dan hubungan kepercayaan yang memungkinkan `iotevents.amazonaws.com` untuk mengambil peran.
- (Opsional) Di panel Pengaturan pemberitahuan tambahan, lakukan hal berikut:
 - Untuk Kelola fungsi Lambda Anda, lakukan salah satu hal berikut:
 - Untuk AWS IoT Events membuat fungsi Lambda baru, pilih Buat fungsi Lambda baru.
 - Untuk menggunakan fungsi Lambda yang ada, pilih Gunakan fungsi Lambda yang ada dan pilih nama fungsi.

Untuk informasi selengkapnya, lihat [Mengelola notifikasi alarm](#) di Panduan AWS IoT Events Pengembang.

- (Opsional) Di bagian Set state action - opsional, lakukan hal berikut:
 - Di bawah Tindakan status alarm, tambahkan tindakan. dan pilih Simpan.

Anda dapat menambahkan hingga 10 tindakan.

AWS IoT Events dapat melakukan tindakan saat alarm aktif. Anda dapat menentukan tindakan bawaan untuk menggunakan timer atau mengatur variabel, atau mengirim data ke AWS sumber daya lain. Untuk informasi selengkapnya, lihat [Tindakan yang didukung](#) di Panduan AWS IoT Events Pengembang.

11. Pilih Buat.

Note

AWS IoT Events Konsol membuat beberapa permintaan API untuk menambahkan alarm ke model aset. Saat Anda memilih Tambah alarm, konsol akan membuka kotak dialog yang menunjukkan kemajuan permintaan API ini. Tetap di halaman ini sampai setiap permintaan API berhasil atau sampai permintaan API gagal. Jika permintaan gagal, tutup kotak dialog, perbaiki masalah, dan pilih Tambahkan alarm untuk mencoba lagi.

Mendefinisikan AWS IoT Events alarm ()AWS CLI

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk menentukan AWS IoT Events alarm yang memantau properti aset. Anda dapat menentukan alarm pada model aset baru atau yang sudah ada. Setelah Anda menentukan alarm pada model aset, Anda membuat alarm AWS IoT Events dan menghubungkannya ke model aset. Dalam proses ini, Anda melakukan hal berikut:

Langkah-langkah

- [Langkah 1: Mendefinisikan alarm pada model aset](#)
- [Langkah 2: Mendefinisikan model AWS IoT Events alarm](#)
- [Langkah 3: Mengaktifkan aliran data antara dan AWS IoT SiteWiseAWS IoT Events](#)

Langkah 1: Mendefinisikan alarm pada model aset

Tambahkan definisi alarm dan properti terkait ke model aset baru atau yang sudah ada.

Untuk menentukan alarm pada model aset (CLI)

1. Buat file bernama `asset-model-payload.json`. Ikuti langkah-langkah di bagian lain ini untuk menambahkan detail model aset Anda ke file, tetapi jangan kirimkan permintaan untuk membuat atau memperbarui model aset. Di bagian ini, Anda menambahkan definisi alarm ke detail model aset dalam `asset-model-payload.json` file.
 - Untuk informasi selengkapnya tentang cara membuat model aset, lihat [Membuat model aset \(AWS CLI\)](#).
 - Untuk informasi selengkapnya tentang cara memperbarui model aset yang ada, lihat [Memperbarui aset atau model komponen \(AWS CLI\)](#).

Note

Model aset Anda harus menentukan setidaknya satu properti aset, termasuk properti aset untuk dipantau dengan alarm.

2. Tambahkan model komposit alarm (`assetModelCompositeModels`) ke model aset. Model komposit AWS IoT Events alarm menentukan `IOT_EVENTS` jenis dan menentukan properti sumber alarm. Anda menambahkan properti sumber alarm setelah Anda membuat model alarm di AWS IoT Events.

Important

Model komposit alarm harus memiliki nama yang sama dengan model AWS IoT Events alarm yang Anda buat nanti. Nama model alarm hanya dapat berisi karakter alfanumerik. Tentukan nama alfanumerik yang unik sehingga Anda dapat menggunakan nama yang sama untuk model alarm.

```
{
  ...
  "assetModelCompositeModels": [
    {
      "name": "BoilerTemperatureHighAlarm",
      "type": "AWS/ALARM",
      "properties": [
        {
          "name": "AWS/ALARM_TYPE",
```

```

        "dataType": "STRING",
        "type": {
          "attribute": {
            "defaultValue": "IOT_EVENTS"
          }
        }
      },
      {
        "name": "AWS/ALARM_STATE",
        "dataType": "STRUCT",
        "dataTypeSpec": "AWS/ALARM_STATE",
        "type": {
          "measurement": {}
        }
      }
    ]
  }
}

```

3. Tambahkan atribut ambang alarm ke model aset. Tentukan nilai default yang akan digunakan untuk ambang batas ini. Anda dapat mengganti nilai default ini pada setiap aset berdasarkan model ini.

Note

Atribut ambang alarm harus berupa INTEGER atau aDOUBLE.

```

{
  ...
  "assetModelProperties": [
    ...
    {
      "name": "Temperature Max Threshold",
      "dataType": "DOUBLE",
      "type": {
        "attribute": {
          "defaultValue": "105.0"
        }
      }
    }
  ]
}

```

```
]
}
```

4. (Opsional) Tambahkan atribut notifikasi alarm ke model aset. Atribut ini menentukan penerima IAM Identity Center dan input lain yang AWS IoT Events digunakan untuk mengirim notifikasi saat alarm berubah status. Anda dapat mengganti default ini pada setiap aset berdasarkan model ini.

Important

Anda dapat mengirim pemberitahuan alarm ke AWS IAM Identity Center pengguna. Untuk menggunakan fitur ini, Anda harus mengaktifkan IAM Identity Center. Anda hanya dapat mengaktifkan Pusat Identitas IAM di satu AWS Wilayah pada satu waktu. Ini berarti Anda dapat menentukan pemberitahuan alarm hanya di Wilayah tempat Anda mengaktifkan Pusat Identitas IAM. Untuk informasi selengkapnya, lihat [Memulai](#) di Panduan AWS IAM Identity Center Pengguna.

Lakukan hal-hal berikut:

- a. Tambahkan atribut yang menentukan ID toko identitas Pusat Identitas IAM Anda. Anda dapat menggunakan operasi [ListInstances](#) API Pusat Identitas IAM untuk membuat daftar toko identitas Anda. Operasi ini hanya berfungsi di Wilayah tempat Anda mengaktifkan Pusat Identitas IAM.

```
aws sso-admin list-instances
```

Kemudian, tentukan ID penyimpanan identitas (misalnya, d-123EXAMPLE) sebagai nilai default untuk atribut.

```
{
  ...
  "assetModelProperties": [
    ...
    {
      "name": "identityStoreId",
      "dataType": "STRING",
      "type": {
        "attribute": {
```

```

        "defaultValue": "d-123EXAMPLE"
      }
    }
  ]
}

```

- b. Tambahkan atribut yang menentukan ID pengguna Pusat Identitas IAM yang menerima pemberitahuan. Untuk menentukan penerima notifikasi default, tambahkan ID pengguna IAM Identity Center sebagai nilai default. Lakukan salah satu hal berikut untuk mendapatkan ID pengguna IAM Identity Center:
 - i. Anda dapat menggunakan IAM Identity Center [ListUsersAPI](#) untuk mendapatkan ID pengguna yang nama penggunanya Anda kenal. Ganti *D-123Example* dengan ID toko identitas Anda, dan ganti *Nama* dengan nama pengguna pengguna.

```

aws identitystore list-users \
  --identity-store-id d-123EXAMPLE \
  --filters AttributePath=UserName,AttributeValue=Name

```

- ii. Gunakan [konsol Pusat Identitas IAM](#) untuk menelusuri pengguna Anda dan menemukan ID pengguna.

Kemudian, tentukan ID pengguna (misalnya,123EXAMPLE-a1b2c3d4-5678-90ab-cdef-33333EXAMPLE) sebagai nilai default untuk atribut, atau tentukan atribut tanpa nilai default.

```

{
  ...
  "assetModelProperties": [
    ...
    {
      "name": "userId",
      "dataType": "STRING",
      "type": {
        "attribute": {
          "defaultValue": "123EXAMPLE-a1b2c3d4-5678-90ab-cdef-33333EXAMPLE"
        }
      }
    }
  ]
}

```

```
}

```

- c. (Opsional) Tambahkan atribut yang menentukan ID pengirim default untuk pemberitahuan pesan SMS (teks). ID pengirim ditampilkan sebagai pengirim pesan pada pesan yang dikirim oleh Amazon Simple Notification Service (Amazon SNS). Untuk informasi selengkapnya, lihat [Meminta ID pengirim untuk pesan SMS dengan Amazon SNS](#) di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon.

```
{
  ...
  "assetModelProperties": [
    ...
    {
      "name": "senderId",
      "dataType": "STRING",
      "type": {
        "attribute": {
          "defaultValue": "MyFactory"
        }
      }
    }
  ]
}
```

- d. (Opsional) Tambahkan atribut yang menentukan alamat email default untuk digunakan sebagai alamat from dalam pemberitahuan email.

```
{
  ...
  "assetModelProperties": [
    ...
    {
      "name": "fromAddress",
      "dataType": "STRING",
      "type": {
        "attribute": {
          "defaultValue": "my.factory@example.com"
        }
      }
    }
  ]
}
```

- e. (Opsional) Tambahkan atribut yang menentukan subjek default untuk digunakan dalam pemberitahuan email.

```
{
  ...
  "assetModelProperties": [
    ...
    {
      "name": "emailSubject",
      "dataType": "STRING",
      "type": {
        "attribute": {
          "defaultValue": "[ALERT] High boiler temperature"
        }
      }
    }
  ]
}
```

- f. (Opsional) Tambahkan atribut yang menentukan pesan tambahan untuk disertakan dalam pemberitahuan. Secara default, pesan notifikasi menyertakan informasi tentang alarm. Anda juga dapat menyertakan pesan tambahan yang memberi pengguna lebih banyak informasi..

```
{
  ...
  "assetModelProperties": [
    ...
    {
      "name": "additionalMessage",
      "dataType": "STRING",
      "type": {
        "attribute": {
          "defaultValue": "Turn off the power before you check the alarm."
        }
      }
    }
  ]
}
```

5. Buat model aset atau perbarui model aset yang ada. Lakukan salah satu hal berikut:
- Untuk membuat model aset, jalankan perintah berikut.

```
aws iotsitewise create-asset-model --cli-input-json file://asset-model-payload.json
```

- Untuk memperbarui model aset yang ada, jalankan perintah berikut. Ganti *asset-model-id* dengan ID model aset.

```
aws iotsitewise update-asset-model \  
  --asset-model-id asset-model-id \  
  --cli-input-json file://asset-model-payload.json
```

Setelah Anda menjalankan perintah, perhatikan `assetModelId` dalam respon.

Contoh: Model aset boiler

Model aset berikut mewakili boiler yang melaporkan data suhu. Model aset ini mendefinisikan alarm yang mendeteksi ketika boiler terlalu panas.

```
{  
  "assetModelName": "Boiler Model",  
  "assetModelDescription": "Represents a boiler.",  
  "assetModelProperties": [  
    {  
      "name": "Temperature",  
      "dataType": "DOUBLE",  
      "unit": "C",  
      "type": {  
        "measurement": {}  
      }  
    },  
    {  
      "name": "Temperature Max Threshold",  
      "dataType": "DOUBLE",  
      "type": {  
        "attribute": {  
          "defaultValue": "105.0"  
        }  
      }  
    },  
    {  
      "name": "identityStoreId",  
      "dataType": "STRING",
```



```
"type": {
  "attribute": {
    "defaultValue": "d-123EXAMPLE"
  }
},
{
  "name": "userId",
  "dataType": "STRING",
  "type": {
    "attribute": {
      "defaultValue": "123EXAMPLE-a1b2c3d4-5678-90ab-cdef-33333EXAMPLE"
    }
  }
},
{
  "name": "senderId",
  "dataType": "STRING",
  "type": {
    "attribute": {
      "defaultValue": "MyFactory"
    }
  }
},
{
  "name": "fromAddress",
  "dataType": "STRING",
  "type": {
    "attribute": {
      "defaultValue": "my.factory@example.com"
    }
  }
},
{
  "name": "emailSubject",
  "dataType": "STRING",
  "type": {
    "attribute": {
      "defaultValue": "[ALERT] High boiler temperature"
    }
  }
},
{
  "name": "additionalMessage",
```

```

    "dataType": "STRING",
    "type": {
      "attribute": {
        "defaultValue": "Turn off the power before you check the alarm."
      }
    }
  ],
  "assetModelHierarchies": [

],
  "assetModelCompositeModels": [
    {
      "name": "BoilerTemperatureHighAlarm",
      "type": "AWS/ALARM",
      "properties": [
        {
          "name": "AWS/ALARM_TYPE",
          "dataType": "STRING",
          "type": {
            "attribute": {
              "defaultValue": "IOT_EVENTS"
            }
          }
        },
        {
          "name": "AWS/ALARM_STATE",
          "dataType": "STRUCT",
          "dataTypeSpec": "AWS/ALARM_STATE",
          "type": {
            "measurement": {}
          }
        }
      ]
    }
  ]
}

```

Langkah 2: Mendefinisikan model AWS IoT Events alarm

Buat model alarm di AWS IoT Events. Di AWS IoT Events, Anda menggunakan ekspresi untuk menentukan nilai dalam model alarm. Anda dapat menggunakan ekspresi untuk menentukan nilai dari AWS IoT SiteWise untuk mengevaluasi dan digunakan sebagai input ke alarm. Saat AWS IoT

SiteWise mengirim nilai properti aset ke model alarm, AWS IoT Events mengevaluasi ekspresi untuk mendapatkan nilai properti atau ID aset. Anda dapat menggunakan ekspresi berikut dalam model alarm:

- Nilai properti aset

Untuk mendapatkan nilai properti aset, gunakan ekspresi berikut. Ganti *assetModelId* dengan ID model aset dan ganti *propertyId* dengan ID properti.

```
$sitewise.assetModel.`assetModelId`.`propertyId`.propertyValue.value
```

- ID Aset

Untuk mendapatkan ID aset, gunakan ekspresi berikut. Ganti *assetModelId* dengan ID model aset dan ganti *propertyId* dengan ID properti.

```
$sitewise.assetModel.`assetModelId`.`propertyId`.assetId
```

Note

Saat Anda membuat model alarm, Anda dapat menentukan literal alih-alih ekspresi yang mengevaluasi AWS IoT SiteWise nilai. Ini dapat mengurangi jumlah atribut yang Anda tentukan pada model aset Anda. Namun, jika Anda mendefinisikan nilai sebagai literal, Anda tidak dapat menyesuaikan nilai tersebut pada aset berdasarkan model aset. AWS IoT SiteWise Monitor Pengguna Anda juga tidak dapat menyesuaikan alarm, karena mereka dapat mengonfigurasi pengaturan alarm hanya pada aset.

Untuk membuat model AWS IoT Events alarm (CLI)

1. Saat Anda membuat model alarm AWS IoT Events, Anda harus menentukan ID dari setiap properti yang digunakan alarm, yang mencakup yang berikut ini:
 - Properti status alarm dalam model aset komposit
 - Properti yang dipantau alarm
 - Atribut ambang
 - (Opsional) Atribut ID penyimpanan identitas Pusat Identitas IAM


- (Opsional) Atribut ID pengguna IAM Identity Center
- (Opsional) Atribut ID pengirim SMS
- (Opsional) Atribut email dari alamat
- (Opsional) Atribut subjek email
- (Opsional) Atribut pesan tambahan

Jalankan perintah berikut untuk mengambil ID properti ini pada model aset. Ganti *asset-model-id* dengan ID model aset dari langkah sebelumnya.

```
aws iotsitewise describe-asset-model --asset-model-id asset-model-id
```

Operasi mengembalikan respons yang berisi detail model aset. Perhatikan ID setiap properti yang digunakan alarm. Anda menggunakan ID ini saat membuat model AWS IoT Events alarm di langkah berikutnya.

2. Buat model alarm di AWS IoT Events. Lakukan hal-hal berikut:
 - a. Buat file bernama `alarm-model-payload.json`.
 - b. Salin objek JSON berikut ke dalam file.
 - c. Masukkan nama (`alarmModelName`), deskripsi (`alarmModelDescription`), dan tingkat keparahan (`severity`) untuk alarm Anda. Untuk tingkat keparahan, tentukan bilangan bulat yang mencerminkan tingkat keparahan perusahaan Anda.

 Important


Model alarm harus memiliki nama yang sama dengan model komposit alarm yang Anda tentukan pada model aset Anda sebelumnya.
Nama model alarm hanya dapat berisi karakter alfanumerik.

```
{  
  "alarmModelName": "BoilerTemperatureHighAlarm",  
  "alarmModelDescription": "Detects when the boiler temperature is high.",  
  "severity": 3  
}
```

- d. Tambahkan aturan perbandingan (`alarmRule`) ke alarm. Aturan ini mendefinisikan properti untuk monitor (`inputProperty`), nilai ambang untuk membandingkan (`threshold`), dan operator perbandingan untuk menggunakan (`comparisonOperator`).
- Ganti `assetModelId` dengan ID model aset.
 - Ganti `alarmPropertyId` dengan ID properti yang dipantau alarm.
 - Ganti `thresholdAttributeId` dengan ID properti atribut threshold.
 - Ganti `GREATER` dengan operator yang akan digunakan untuk membandingkan nilai properti dengan ambang batas. Pilih dari salah satu pilihan berikut:
 - LESS
 - LESS_OR_EQUAL
 - EQUAL
 - NOT_EQUAL
 - GREATER_OR_EQUAL
 - GREATER

```
{
  "alarmModelName": "BoilerTemperatureHighAlarm",
  "alarmModelDescription": "Detects when the boiler temperature is high.",
  "severity": 3,
  "alarmRule": {
    "simpleRule": {
      "inputProperty":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.propertyValue.value",
      "comparisonOperator": "GREATER",
      "threshold":
"$sitewise.assetModel.`assetModelId`.`thresholdAttributeId`.propertyValue.value"
    }
  }
}
```

- e. Tambahkan action (`alarmEventActions`) untuk mengirim status alarm ke AWS IoT SiteWise saat alarm berubah status.

 Note

Untuk konfigurasi lanjutan, Anda dapat menentukan tindakan tambahan yang harus dilakukan saat alarm berubah status. Misalnya, Anda mungkin memanggil AWS Lambda fungsi atau memublikasikan ke topik MQTT. Untuk informasi selengkapnya, lihat [Bekerja dengan AWS layanan lain](#) di Panduan AWS IoT Events Pengembang.

- Ganti *assetModelId* dengan ID model aset.
- Ganti *alarmPropertyId* dengan ID properti yang dipantau alarm.
- Ganti *alarmStatePropertyId* dengan ID properti status alarm dalam model komposit alarm.

```
{
  "alarmModelName": "BoilerTemperatureHighAlarm",
  "alarmModelDescription": "Detects when the boiler temperature is high.",
  "severity": 3,
  "alarmRule": {
    "simpleRule": {
      "inputProperty":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.propertyValue.value",
      "comparisonOperator": "GREATER",
      "threshold":
"$sitewise.assetModel.`assetModelId`.`thresholdAttributeId`.propertyValue.value"
    }
  },
  "alarmEventActions": {
    "alarmActions": [
      {
        "iotSiteWise": {
          "assetId":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.assetId",
          "propertyId": "'alarmStatePropertyId'"
        }
      }
    ]
  }
}
```

- f. (Opsional) Konfigurasi pengaturan pemberitahuan alarm. Tindakan pemberitahuan alarm menggunakan fungsi Lambda di akun Anda untuk mengirim pemberitahuan alarm. Untuk informasi selengkapnya, lihat [Persyaratan untuk pemberitahuan alarm](#). Dalam pengaturan pemberitahuan alarm, Anda dapat mengonfigurasi notifikasi SMS dan email untuk dikirim ke pengguna IAM Identity Center. Lakukan hal-hal berikut:
- i. Tambahkan konfigurasi notifikasi alarm (`alarmNotification`) ke payload `alarm-model-payload.json`.
- Ganti `alarmNotificationFunctionARN` dengan ARN dari fungsi Lambda yang menangani notifikasi alarm.

```
{
  "alarmModelName": "BoilerTemperatureHighAlarm",
  "alarmModelDescription": "Detects when the boiler temperature is high.",
  "severity": 3,
  "alarmRule": {
    "simpleRule": {
      "inputProperty":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.propertyValue.value",
      "comparisonOperator": "GREATER",
      "threshold":
"$sitewise.assetModel.`assetModelId`.`thresholdAttributeId`.propertyValue.value"
    }
  },
  "alarmEventActions": {
    "alarmActions": [
      {
        "iotSiteWise": {
          "assetId":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.assetId",
          "propertyId": "'alarmStatePropertyId'"
        }
      }
    ]
  },
  "alarmNotification": {
    "notificationActions": [
      {
        "action": {
          "lambdaAction": {
```

```

        "functionArn": "alarmNotificationFunctionArn"
    }
}
]
}
}

```

- ii. (Opsional) Konfigurasi notifikasi SMS (`smsConfigurations`) untuk dikirim ke pengguna Pusat Identitas IAM saat alarm berubah status.
 - Ganti `identityStoreIdAttributeId` dengan ID atribut yang berisi ID penyimpanan identitas Pusat Identitas IAM.
 - Ganti `userIdAttributeId` dengan ID atribut yang berisi ID pengguna IAM Identity Center.
 - Ganti `senderIdAttributeId` dengan ID atribut yang berisi ID pengirim Amazon SNS, atau hapus `senderId` dari payload.
 - Ganti `additionalMessageAttributeId` dengan ID atribut yang berisi pesan tambahan, atau hapus `additionalMessage` dari payload.

```

{
  "alarmModelName": "BoilerTemperatureHighAlarm",
  "alarmModelDescription": "Detects when the boiler temperature is high.",
  "severity": 3,
  "alarmRule": {
    "simpleRule": {
      "inputProperty":
        "$sitewise.assetModel.`assetModelId`.`alarmPropertyId` .propertyValue.value",
      "comparisonOperator": "GREATER",
      "threshold":
        "$sitewise.assetModel.`assetModelId`.`thresholdAttributeId` .propertyValue.value"
    }
  },
  "alarmEventActions": {
    "alarmActions": [
      {
        "iotSiteWise": {
          "assetId":
            "$sitewise.assetModel.`assetModelId`.`alarmPropertyId` .assetId",
          "propertyId": "'alarmStatePropertyId'"
        }
      }
    ]
  }
}

```



```
    }
  }
]
},
"alarmNotification": {
  "notificationActions": [
    {
      "action": {
        "lambdaAction": {
          "functionArn": "alarmNotificationFunctionArn"
        }
      },
      "smsConfigurations": [
        {
          "recipients": [
            {
              "ssoIdentity": {
                "identityStoreId":
"$$$sitewise.assetModel.`assetModelId`.`identityStoreIdAttributeId`.propertyValue.value",
                "userId":
"$$$sitewise.assetModel.`assetModelId`.`userIdAttributeId`.propertyValue.value"
              }
            }
          ],
          "senderId":
"$$$sitewise.assetModel.`assetModelId`.`senderIdAttributeId`.propertyValue.value",
          "additionalMessage":
"$$$sitewise.assetModel.`assetModelId`.`additionalMessageAttributeId`.propertyValue."
        }
      ]
    }
  ]
}
}
```

- iii. (Opsional) Konfigurasi notifikasi email (`emailConfigurations`) untuk dikirim ke pengguna Pusat Identitas IAM saat alarm berubah status.
 - Ganti *identityStoreIdAttributeId* dengan ID properti atribut ID penyimpanan identitas Pusat Identitas IAM.
 - Ganti Id dengan *userIdAttributeID* properti atribut ID pengguna IAM Identity Center.

- Ganti *fromAddressAttributeId* dengan ID properti atribut alamat “dari”, atau hapus from dari muatan.
- Ganti *emailSubjectAttributeId* dengan ID properti atribut subjek email, atau hapus subject dari payload.
- Ganti *additionalMessageAttributeId* dengan ID properti atribut pesan tambahan, atau hapus additionalMessage dari payload.

```
{
  "alarmModelName": "BoilerTemperatureHighAlarm",
  "alarmModelDescription": "Detects when the boiler temperature is high.",
  "severity": 3,
  "alarmRule": {
    "simpleRule": {
      "inputProperty":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.propertyValue.value",
      "comparisonOperator": "GREATER",
      "threshold":
"$sitewise.assetModel.`assetModelId`.`thresholdAttributeId`.propertyValue.value"
    }
  },
  "alarmEventActions": {
    "alarmActions": [
      {
        "iotSiteWise": {
          "assetId":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.assetId",
          "propertyId": "'alarmStatePropertyId'"
        }
      }
    ]
  },
  "alarmNotification": {
    "notificationActions": [
      {
        "action": {
          "lambdaAction": {
            "functionArn": "alarmNotificationFunctionArn"
          }
        }
      }
    ],
    "smsConfigurations": [
      {
```

```

    "recipients": [
      {
        "ssoIdentity": {
          "identityStoreId":
"$sitewise.assetModel.`assetModelId`.`identityStoreIdAttributeId`.propertyValue.value"
          "userId":
"$sitewise.assetModel.`assetModelId`.`userIdAttributeId`.propertyValue.value"
        }
      }
    ],
    "senderId":
"$sitewise.assetModel.`assetModelId`.`senderIdAttributeId`.propertyValue.value",
    "additionalMessage":
"$sitewise.assetModel.`assetModelId`.`additionalMessageAttributeId`.propertyValue.value"
  }
],
  "emailConfigurations": [
    {
      "from":
"$sitewise.assetModel.`assetModelId`.`fromAddressAttributeId`.propertyValue.value"
      "recipients": {
        "to": [
          {
            "ssoIdentity": {
              "identityStoreId":
"$sitewise.assetModel.`assetModelId`.`identityStoreIdAttributeId`.propertyValue.value"
              "userId":
"$sitewise.assetModel.`assetModelId`.`userIdAttributeId`.propertyValue.value"
            }
          }
        ]
      },
      "content": {
        "subject":
"$sitewise.assetModel.`assetModelId`.`emailSubjectAttributeId`.propertyValue.value"
        "additionalMessage":
"$sitewise.assetModel.`assetModelId`.`additionalMessageAttributeId`.propertyValue.value"
      }
    }
  ]
}

```

}

- g. (Opsional) Tambahkan kemampuan alarm (`alarmCapabilities`) ke payload `diAlarm-model-payload.json`. Dalam objek ini, Anda dapat menentukan apakah alur pengakuan diaktifkan dan status aktifkan default untuk aset berdasarkan model aset. Untuk informasi lebih lanjut tentang aliran pengakuan, lihat [Status alarm](#).

```
{
  "alarmModelName": "BoilerTemperatureHighAlarm",
  "alarmModelDescription": "Detects when the boiler temperature is high.",
  "severity": 3,
  "alarmRule": {
    "simpleRule": {
      "inputProperty":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.propertyValue.value",
      "comparisonOperator": "GREATER",
      "threshold":
"$sitewise.assetModel.`assetModelId`.`thresholdAttributeId`.propertyValue.value"
    }
  },
  "alarmEventActions": {
    "alarmActions": [
      {
        "iotSiteWise": {
          "assetId":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.assetId",
          "propertyId": "'alarmStatePropertyId'"
        }
      }
    ]
  },
  "alarmNotification": {
    "notificationActions": [
      {
        "action": {
          "lambdaAction": {
            "functionArn": "alarmNotificationFunctionArn"
          }
        }
      },
      {
        "smsConfigurations": [
          {
            "recipients": [

```

```

        "ssoIdentity": {
            "identityStoreId":
"$sitewise.assetModel.`assetModelId`.`identityStoreIdAttributeId`.propertyValue.value"
            "userId":
"$sitewise.assetModel.`assetModelId`.`userIdAttributeId`.propertyValue.value"
        }
    ],
    "senderId":
"$sitewise.assetModel.`assetModelId`.`senderIdAttributeId`.propertyValue.value",
    "additionalMessage":
"$sitewise.assetModel.`assetModelId`.`additionalMessageAttributeId`.propertyValue.value"
},
    "emailConfigurations": [
        {
            "from":
"$sitewise.assetModel.`assetModelId`.`fromAddressAttributeId`.propertyValue.value",
            "recipients": {
                "to": [
                    {
                        "ssoIdentity": {
                            "identityStoreId":
"$sitewise.assetModel.`assetModelId`.`identityStoreIdAttributeId`.propertyValue.value"
                            "userId":
"$sitewise.assetModel.`assetModelId`.`userIdAttributeId`.propertyValue.value"
                        }
                    }
                ]
            },
            "content": {
                "subject":
"$sitewise.assetModel.`assetModelId`.`emailSubjectAttributeId`.propertyValue.value",
                "additionalMessage":
"$sitewise.assetModel.`assetModelId`.`additionalMessageAttributeId`.propertyValue.value"
            }
        }
    ]
},
    "alarmCapabilities": {
        "initializationConfiguration": {
            "disabledOnInitialization": false
        }
    }
}

```

```

    },
    "acknowledgeFlow": {
      "enabled": true
    }
  }
}

```

- h. Tambahkan peran layanan IAM (`roleArn`) yang AWS IoT Events dapat mengasumsikan untuk mengirim data ke AWS IoT SiteWise. Peran ini membutuhkan `iotsitewise:BatchPutAssetPropertyValue` izin dan hubungan kepercayaan yang memungkinkan `iotevents.amazonaws.com` untuk mengambil peran. Untuk mengirim notifikasi, peran ini juga memerlukan `sso-directory:DescribeUser` izin `lambda:InvokeFunction` dan. Untuk informasi selengkapnya, lihat [Peran layanan alarm](#) di Panduan AWS IoT Events Pengembang.
- Ganti `roleArn` dengan ARN dari peran yang AWS IoT Events dapat diasumsikan untuk melakukan tindakan ini.

```

{
  "alarmModelName": "BoilerTemperatureHighAlarm",
  "alarmModelDescription": "Detects when the boiler temperature is high.",
  "severity": 3,
  "alarmRule": {
    "simpleRule": {
      "inputProperty":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.propertyValue.value",
      "comparisonOperator": "GREATER",
      "threshold":
"$sitewise.assetModel.`assetModelId`.`thresholdAttributeId`.propertyValue.value"
    }
  },
  "alarmEventActions": [
    {
      "alarmActions": [
        {
          "iotSiteWise": {
            "assetId":
"$sitewise.assetModel.`assetModelId`.`alarmPropertyId`.assetId",
            "propertyId": "'alarmStatePropertyId'"
          }
        }
      ]
    }
  ]
}

```

```

    },
    "alarmNotification": {
      "notificationActions": [
        {
          "action": {
            "lambdaAction": {
              "functionArn": "alarmNotificationFunctionArn"
            }
          },
        },
        "smsConfigurations": [
          {
            "recipients": [
              {
                "ssoIdentity": {
                  "identityStoreId":
"$sitewise.assetModel.`assetModelId`.`identityStoreIdAttributeId` .propertyValue.value"
                "userId":
"$sitewise.assetModel.`assetModelId`.`userIdAttributeId` .propertyValue.value"
              }
            ],
            "senderId":
"$sitewise.assetModel.`assetModelId`.`senderIdAttributeId` .propertyValue.value",
            "additionalMessage":
"$sitewise.assetModel.`assetModelId`.`additionalMessageAttributeId` .propertyValue.value"
          }
        ],
        "emailConfigurations": [
          {
            "from":
"$sitewise.assetModel.`assetModelId`.`fromAddressAttributeId` .propertyValue.value",
            "recipients": {
              "to": [
                {
                  "ssoIdentity": {
                    "identityStoreId":
"$sitewise.assetModel.`assetModelId`.`identityStoreIdAttributeId` .propertyValue.value"
                    "userId":
"$sitewise.assetModel.`assetModelId`.`userIdAttributeId` .propertyValue.value"
                  }
                }
              ]
            },
            "content": {

```

```

        "subject":
"$sitewise.assetModel.`assetModelId`.`emailSubjectAttributeId`.propertyValue.value",
        "additionalMessage":
"$sitewise.assetModel.`assetModelId`.`additionalMessageAttributeId`.propertyValue.value"
    }
}
]
}
],
},
"alarmCapabilities": {
  "initializationConfiguration": {
    "disabledOnInitialization": false
  },
  "acknowledgeFlow": {
    "enabled": false
  }
},
"roleArn": "arn:aws:iam::123456789012:role/MyIoTEventsAlarmRole"
}

```

- i. Jalankan perintah berikut untuk membuat model AWS IoT Events alarm dari payload `dialarm-model-payload.json`.

```
aws iotevents create-alarm-model --cli-input-json file://alarm-model-payload.json
```

- j. Operasi mengembalikan respons yang mencakup ARN dari model alarm, `alarmModelArn`. Salin ARN ini untuk mengatur definisi alarm pada model aset Anda di langkah berikutnya.

Langkah 3: Mengaktifkan aliran data antara dan AWS IoT SiteWise dan AWS IoT Events

Setelah Anda membuat sumber daya yang diperlukan di AWS IoT SiteWise dan AWS IoT Events, Anda dapat mengaktifkan aliran data antara sumber daya untuk mengaktifkan alarm Anda. Di bagian ini, Anda memperbarui definisi alarm dalam model aset untuk menggunakan model alarm yang Anda buat pada langkah sebelumnya.

Untuk mengaktifkan aliran data antara AWS IoT SiteWise dan AWS IoT Events (CLI)

- Atur model alarm sebagai sumber alarm dalam model aset. Lakukan hal-hal berikut:

- a. Jalankan perintah berikut untuk mengambil definisi model aset yang ada. Ganti *asset-model-id* dengan ID model aset.

```
aws iotsitewise describe-asset-model --asset-model-id asset-model-id
```

Operasi mengembalikan respons yang berisi detail model aset.

- b. Buat file bernama `update-asset-model-payload.json` dan salin respons perintah sebelumnya ke dalam file.
- c. Hapus pasangan kunci-nilai berikut dari file: `update-asset-model-payload.json`
 - `assetModelId`
 - `assetModelArn`
 - `assetModelCreationDate`
 - `assetModelLastUpdateDate`
 - `assetModelStatus`
- d. Tambahkan properti sumber alarm (`AWS/ALARM_SOURCE`) ke model komposit alarm yang Anda tentukan sebelumnya. Ganti *alarmModelArn* dengan ARN model alarm, yang menetapkan nilai properti sumber alarm.

```
{
  ...
  "assetModelCompositeModels": [
    ...
    {
      "name": "BoilerTemperatureHighAlarm",
      "type": "AWS/ALARM",
      "properties": [
        {
          "id": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
          "name": "AWS/ALARM_TYPE",
          "dataType": "STRING",
          "type": {
            "attribute": {
              "defaultValue": "IOT_EVENTS"
            }
          }
        },
        {

```

```

    "id": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
    "name": "AWS/ALARM_STATE",
    "dataType": "STRUCT",
    "dataTypeSpec": "AWS/ALARM_STATE",
    "type": {
      "measurement": {}
    }
  },
  {
    "name": "AWS/ALARM_SOURCE",
    "dataType": "STRING",
    "type": {
      "attribute": {
        "defaultValue": "alarmModelArn"
      }
    }
  }
]
}
]
}

```

- e. Jalankan perintah berikut untuk memperbarui model aset dengan definisi yang disimpan dalam `update-asset-model-payload.json` file. Ganti *asset-model-id* dengan ID model aset.

```

aws iotsitewise update-asset-model \
  --asset-model-id asset-model-id \
  --cli-input-json file://update-asset-model-payload.json

```

Model aset Anda sekarang mendefinisikan alarm yang mendeteksi. AWS IoT Events Alarm memantau properti target di semua aset berdasarkan model aset ini. Anda dapat mengonfigurasi alarm pada setiap aset untuk menyesuaikan properti seperti ambang batas atau penerima Pusat Identitas IAM untuk setiap aset. Untuk informasi selengkapnya, lihat [Mengkonfigurasi alarm pada aset](#).

Mendefinisikan alarm eksternal

Alarm eksternal berisi status alarm yang Anda deteksi di luar. AWS IoT SiteWise

Mendefinisikan alarm eksternal (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk menentukan alarm eksternal pada model aset yang ada. Untuk menentukan alarm eksternal pada model aset baru, buat model aset, lalu selesaikan langkah-langkah ini. Untuk informasi selengkapnya, lihat [Membuat model aset](#).

Untuk menentukan alarm pada model aset

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model.
3. Pilih model aset untuk menentukan alarm.
4. Pilih tab Definisi alarm.
5. Pilih Tambahkan alarm.
6. Di Opsi jenis alarm, pilih Alarm eksternal.
7. Masukkan nama untuk alarm Anda.
8. (Opsional) Masukkan deskripsi untuk alarm Anda.
9. Pilih Tambahkan alarm.

Mendefinisikan alarm eksternal (CLI)

Anda dapat menggunakan AWS CLI untuk menentukan alarm eksternal pada model aset baru atau yang sudah ada.

Untuk menambahkan alarm eksternal ke model aset, Anda menambahkan model komposit alarm ke model aset. Model komposit alarm eksternal menentukan EXTERNAL jenis dan tidak menentukan properti sumber alarm. Contoh berikut alarm komposit mendefinisikan alarm suhu eksternal.

```
{
  ...
  "assetModelCompositeModels": [
    {
      "name": "BoilerTemperatureHighAlarm",
      "type": "AWS/ALARM",
      "properties": [
        {
          "name": "AWS/ALARM_TYPE",
          "dataType": "STRING",
```

```
    "type": {
      "attribute": {
        "defaultValue": "EXTERNAL"
      }
    },
    {
      "name": "AWS/ALARM_STATE",
      "dataType": "STRUCT",
      "dataTypeSpec": "AWS/ALARM_STATE",
      "type": {
        "measurement": {}
      }
    }
  ]
}
```

Untuk informasi selengkapnya tentang cara menambahkan model komposit ke model aset baru atau yang sudah ada, lihat berikut ini:

- [Membuat model aset \(AWS CLI\)](#)
- [Memperbarui aset atau model komponen \(AWS CLI\)](#)

Setelah Anda menentukan alarm eksternal, Anda dapat menelan status alarm ke aset berdasarkan model aset. Untuk informasi selengkapnya, lihat [Menelan status alarm eksternal](#).

Mengkonfigurasi alarm pada aset

Setelah menentukan AWS IoT Events alarm pada model aset, Anda dapat mengonfigurasi alarm pada setiap aset berdasarkan model aset. Anda dapat mengedit nilai ambang batas dan pengaturan notifikasi untuk alarm. Masing-masing nilai ini adalah atribut pada aset, sehingga Anda dapat memperbarui nilai default atribut untuk mengonfigurasi nilai-nilai ini.

Note

Anda dapat mengonfigurasi nilai-nilai ini untuk AWS IoT Events alarm, tetapi tidak pada alarm eksternal.

Topik

- [Mengkonfigurasi nilai ambang \(konsol\)](#)
- [Mengkonfigurasi nilai ambang \(\)AWS CLI](#)
- [Mengkonfigurasi pengaturan notifikasi \(konsol\)](#)
- [Mengkonfigurasi pengaturan notifikasi \(CLI\)](#)

Mengkonfigurasi nilai ambang (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk memperbarui nilai atribut yang menentukan nilai ambang alarm.

Untuk memperbarui nilai ambang alarm (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang ingin Anda perbarui nilai ambang alarm.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih Edit.
5. Temukan atribut yang digunakan alarm untuk nilai ambangnya, lalu masukkan nilai barunya.
6. Pilih Simpan.

Mengkonfigurasi nilai ambang ()AWS CLI

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk memperbarui nilai atribut yang menentukan nilai ambang alarm.

Anda harus mengetahui aset `assetId` dan properti Anda `propertyId` untuk menyelesaikan prosedur ini. Anda juga dapat menggunakan ID eksternal. Jika Anda membuat aset dan tidak mengetahuinya `assetId`, gunakan [ListAssets](#) API untuk mencantumkan semua aset untuk model tertentu. Gunakan [DescribeAsset](#) operasi untuk melihat properti aset Anda termasuk ID properti.

Gunakan [BatchPutAssetPropertyValue](#) operasi untuk menetapkan nilai atribut ke aset Anda. Anda dapat menggunakan operasi ini untuk mengatur beberapa atribut sekaligus. Payload operasi ini berisi daftar entri, dan setiap entri berisi ID aset, ID properti, dan nilai atribut.

Untuk memperbarui nilai atribut (AWS CLI)

1. Buat file bernama `batch-put-payload.json` dan salin objek JSON berikut ke dalam file. Contoh payload ini menunjukkan cara mengatur lintang dan bujur turbin angin. Perbarui ID, nilai, dan stempel waktu untuk mengubah payload untuk kasus penggunaan Anda.

```
{
  "entries": [
    {
      "entryId": "windfarm3-turbine7-latitude",
      "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "propertyId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "propertyValues": [
        {
          "value": {
            "doubleValue": 47.6204
          },
          "timestamp": {
            "timeInSeconds": 1575691200
          }
        }
      ]
    },
    {
      "entryId": "windfarm3-turbine7-longitude",
      "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "propertyId": "a1b2c3d4-5678-90ab-cdef-55555EXAMPLE",
      "propertyValues": [
        {
          "value": {
            "doubleValue": 122.3491
          },
          "timestamp": {
            "timeInSeconds": 1575691200
          }
        }
      ]
    }
  ]
}
```

```
}
```

- Setiap entri dalam payload berisi `entryId` yang dapat Anda definisikan sebagai string unik apa pun. Jika ada entri permintaan yang gagal, setiap kesalahan akan berisi permintaan `entryId` yang sesuai sehingga Anda tahu permintaan mana yang harus dicoba lagi.
- Untuk menetapkan nilai atribut, Anda dapat menyertakan satu timestamp-quality-value (TQV) struktur dalam daftar `propertyValues` untuk setiap properti atribut. Struktur ini harus berisi yang baru `value` dan saat `initimestamp`.
 - `value`— Struktur yang berisi salah satu bidang berikut, tergantung pada jenis properti yang ditetapkan:
 - `booleanValue`
 - `doubleValue`
 - `integerValue`
 - `stringValue`
 - `timestamp`— Struktur yang berisi waktu epoch Unix saat ini dalam hitungan detik, `timeInSeconds` AWS IoT SiteWise menolak setiap titik data dengan stempel waktu yang ada lebih dari 7 hari di masa lalu atau lebih baru dari 5 menit di masa depan.

Untuk informasi lebih lanjut tentang cara menyiapkan muatan [BatchPutAssetPropertyValue](#), lihat [Menyerap data menggunakan API AWS IoT SiteWise](#).

2. Jalankan perintah berikut untuk mengirim nilai atribut ke AWS IoT SiteWise:

```
aws iotsitewise batch-put-asset-property-value -\-cli-input-json file://batch-put-payload.json
```

Mengkonfigurasi pengaturan notifikasi (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk memperbarui nilai atribut yang menentukan pengaturan notifikasi untuk alarm.

Untuk memperbarui pengaturan notifikasi alarm (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.

3. Pilih aset yang ingin Anda perbarui pengaturannya.
4. Pilih Edit.
5. Temukan atribut yang digunakan alarm untuk pengaturan notifikasi yang ingin Anda ubah, lalu masukkan nilai barunya.
6. Pilih Simpan.

Mengkonfigurasi pengaturan notifikasi (CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk memperbarui nilai atribut yang menentukan pengaturan notifikasi untuk alarm.

Anda harus mengetahui aset `assetId` dan properti Anda `propertyId` untuk menyelesaikan prosedur ini. Anda juga dapat menggunakan ID eksternal. Jika Anda membuat aset dan tidak mengetahuinya `assetId`, gunakan [ListAssets](#) API untuk mencantumkan semua aset untuk model tertentu. Gunakan [DescribeAsset](#) operasi untuk melihat properti aset Anda termasuk ID properti.

Gunakan [BatchPutAssetPropertyValue](#) operasi untuk menetapkan nilai atribut ke aset Anda. Anda dapat menggunakan operasi ini untuk mengatur beberapa atribut sekaligus. Payload operasi ini berisi daftar entri, dan setiap entri berisi ID aset, ID properti, dan nilai atribut.

Untuk memperbarui nilai atribut (AWS CLI)

1. Buat file bernama `batch-put-payload.json` dan salin objek JSON berikut ke dalam file. Contoh payload ini menunjukkan cara mengatur lintang dan bujur turbin angin. Perbarui ID, nilai, dan stempel waktu untuk mengubah payload untuk kasus penggunaan Anda.

```
{
  "entries": [
    {
      "entryId": "windfarm3-turbine7-latitude",
      "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "propertyId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "propertyValues": [
        {
          "value": {
            "doubleValue": 47.6204
          },
          "timestamp": {
            "timeInSeconds": 1575691200
          }
        }
      ]
    }
  ]
}
```



```
    }
  ]
},
{
  "entryId": "windfarm3-turbine7-longitude",
  "assetId": "a1b2c3d4-5678-90ab-cdef-2222EXAMPLE",
  "propertyId": "a1b2c3d4-5678-90ab-cdef-5555EXAMPLE",
  "propertyValues": [
    {
      "value": {
        "doubleValue": 122.3491
      },
      "timestamp": {
        "timeInSeconds": 1575691200
      }
    }
  ]
}
]
```

- Setiap entri dalam payload berisi `entryId` yang dapat Anda definisikan sebagai string unik apa pun. Jika ada entri permintaan yang gagal, setiap kesalahan akan berisi permintaan `entryId` yang sesuai sehingga Anda tahu permintaan mana yang harus dicoba lagi.
- Untuk menetapkan nilai atribut, Anda dapat menyertakan satu `timestamp-quality-value` (TQV) struktur dalam daftar `propertyValues` untuk setiap properti atribut. Struktur ini harus berisi yang baru `value` dan saat `timestamp`.
 - `value`— Struktur yang berisi salah satu bidang berikut, tergantung pada jenis properti yang ditetapkan:
 - `booleanValue`
 - `doubleValue`
 - `integerValue`
 - `stringValue`
 - `timestamp`— Struktur yang berisi waktu epoch Unix saat ini dalam hitungan detik,. `timeInSeconds` AWS IoT SiteWise menolak setiap titik data dengan stempel waktu yang ada lebih dari 7 hari di masa lalu atau lebih baru dari 5 menit di masa depan.

Untuk informasi lebih lanjut tentang cara menyiapkan muatan [BatchPutAssetPropertyValue](#), lihat [Menyerap data menggunakan API AWS IoT SiteWise](#).

2. Jalankan perintah berikut untuk mengirim nilai atribut ke AWS IoT SiteWise:

```
aws iotsitewise batch-put-asset-property-value -\-cli-input-json file://batch-put-payload.json
```

Menanggapi alarm

Saat AWS IoT Events alarm berubah status, Anda dapat melakukan hal berikut untuk merespons alarm:

- Akui alarm untuk menunjukkan bahwa Anda menangani masalah tersebut.
- Tunda alarm untuk menonaktifkannya sementara.
- Nonaktifkan alarm untuk menonaktifkannya secara permanen sampai Anda mengaktifkannya lagi.
- Aktifkan alarm yang dinonaktifkan untuk mendeteksi status alarm.
- Setel ulang alarm untuk menghapus status dan nilai terbarunya.

Anda dapat menggunakan AWS IoT SiteWise konsol atau AWS IoT Events API untuk merespons alarm.

Note

Anda dapat merespons AWS IoT Events alarm, tetapi bukan alarm eksternal.

Topik

- [Menanggapi alarm \(konsol\)](#)
- [Menanggapi alarm \(API\)](#)

Menanggapi alarm (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk mengakui, menunda, menonaktifkan, atau mengaktifkan alarm.

Topik

- [Akui alarm \(konsol\)](#)
- [Tunda alarm \(konsol\)](#)
- [Nonaktifkan alarm \(konsol\)](#)
- [Aktifkan alarm \(konsol\)](#)
- [Setel ulang alarm \(konsol\)](#)

Akui alarm (konsol)

Anda dapat menerima alarm untuk menunjukkan bahwa Anda menangani masalah tersebut.

Note

Anda harus mengaktifkan aliran pengakuan pada alarm sehingga Anda dapat mengenali alarm. Opsi ini diaktifkan secara default jika Anda menentukan alarm dari AWS IoT SiteWise konsol.

Untuk mengakui alarm (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang ingin Anda akui alarmnya.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih tab Alarm.
5. Pilih alarm untuk mengakui, lalu pilih Tindakan untuk membuka menu tindakan respons.
6. Pilih Akui. Status alarm berubah menjadi Diakui.

Tunda alarm (konsol)

Anda dapat menunda alarm untuk menonaktifkannya sementara. Tentukan durasi untuk menunda alarm.

Untuk menunda alarm (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang ingin Anda tunda alarm.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih tab Alarm.
5. Pilih alarm untuk ditunda, lalu pilih Tindakan untuk membuka menu tindakan respons.
6. Pilih Tunda. Model terbuka di mana Anda menentukan durasi untuk menunda.
7. Pilih panjang Tunda atau masukkan panjang Tunda khusus.
8. Pilih Simpan. Status alarm berubah menjadi Snoozed.

Nonaktifkan alarm (konsol)

Anda dapat menonaktifkan alarm sehingga tidak mendeteksi lagi. Setelah Anda menonaktifkan alarm, Anda harus mengaktifkannya lagi jika Anda ingin alarm mendeteksi.

Untuk menonaktifkan alarm (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang ingin Anda nonaktifkan alarmnya.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih tab Alarm.
5. Pilih alarm untuk dinonaktifkan, lalu pilih Tindakan untuk membuka menu tindakan respons.
6. Pilih Disable (Nonaktifkan). Status alarm berubah menjadi Dinonaktifkan.

Aktifkan alarm (konsol)

Anda dapat mengaktifkan alarm untuk mendeteksi lagi setelah Anda menonaktifkan atau menundanya.

Untuk mengaktifkan alarm (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang ingin Anda aktifkan alarmnya.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih tab Alarm.
5. Pilih alarm yang akan diaktifkan, lalu pilih Tindakan untuk membuka menu tindakan respons.
6. Pilih Aktifkan. Status alarm berubah menjadi Normal.

Setel ulang alarm (konsol)

Anda dapat mengatur ulang alarm untuk menghapus status dan nilai terbarunya.

Untuk mengatur ulang alarm (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang ingin Anda atur ulang alarm.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih tab Alarm.
5. Pilih alarm yang akan diaktifkan, lalu pilih Tindakan untuk membuka menu tindakan respons.
6. Pilih Reset. Status alarm berubah menjadi Normal.

Menanggapi alarm (API)

Anda dapat menggunakan AWS IoT Events API untuk mengakui, menunda, menonaktifkan, mengaktifkan, atau mengatur ulang alarm. Untuk informasi selengkapnya, lihat operasi berikut di Referensi AWS IoT Events API:

- [BatchAcknowledgeAlarm](#)
- [BatchSnoozeAlarm](#)
- [BatchDisableAlarm](#)
- [BatchEnableAlarm](#)
- [BatchResetAlarm](#)

Untuk informasi selengkapnya, lihat [Menanggapi alarm](#) di Panduan AWS IoT Events Pengembang.

Menelan status alarm eksternal

Alarm eksternal adalah alarm yang Anda evaluasi di luar. AWS IoT SiteWise Anda dapat menggunakan alarm eksternal ketika Anda memiliki sumber data yang melaporkan status alarm yang ingin Anda konsumsi. AWS IoT SiteWise

Properti status alarm memerlukan format khusus untuk nilai data status alarm. Setiap nilai data harus berupa objek JSON yang diserialisasikan ke string. Kemudian, Anda menelan string serial sebagai nilai string. Untuk informasi selengkapnya, lihat [Properti status alarm](#).

Example Contoh nilai data status alarm (tidak diserialisasi)

```
{
```

```
"stateName": "Active"  
}
```

Example Contoh nilai data status alarm (serial)

```
{\"stateName\": \"Active\"}
```

Note

Jika sumber data tidak dapat melaporkan data dalam format ini, atau Anda tidak dapat mengonversi data ke format ini sebelum menelannya, Anda mungkin memilih untuk tidak menggunakan properti alarm. Sebagai gantinya, Anda dapat menelan data sebagai properti pengukuran dengan tipe data string, misalnya. Lihat informasi yang lebih lengkap di [Mendefinisikan aliran data dari peralatan \(pengukuran\)](#) dan [Menelan data ke AWS IoT SiteWise](#).

Memetakan aliran status alarm eksternal

Anda dapat menentukan alias properti untuk memetakan aliran data Anda ke properti status alarm Anda. Ini membantu Anda dengan mudah mengidentifikasi properti status alarm saat Anda menelan atau mengambil data. Untuk informasi selengkapnya tentang alias properti, lihat [Memetakan aliran data industri ke properti aset](#).

Topik

- [Memetakan aliran status alarm eksternal \(konsol\)](#)
- [Memetakan aliran status alarm eksternal \(\)AWS CLI](#)

Memetakan aliran status alarm eksternal (konsol)

Anda dapat menentukan alias properti untuk memetakan aliran data Anda ke properti status alarm Anda. Ini membantu Anda dengan mudah mengidentifikasi properti status alarm saat Anda menelan atau mengambil data. Untuk informasi selengkapnya tentang alias properti, lihat [Memetakan aliran data industri ke properti aset](#).

Anda dapat menggunakan AWS IoT SiteWise konsol untuk menyetel alias untuk properti status alarm.

Untuk menyetel alias properti untuk properti status alarm (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset yang ingin Anda tetapkan alias properti.

 Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.


4. Pilih Alarms.
5. Pilih alarm eksternal yang ingin Anda atur alias properti.
6. Pilih Lihat.
7. Di panel Detail status alarm, pilih Edit.
8. Masukkan alias properti.
9. Pilih Perbarui.

Memetakan aliran status alarm eksternal (AWS CLI)

Anda dapat menentukan alias properti untuk memetakan aliran data Anda ke properti status alarm Anda. Ini membantu Anda dengan mudah mengidentifikasi properti status alarm saat Anda menelan atau mengambil data. Untuk informasi selengkapnya tentang alias properti, lihat [Memetakan aliran data industri ke properti aset](#).

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk menyetel alias untuk properti status alarm.

Anda harus mengetahui aset `assetId` dan properti Anda `propertyId` untuk menyelesaikan prosedur ini. Anda juga dapat menggunakan ID eksternal. Jika Anda membuat aset dan tidak mengetahuinya `assetId`, gunakan [ListAssets](#) API untuk mencantumkan semua aset untuk model tertentu. Gunakan [DescribeAsset](#) operasi untuk melihat properti aset Anda termasuk ID properti.

 Note

[DescribeAsset](#) Tanggapan tersebut mencakup daftar model aset komposit untuk aset tersebut. Setiap alarm adalah model komposit. Untuk menemukan `propertyId`, temukan

model komposit untuk alarm, dan kemudian temukan AWS/ALARM_STATE properti dalam model komposit itu.

Untuk informasi selengkapnya tentang cara menyetel alias properti, lihat [Mengatur alias properti \(AWS CLI\)](#).

Menelan data status alarm

Properti status alarm mengharapkan status alarm sebagai string JSON serial. Untuk menyerap status alarm ke alarm eksternal AWS IoT SiteWise, Anda menelan string serial ini sebagai nilai string stempel waktu. Contoh berikut menunjukkan nilai data status untuk alarm aktif.

```
{\"stateName\": \"Active\"}
```

Untuk mengidentifikasi properti status alarm, Anda dapat menentukan salah satu dari berikut ini:

- Alarm assetId dan propertyId properti alarm yang Anda kirim datanya.
- ThepropertyAlias, yang merupakan alias aliran data (misalnya, /company/windfarm/3/turbine/7/temperature/high). Untuk menggunakan opsi ini, Anda harus terlebih dahulu mengatur alias properti alarm Anda. Untuk mempelajari cara menyetel alias properti untuk properti status alarm, lihat [Memetakan aliran status alarm eksternal](#).

Contoh payload [BatchPutAssetPropertyValue](#) API berikut menunjukkan cara memformat status alarm eksternal. Alarm eksternal ini melaporkan ketika pembacaan rotasi turbin angin per menit (RPM) terlalu tinggi.

Example Contoh BatchPutAssetPropertyValue payload untuk data status alarm

```
{
  "entries": [
    {
      "entryId": "unique entry ID",
      "propertyAlias": "/company/windfarm/3/turbine/7/temperature/high",
      "propertyValues": [
        {
          "value": {
            "stringValue": "{\"stateName\": \"Active\"}"
          }
        }
      ]
    }
  ]
}
```

```
        "timestamp": {
          "timeInSeconds": 1607550262
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya tentang cara menggunakan BatchPutAssetPropertyValue API untuk mencerna data, lihat [Menyerap data menggunakan API AWS IoT SiteWise](#).

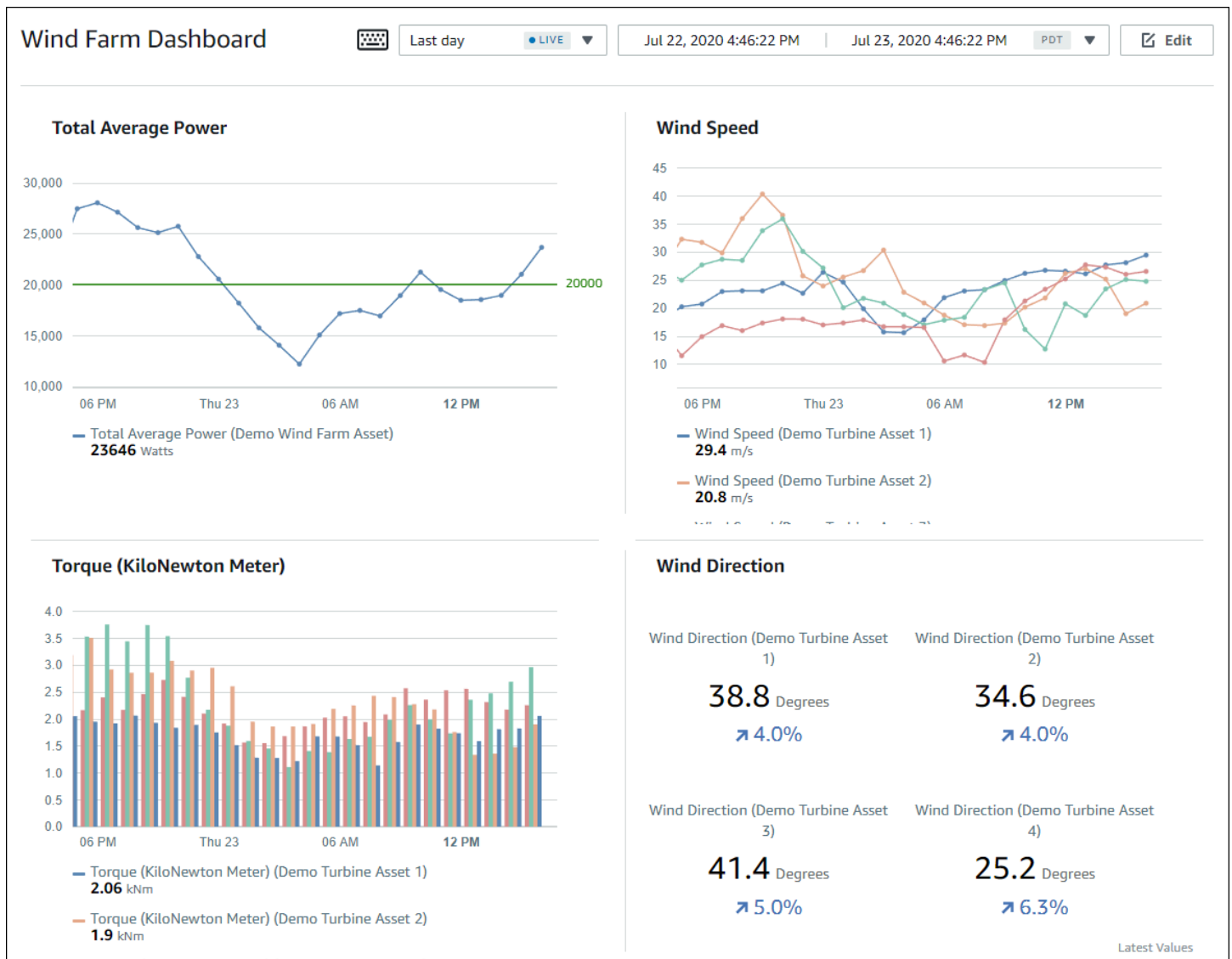
Untuk informasi selengkapnya tentang cara lain untuk menelan data, lihat [Menelan data ke AWS IoT SiteWise](#).

Memantau data dengan AWS IoT SiteWise Monitor

Anda dapat menggunakan AWS IoT SiteWise untuk memantau data dari proses, perangkat, dan peralatan Anda dengan membuat portal web SiteWise Monitor. SiteWise Monitor adalah fitur AWS IoT SiteWise yang dapat Anda gunakan untuk membuat portal dalam bentuk aplikasi web yang dikelola. Anda kemudian dapat menggunakan portal ini untuk melihat dan membagikan data operasional Anda. Anda dapat membuat proyek dengan dasbor untuk memvisualisasikan data dari proses, perangkat, dan peralatan yang terhubung. AWS IoT

Pakar domain, seperti insinyur proses, dapat menggunakan portal ini untuk dengan cepat mendapatkan wawasan tentang data operasional mereka untuk memahami perilaku perangkat dan peralatan.

Berikut ini adalah contoh dasbor yang menampilkan data untuk ladang angin.



Karena AWS IoT SiteWise menangkap data dari waktu ke waktu, Anda dapat menggunakan SiteWise Monitor untuk melihat data operasional dari waktu ke waktu, atau nilai terakhir yang dilaporkan pada titik waktu tertentu. Ini memungkinkan Anda mengungkap wawasan yang mungkin sulit ditemukan.

SiteWise Memantau peran

Empat peran berinteraksi dengan SiteWise Monitor:

AWS administrator

AWS Administrator menggunakan AWS IoT SiteWise konsol untuk membuat portal. AWS Administrator juga dapat menetapkan administrator portal dan menambahkan pengguna portal.

Administrator portal kemudian menetapkan pengguna portal ke proyek sebagai pemilik atau pemirsa. AWS Administrator bekerja secara eksklusif di AWS konsol.

Administrator portal

Setiap portal SiteWise Monitor memiliki satu atau lebih administrator portal. Administrator portal menggunakan portal untuk membuat proyek yang berisi koleksi aset dan dasbor. Administrator portal kemudian menetapkan aset dan pemilik untuk setiap proyek. Dengan mengontrol akses ke proyek, administrator portal menentukan aset mana yang dapat dilihat oleh pemilik proyek dan pemirsa.

Pemilik proyek

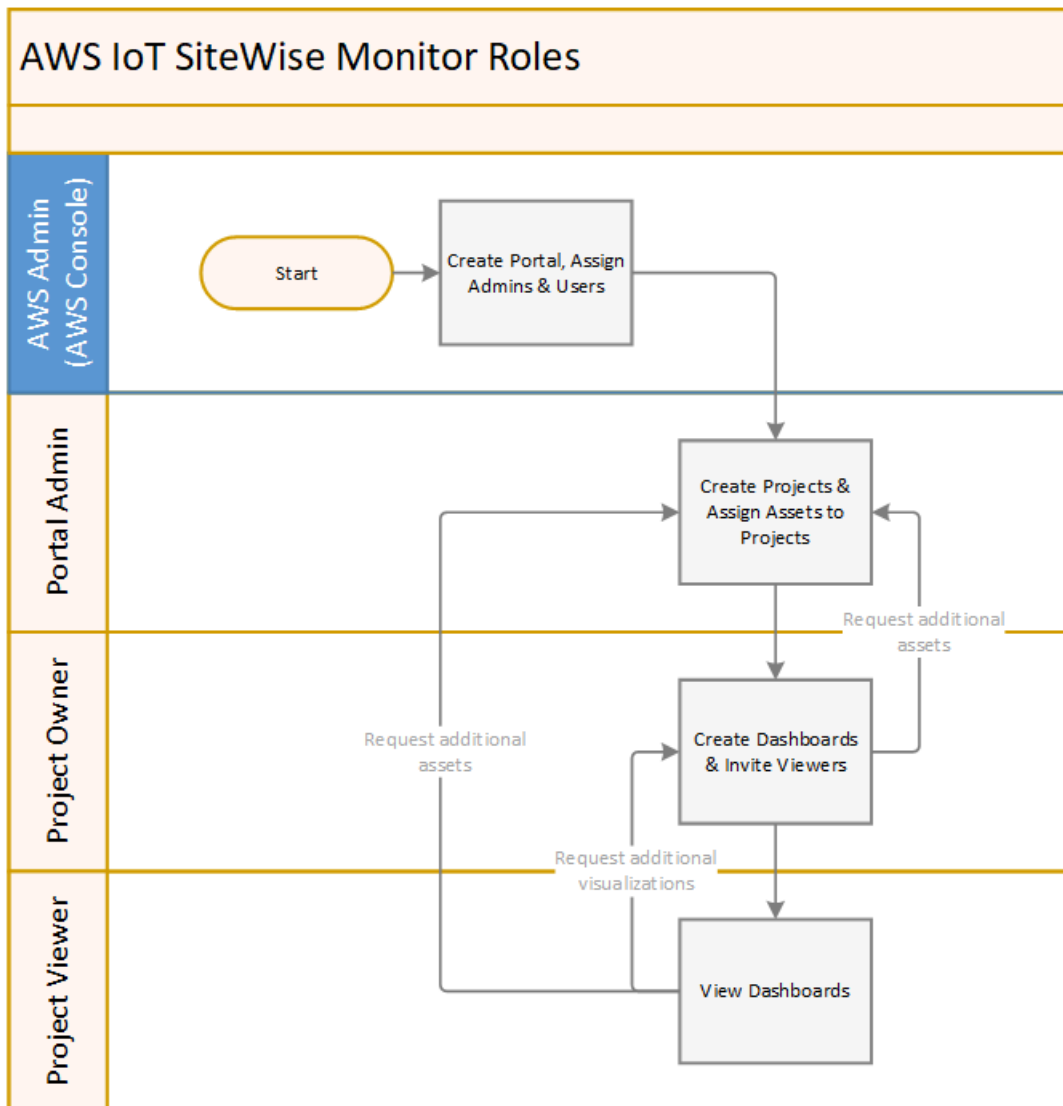
Setiap proyek SiteWise Monitor memiliki pemilik. Pemilik proyek membuat visualisasi dalam bentuk dasbor untuk mewakili data operasional secara konsisten. Ketika dasbor siap untuk dibagikan, pemilik proyek dapat mengundang pemirsa ke proyek. Pemilik proyek juga dapat menugaskan pemilik lain ke proyek. Pemilik proyek dapat mengonfigurasi ambang batas dan pengaturan notifikasi untuk alarm.

Penampil proyek

Setiap proyek SiteWise Monitor memiliki pemirsa. Pemirsa proyek dapat terhubung ke portal untuk melihat dasbor yang dibuat oleh pemilik proyek. Di setiap dasbor, pemirsa proyek dapat menyesuaikan rentang waktu untuk lebih memahami data operasional. Pemirsa proyek hanya dapat melihat dasbor di proyek yang dapat mereka akses. Pemirsa proyek dapat mengenali dan menunda alarm.

Bergantung pada organisasi Anda, orang yang sama mungkin melakukan banyak peran.

Gambar berikut menggambarkan bagaimana keempat peran ini berinteraksi di portal SiteWise Monitor.



Anda dapat mengelola siapa yang memiliki akses ke data Anda dengan menggunakan AWS IAM Identity Center atau IAM. Pengguna data Anda dapat masuk ke SiteWise Monitor dari desktop atau browser seluler menggunakan IAM Identity Center atau IAM mereka.

Federasi SAFL

IAM Identity Center dan IAM mendukung federasi identitas dengan [SAFL \(Security Assertion Markup Language\) 2.0](#). SAFL 2.0 adalah standar terbuka yang digunakan oleh banyak penyedia identitas eksternal (IdPs) untuk mengautentikasi pengguna dan meneruskan identitas dan informasi keamanan mereka ke penyedia layanan (SP). SP biasanya aplikasi atau layanan. Federasi SAFL memungkinkan administrator dan pengguna portal SiteWise Monitor Anda untuk masuk ke portal yang ditetapkan dengan kredensi eksternal, seperti nama pengguna dan kata sandi perusahaan mereka.

Anda dapat mengonfigurasi IAM Identity Center dan IAM untuk menggunakan federasi berbasis SAML untuk akses ke portal Monitor Anda. SiteWise

Pusat Identitas IAM

Administrator dan pengguna portal Anda dapat masuk ke portal AWS akses dengan nama pengguna dan kata sandi perusahaan mereka. Mereka kemudian dapat menavigasi ke portal SiteWise Monitor yang ditugaskan. IAM Identity Center menggunakan sertifikat untuk mengatur hubungan kepercayaan SAFL antara penyedia identitas Anda dan AWS. Untuk informasi lebih lanjut, [profil SCIM dan implementasi SAFL 2.0](#) di AWS IAM Identity Center Panduan Pengguna.

IAM

Administrator dan pengguna portal Anda dapat meminta kredensial keamanan sementara untuk mengakses portal Monitor yang ditetapkan SiteWise . Anda membuat identitas penyedia identitas SALL di IAM untuk mengatur hubungan kepercayaan antara penyedia identitas Anda dan AWS. Untuk informasi selengkapnya, lihat [Menggunakan federasi berbasis SAML untuk akses API AWS](#), di Panduan Pengguna IAM.

Administrator dan pengguna portal Anda dapat masuk ke portal perusahaan Anda dan memilih opsi untuk pergi ke konsol AWS Manajemen. Mereka kemudian dapat menavigasi ke portal SiteWise Monitor yang ditugaskan. Portal perusahaan Anda menangani pertukaran kepercayaan antara penyedia identitas Anda dan AWS. Untuk informasi selengkapnya, lihat [Mengaktifkan pengguna federasi SAFL 2.0 untuk mengakses Konsol AWS Manajemen di Panduan Pengguna IAM](#).

Note

Saat menambahkan pengguna atau administrator ke portal, hindari membuat kebijakan IAM yang membatasi izin pengguna, seperti IP terbatas. Kebijakan terlampir dengan izin terbatas tidak akan dapat terhubung ke AWS IoT SiteWise portal.

SiteWise Pantau konsep

Untuk menggunakan SiteWise Monitor, Anda harus terbiasa dengan konsep-konsep berikut:

Portal

AWS IoT SiteWise Monitor Portal adalah aplikasi web yang dapat Anda gunakan untuk memvisualisasikan dan berbagi AWS IoT SiteWise data Anda. Portal memiliki satu atau lebih administrator dan berisi nol atau lebih proyek.

Proyek

Setiap portal SiteWise Monitor berisi serangkaian proyek. Setiap proyek memiliki subset AWS IoT SiteWise aset Anda yang terkait dengannya. Pemilik proyek membuat satu atau beberapa dasbor untuk menyediakan cara yang konsisten untuk melihat data yang terkait dengan aset tersebut. Pemilik proyek dapat mengundang pemirsa ke proyek untuk memungkinkan mereka melihat aset dan dasbor dalam proyek. Proyek ini adalah unit dasar berbagi dalam SiteWise Monitor. Pemilik proyek dapat mengundang pengguna yang diberi akses ke portal oleh AWS administrator. Pengguna harus memiliki akses ke portal sebelum proyek di portal itu dapat dibagikan dengan pengguna tersebut.

Aset

Ketika data dicerna AWS IoT SiteWise dari peralatan industri Anda, perangkat, peralatan, dan proses Anda masing-masing direpresentasikan sebagai aset. Setiap aset memiliki properti dan alarm yang terkait dengannya. Administrator portal menetapkan set aset untuk setiap proyek.

Properti

Properti adalah data deret waktu yang terkait dengan aset. Misalnya, peralatan mungkin memiliki nomor seri, lokasi, merek dan model, dan tanggal pemasangan. Mungkin juga memiliki nilai deret waktu untuk ketersediaan, kinerja, kualitas, suhu, tekanan, dan sebagainya.

Alarm

Alarm memantau properti untuk mengidentifikasi kapan peralatan berada di luar jangkauan operasinya. Setiap alarm mendefinisikan ambang batas dan properti untuk dipantau. Ketika properti melebihi ambang batas, alarm menjadi aktif dan menunjukkan bahwa Anda atau seseorang di tim Anda harus mengatasi masalah tersebut. Pemilik proyek dapat menyesuaikan ambang batas dan pengaturan notifikasi untuk alarm. Pemirsa proyek dapat mengenali dan menunda alarm, dan mereka dapat meninggalkan pesan dengan detail tentang alarm atau tindakan yang mereka ambil untuk mengatasinya.

Dasbor

Setiap proyek berisi satu set dasbor. Dasbor menyediakan satu set visualisasi untuk nilai-nilai dari satu set aset. Pemilik proyek membuat dasbor dan visualisasi yang dikandungnya. Ketika

pemilik proyek siap untuk membagikan kumpulan dasbor, pemilik dapat mengundang pemirsa ke proyek, yang memberi mereka akses ke semua dasbor dalam proyek. Jika Anda menginginkan kumpulan pemirsa yang berbeda untuk dasbor yang berbeda, Anda harus membagi dasbor di antara proyek. Saat pemirsa melihat dasbor, mereka dapat menyesuaikan rentang waktu untuk melihat data tertentu.

Visualisasi

Di setiap dasbor, pemilik proyek memutuskan bagaimana menampilkan properti dan alarm aset yang terkait dengan proyek. Ketersediaan dapat direpresentasikan sebagai grafik garis, sementara nilai lainnya mungkin ditampilkan sebagai grafik batang atau indikator kinerja utama (KPI). Alarm paling baik ditampilkan sebagai kisi-kisi status dan jadwal status. Pemilik proyek menyesuaikan setiap visualisasi untuk memberikan pemahaman terbaik tentang data untuk aset tersebut.

Memulai dengan AWS IoT SiteWise Monitor

Jika Anda AWS administrator untuk organisasi Anda, Anda membuat portal dari AWS IoT SiteWise konsol. Selesaikan langkah-langkah berikut untuk membuat portal sehingga anggota organisasi Anda dapat melihat AWS IoT SiteWise data Anda:

1. Konfigurasi dan buat portal
2. Tambahkan administrator portal dan kirim email undangan
3. Tambahkan pengguna portal

Setelah Anda membuat portal, administrator portal dapat melihat AWS IoT SiteWise aset Anda dan menentukannya ke proyek di portal. Pemilik proyek kemudian dapat membuat dasbor untuk memvisualisasikan properti aset yang membantu pemirsa proyek memahami kinerja perangkat, proses, dan peralatan Anda.

Note

Saat menambahkan pengguna atau administrator ke portal, hindari membuat kebijakan AWS Identity and Access Management (IAM) yang membatasi izin pengguna, seperti IP terbatas. Kebijakan terlampir dengan izin terbatas tidak akan dapat terhubung ke AWS IoT SiteWise portal.

Anda dapat mengikuti tutorial yang berjalan melalui langkah-langkah yang diperlukan untuk menyiapkan portal dengan proyek, dasbor, dan beberapa pengguna untuk skenario tertentu menggunakan data ladang angin. Untuk informasi selengkapnya, lihat [Memvisualisasikan dan berbagi data ladang angin di Monitor SiteWise](#).

Topik

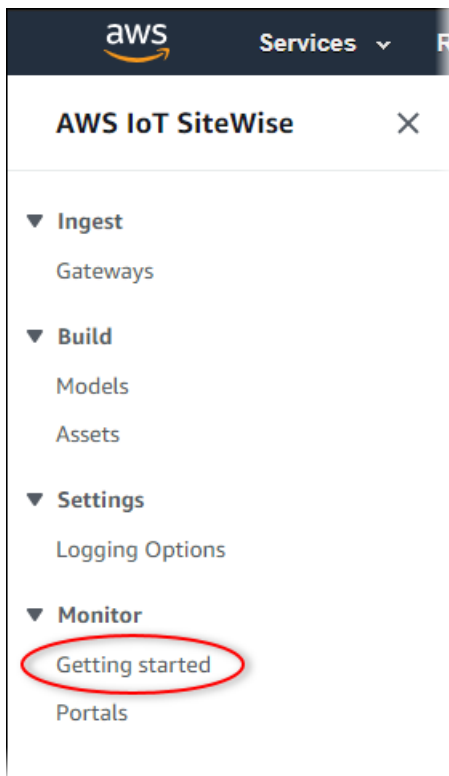
- [Membuat portal](#)
- [Mengkonfigurasi portal Anda](#)
- [Mengundang administrator](#)
- [Menambahkan pengguna portal](#)

Membuat portal

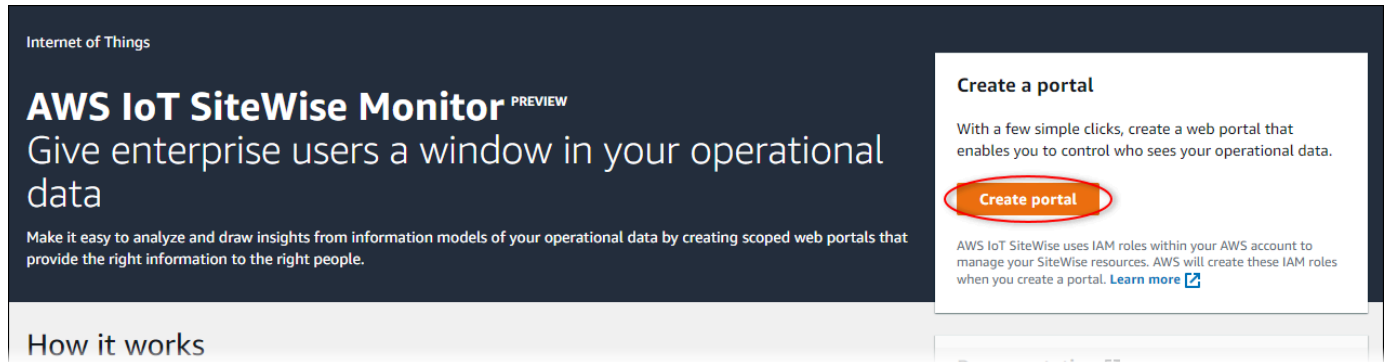
Anda membuat portal SiteWise Monitor di AWS IoT SiteWise konsol.

Untuk membuat portal

1. Masuk ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Monitor, Memulai.



3. Pilih Buat Portal.



Selanjutnya, Anda harus memberikan beberapa informasi dasar untuk mengkonfigurasi portal Anda.

Mengkonfigurasi portal Anda

Pengguna Anda menggunakan portal untuk melihat data Anda. Anda dapat menyesuaikan nama portal, deskripsi, branding, otentikasi pengguna, email kontak dukungan, dan izin.

AWS IoT SiteWise > Monitor > Portals > Create portal

Step 1
Portal configurationStep 2 - optional
Additional featuresStep 3
Invite administratorsStep 4
Assign users

Portal configuration

Each web portal provides enterprise users with access to your IoT SiteWise assets. [Learn more](#)

Portal details

Portal name

Choose a portal name to identify the web portal to your users. Company name is recommended.

example-factory-1

Name should be 1-128 characters and only contain A-Z a-z 0-9 _ and -.

Description - optional

Create a description of your portal

Example Corp Factory #1 in Renton, WA

Description should contain a maximum of 2048 characters.

Portal branding

You can provide your logo image to display your brand in this web portal.

Logo image


Upload a square, high-resolution .png file. The image is displayed on a dark background.

Choose file

The file size must be less than 1 MB.

User authentication

Your users can sign in to this portal with their AWS Single Sign-On (AWS SSO) or AWS Identity and Access Management (IAM) credentials. If you choose AWS SSO, you must enable the service for your AWS account.

 You haven't enabled AWS SSO in your account yet. When you create your first portal user, this automatically enables AWS SSO in your AWS account.

Create user

AWS SSO

Your users can sign in to the portal with their corporate usernames and passwords.

IAM

Your users can sign in to the portal with their IAM credentials.

Support contact email

You can provide an email address for cases where there's a problem or issue with this portal and your users need to contact support to resolve.

Email

support@example.com

Tags

This resource doesn't have any tags.

Add tag

You can add up to 50 more tags.

Permissions

SiteWise Monitor assumes this role to give permissions to your federated users to access AWS IoT SiteWise resources. [Learn](#)

Untuk mengkonfigurasi portal

1. Masukkan nama untuk portal Anda.
2. (Opsional) Masukkan deskripsi untuk portal Anda. Jika Anda memiliki beberapa portal, gunakan deskripsi yang bermakna untuk membantu Anda melacak isi setiap portal.
3. (Opsional) Unggah gambar untuk menampilkan merek Anda di portal. Pilih persegi, gambar PNG. Jika Anda mengunggah gambar non-persegi, portal menskalakan gambar menjadi persegi.
4. Pilih salah satu opsi berikut:
 - Pilih Pusat Identitas IAM jika pengguna portal Anda masuk ke portal ini dengan nama pengguna dan kata sandi perusahaan mereka.

Jika Anda belum mengaktifkan Pusat Identitas IAM di akun Anda, lakukan hal berikut:

- a. Pilih Create user (Buat pengguna).
- b. Pada halaman Buat pengguna, untuk membuat portal pertama, masukkan alamat email pengguna, nama depan, dan nama belakang, lalu pilih Buat pengguna.

The screenshot shows a 'Create user' dialog box. It contains the following elements:

- Title:** Create user
- Message:** When you create your first portal user, this automatically enables AWS SSO in your AWS account.
- Fields:**
 - Email address:** janedoe@example.com
 - First name:** Jane
 - Last name:** Doe
- Buttons:** Cancel, Create user

Note

- AWS secara otomatis mengaktifkan Pusat Identitas IAM di akun Anda saat Anda membuat pengguna portal pertama.
- Anda dapat mengonfigurasi Pusat Identitas IAM hanya dalam satu Wilayah pada satu waktu. SiteWise Monitor terhubung ke Wilayah yang Anda konfigurasi untuk Pusat Identitas IAM. Ini berarti Anda menggunakan satu


Wilayah untuk akses Pusat Identitas IAM, tetapi Anda dapat membuat portal di Wilayah mana pun.

- Pilih IAM jika pengguna portal Anda masuk ke portal ini dengan kredensi IAM mereka.

 Important

Pengguna atau peran harus memiliki `iotsitewise:DescribePortal` izin untuk masuk ke portal.

5. Masukkan alamat email yang dapat dihubungi pengguna portal ketika mereka memiliki masalah dengan portal dan perlu bantuan untuk menyelesaikannya.
6. (Opsional) Tambahkan tag untuk portal Anda. Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda AWS IoT SiteWise](#).
7. Pilih salah satu opsi berikut:
 - Pilih Buat dan gunakan peran layanan baru. Secara default, SiteWise Monitor secara otomatis membuat peran layanan untuk setiap portal. Peran ini memungkinkan pengguna portal Anda mengakses AWS IoT SiteWise sumber daya Anda. Untuk informasi selengkapnya, lihat [Menggunakan peran layanan untuk AWS IoT SiteWise Monitor](#).
 - Pilih Gunakan peran layanan yang ada, lalu pilih peran target.
8. Pilih Selanjutnya
9. (Opsional) Aktifkan alarm untuk portal Anda. Untuk informasi selengkapnya, lihat [Mengaktifkan alarm untuk portal Anda](#).
10. Pilih Buat. AWS IoT SiteWise akan membuat portal Anda.

 Note

Jika Anda menutup konsol, Anda dapat menyelesaikan proses penyiapan dengan menambahkan administrator dan pengguna. Untuk informasi selengkapnya, lihat [Menambahkan atau menghapus administrator portal](#). Jika Anda tidak ingin menyimpan portal ini, hapus agar tidak menggunakan sumber daya. Untuk informasi selengkapnya, lihat [Menghapus portal](#).

Kolom Status dapat menjadi salah satu nilai berikut.

- **AWS IoT SiteWise CREATING** - sedang memproses permintaan Anda untuk membuat portal. Proses ini bisa memakan waktu beberapa menit untuk menyelesaikannya.
- **UPDATE** - AWS IoT SiteWise sedang memproses permintaan Anda untuk memperbarui portal. Proses ini bisa memakan waktu beberapa menit untuk menyelesaikannya.
- **PENDING** - AWS IoT SiteWise sedang menunggu propagasi catatan DNS selesai. Proses ini bisa memakan waktu beberapa menit untuk menyelesaikannya. Anda dapat menghapus portal saat statusnya **TERTUNDA**.
- **MENGHAPUS** - AWS IoT SiteWise adalah memproses permintaan Anda untuk menghapus portal. Proses ini bisa memakan waktu beberapa menit untuk menyelesaikannya.
- **AKTIF** - Ketika portal menjadi aktif, pengguna portal Anda dapat mengaksesnya.
- **GAGAL** - AWS IoT SiteWise tidak dapat memproses permintaan Anda untuk membuat, memperbarui, atau menghapus portal. Jika Anda mengaktifkan AWS IoT SiteWise untuk mengirim log ke Amazon CloudWatch Logs, Anda dapat menggunakan log ini untuk memecahkan masalah. Untuk informasi selengkapnya, lihat [Pemantauan AWS IoT SiteWise dengan CloudWatch Log](#).

Sebuah pesan muncul saat portal Anda dibuat.

A green banner with a white checkmark icon on the left and a white 'X' icon on the right. The text in the center reads: "Successfully created portal URL at https://a1b2c3d4-5678-90ab-cdef-1111EXAMPLE.app.iotsitewise.aws".

Successfully created portal URL at <https://a1b2c3d4-5678-90ab-cdef-1111EXAMPLE.app.iotsitewise.aws>

Selanjutnya, Anda harus mengundang satu atau lebih administrator portal ke portal. Sejauh ini, Anda membuat portal tetapi tidak ada yang bisa mengaksesnya.

Mengundang administrator

Untuk memulai di portal baru Anda, Anda harus menetapkan administrator portal. Administrator portal membuat proyek, memilih pemilik proyek, dan menetapkan aset untuk proyek. Administrator portal dapat melihat semua AWS IoT SiteWise aset Anda.

Berdasarkan layanan otentikasi pengguna, pilih salah satu opsi berikut:


IAM Identity Center

Jika Anda menggunakan SiteWise Monitor untuk pertama kalinya, Anda dapat memilih pengguna yang Anda buat sebelumnya untuk menjadi administrator portal. Jika Anda ingin menambahkan pengguna lain sebagai administrator portal, Anda dapat membuat pengguna Pusat Identitas IAM

dari halaman ini. Atau, Anda dapat menghubungkan penyedia identitas eksternal ke IAM Identity Center. Untuk informasi selengkapnya, silakan lihat [Panduan Pengguna AWS IAM Identity Center](#).

Untuk mengundang administrator

1. Pilih kotak centang untuk pengguna yang Anda inginkan sebagai administrator portal Anda. Ini menambahkan pengguna ke daftar administrator Portal.

 Note

Jika Anda menggunakan Pusat Identitas IAM sebagai penyimpanan identitas, dan Anda masuk ke akun AWS Organizations manajemen, Anda dapat memilih Buat pengguna untuk membuat pengguna Pusat Identitas IAM. IAM Identity Center mengirimkan email kepada pengguna baru untuk mengatur kata sandi mereka. Anda kemudian dapat menetapkan pengguna ke portal sebagai administrator. Untuk informasi selengkapnya, lihat [Mengelola identitas di Pusat Identitas IAM](#).

2. (Opsional) Pilih Kirim undangan ke pengguna yang dipilih. Klien email Anda terbuka, dan undangan diisi di badan pesan.

Anda dapat menyesuaikan email sebelum mengirimkannya ke administrator portal Anda. Anda juga dapat mengirim email ke administrator portal Anda nanti. Jika Anda mencoba SiteWise Monitor untuk pertama kalinya dan menambahkan Pusat Identitas IAM baru atau pengguna IAM atau peran sebagai administrator portal, Anda tidak perlu mengirim email sendiri.

3. Jika Anda menambahkan pengguna yang tidak Anda inginkan sebagai administrator, kosongkan kotak centang untuk pengguna tersebut.
4. Setelah selesai mengundang administrator portal, pilih Berikutnya.

IAM

Anda dapat memilih pengguna atau peran untuk menjadi administrator portal. Jika ingin menambahkan pengguna atau peran lain sebagai administrator portal, Anda dapat membuat pengguna atau peran di konsol IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di AWS akun Anda](#) dan [Membuat peran IAM](#) di Panduan Pengguna IAM.

Untuk mengundang administrator

1. Lakukan hal-hal berikut:
 - Pilih pengguna IAM untuk menambahkan pengguna IAM sebagai administrator portal Anda.
 - Pilih peran IAM untuk menambahkan peran IAM sebagai administrator portal Anda.
2. Pilih kotak centang untuk pengguna atau peran yang Anda inginkan sebagai administrator portal Anda. Ini menambahkan pengguna atau peran ke daftar administrator Portal.
3. Jika Anda menambahkan pengguna atau peran yang tidak Anda inginkan sebagai administrator, kosongkan kotak centang untuk pengguna atau peran tersebut.
4. Setelah selesai mengundang administrator portal, pilih Berikutnya.

Important

Pengguna atau peran harus memiliki `iotsitewise:DescribePortal` izin untuk masuk ke portal.

Note

Jika Anda menggunakan Pusat Identitas IAM sebagai penyimpanan identitas, dan Anda masuk ke akun AWS Organizations manajemen, Anda dapat memilih Buat pengguna untuk membuat pengguna Pusat Identitas IAM. IAM Identity Center mengirimkan email kepada pengguna baru untuk mengatur kata sandi mereka. Anda kemudian dapat menetapkan pengguna ke portal sebagai administrator. Untuk informasi selengkapnya, lihat [Mengelola identitas di Pusat Identitas IAM](#).

Anda dapat mengubah daftar administrator portal nanti. Untuk informasi selengkapnya, lihat [Menambahkan atau menghapus administrator portal](#).

Note

Karena hanya administrator portal yang dapat membuat proyek dan menetapkan aset kepada mereka, Anda harus menentukan setidaknya satu administrator portal.

Sebagai langkah terakhir, Anda menambahkan pengguna yang dapat mengakses portal baru Anda.

Menambahkan pengguna portal

Anda mengontrol pengguna mana yang memiliki akses ke portal Anda. Di setiap portal, administrator portal membuat satu atau lebih proyek dan menetapkan pengguna portal sebagai pemilik atau pemirsa untuk setiap proyek. Setiap pemilik proyek dapat mengundang pengguna portal tambahan untuk memiliki atau melihat proyek.

Berdasarkan layanan otentikasi pengguna, pilih salah satu opsi berikut:

IAM Identity Center

Jika Anda ingin menambahkan pengguna ke daftar Pengguna, selesaikan langkah-langkah berikut.

Untuk menambahkan pengguna portal

1. Pilih pengguna dari daftar Pengguna untuk ditambahkan ke portal. Ini menambahkan pengguna ke daftar pengguna Portal. Jika Anda menggunakan SiteWise Monitor untuk pertama kalinya, Anda tidak perlu menambahkan administrator portal sebagai pengguna portal.

Note

Jika Anda menggunakan Pusat Identitas IAM sebagai penyimpanan identitas, dan Anda masuk ke akun AWS Organizations manajemen, Anda dapat memilih Buat pengguna untuk membuat pengguna Pusat Identitas IAM. IAM Identity Center mengirimkan email kepada pengguna baru untuk mengatur kata sandi mereka. Anda kemudian dapat menetapkan pengguna ke portal sebagai pengguna. Untuk informasi selengkapnya, lihat [Mengelola identitas di Pusat Identitas IAM](#).

2. Jika Anda menambahkan pengguna yang tidak ingin Anda akses ke portal, kosongkan kotak centang untuk pengguna tersebut.
3. Setelah selesai memilih pengguna, pilih Tetapkan pengguna.

AWS IoT SiteWise > Monitor > Portals > Create portal

Step 1
Portal configuration

Step 2
Invite administrators

Step 3
Assign users

Assign users

Select the users you want to be able to access and view this portal. Portal administrators will send invitations to these users at a later date. [Learn more](#)

Users (2) Create user

Find resources

	Display name	Email
<input type="checkbox"/>	Jane Doe	janedoe@example.com
<input checked="" type="checkbox"/>	John Doe	johndoe@example.com

Selected users (1)

Cancel Previous **Assign users**

IAM

Jika Anda melihat pengguna atau peran yang ingin Anda tambahkan dalam daftar peran pengguna IAM atau IAM, selesaikan langkah-langkah berikut.

Untuk menambahkan pengguna portal

1. Lakukan opsi berikut:

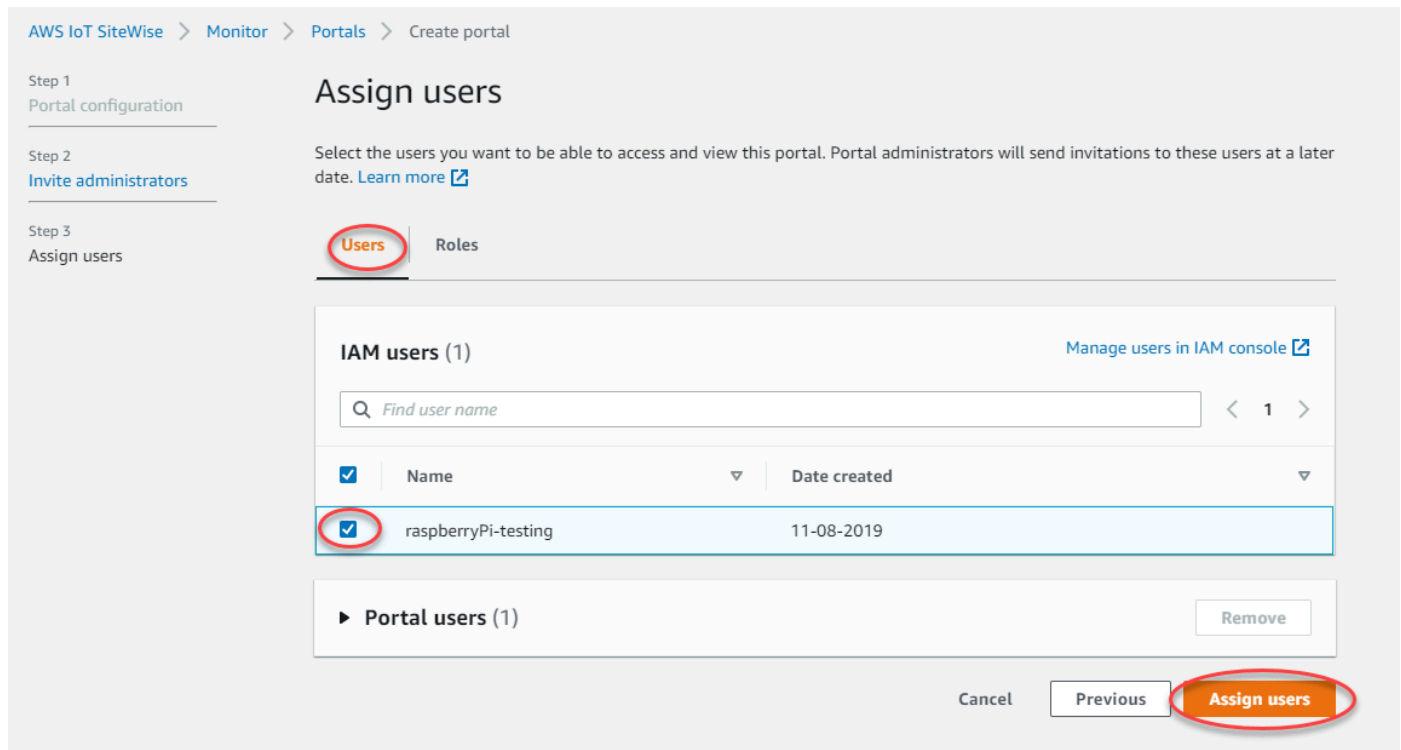
- Pilih pengguna IAM untuk menambahkan pengguna IAM sebagai pengguna portal.
- Pilih peran IAM untuk menambahkan peran IAM sebagai pengguna portal.

Jika Anda menggunakan SiteWise Monitor untuk pertama kalinya, Anda tidak perlu menambahkan administrator portal sebagai pengguna portal.

2. Pilih kotak centang untuk pengguna atau peran yang Anda inginkan sebagai pengguna portal. Ini menambahkan pengguna atau peran ke daftar pengguna Portal.
3. Jika Anda menambahkan pengguna yang tidak ingin Anda akses ke portal, kosongkan kotak centang untuk pengguna tersebut.
4. Setelah selesai memilih pengguna, pilih **Tetapkan pengguna**.

⚠ Important

Pengguna atau peran harus memiliki `iotsitewise:DescribePortal` izin untuk masuk ke portal.



AWS IoT SiteWise > Monitor > Portals > Create portal

Step 1
Portal configuration

Step 2
Invite administrators

Step 3
Assign users

Assign users

Select the users you want to be able to access and view this portal. Portal administrators will send invitations to these users at a later date. [Learn more](#)

Users Roles

IAM users (1) [Manage users in IAM console](#)

Find user name

<input checked="" type="checkbox"/>	Name	Date created
<input checked="" type="checkbox"/>	raspberryPi-testing	11-08-2019

Portal users (1) [Remove](#)

Cancel Previous **Assign users**

AWS IoT SiteWise > Monitor > Portals > Create portal

Step 1
Portal configuration

Step 2
Invite administrators

Step 3
Assign users

Assign users

Select the users you want to be able to access and view this portal. Portal administrators will send invitations to these users at a later date. [Learn more](#)

Users **Roles**

IAM roles (66) [Manage roles in IAM console](#)

 < 1 2 3 4 5 6 7 >

<input type="checkbox"/>	Name	Date created
<input type="checkbox"/>	[REDACTED]	
<input type="checkbox"/>	[REDACTED]	
<input checked="" type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_4wZigNpA1	03-16-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_EcKt-2Oar	03-11-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_GTnd0O4Wr	03-16-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_rHINLNCs-	03-11-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_XB330QUIO	03-10-2021
<input type="checkbox"/>	[REDACTED]	
<input type="checkbox"/>	[REDACTED]	
<input type="checkbox"/>	[REDACTED]	

► Portal users (2) Remove

Cancel Previous **Assign users**

Selamat! Anda berhasil membuat portal, administrator portal yang ditetapkan, dan pengguna yang ditetapkan yang dapat menggunakan portal tersebut saat diundang untuk melakukannya. Administrator portal Anda sekarang dapat membuat proyek dan menambahkan aset ke proyek tersebut. Kemudian, pemilik proyek Anda dapat membuat dasbor untuk memvisualisasikan data untuk setiap aset proyek.

Anda dapat mengubah daftar pengguna portal nanti. Untuk informasi selengkapnya, lihat [Menambahkan atau menghapus pengguna portal](#).

Jika Anda perlu membuat perubahan pada portal, lihat [Mengelola portal SiteWise Monitor Anda](#).

Untuk memulai di portal, lihat [Memulai](#) di Panduan Aplikasi SiteWise Monitor.

Membuat dasbor () AWS Command Line Interface

Saat Anda menentukan visualisasi (atau widget) di dasbor menggunakan AWS CLI, Anda harus menentukan informasi berikut dalam dokumen JSON. `dashboardDefinition` Definisi ini adalah parameter dari [CreateDashboard](#) dan [UpdateDashboard](#) operasi.

`widgets`

Daftar struktur definisi widget yang masing-masing berisi informasi berikut:

`type`

Jenis widget. AWS IoT SiteWise menyediakan jenis widget berikut:

- `sc-line-chart`— Bagan garis. Untuk informasi selengkapnya, lihat [Bagan garis](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.
- `sc-scatter-chart`— Bagan pencar. Untuk informasi selengkapnya, lihat [Bagan pencar](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.
- `sc-bar-chart`— Bagan batang. Untuk informasi selengkapnya, lihat [Bagan batang](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.
- `sc-status-grid`- Widget status yang menunjukkan nilai terbaru properti aset sebagai kisi. Untuk informasi selengkapnya, lihat [Widget status](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.
- `sc-status-timeline`— Widget status yang menunjukkan nilai historis properti aset sebagai garis waktu. Untuk informasi selengkapnya, lihat [Widget status](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.
- `sc-kpi`— Visualisasi indikator kinerja utama (KPI). Untuk informasi selengkapnya, lihat [widget KPI](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.
- `sc-table`— Sebuah widget tabel. Untuk informasi selengkapnya, lihat [Widget tabel](#) di Panduan AWS IoT SiteWise Monitor Aplikasi.

`title`

Judul widget.

x

Posisi horizontal widget, mulai dari kiri grid. Nilai ini mengacu pada posisi widget di grid dashboard.

y

Posisi vertikal widget, mulai dari bagian atas grid. Nilai ini mengacu pada posisi widget di grid dashboard.

width

Lebar widget, dinyatakan dalam jumlah spasi pada grid dashboard.

height

Ketinggian widget, dinyatakan dalam jumlah spasi pada grid dashboard.

metrics

Daftar struktur metrik yang masing-masing menentukan aliran data untuk widget ini. Setiap struktur dalam daftar harus berisi informasi berikut:

label

Label untuk ditampilkan untuk metrik ini.

type

Jenis sumber data untuk metrik ini. AWS IoT SiteWise menyediakan jenis metrik berikut:

- `iotsitewise`— Dasbor mengambil data untuk properti aset di AWS IoT SiteWise. Jika Anda memilih opsi ini, Anda harus menentukan `assetId` dan `propertyId` untuk metrik ini.

assetId

(Opsional) ID aset di AWS IoT SiteWise.

Bidang ini diperlukan jika Anda memilih `iotsitewise` untuk `type` dalam metrik ini.

propertyId

(Opsional) ID properti aset di AWS IoT SiteWise.

Bidang ini diperlukan jika Anda memilih `iotsitewise` untuk `type` dalam metrik ini.

analysis

(Opsional) Struktur yang mendefinisikan analisis, seperti garis tren, untuk ditampilkan untuk widget. Untuk informasi selengkapnya, lihat [Mengonfigurasi garis tren](#) di Panduan AWS IoT SiteWise Monitor Aplikasi. Anda dapat menambahkan salah satu dari setiap jenis garis tren per properti di widget. Struktur analisis berisi informasi berikut:

trends

(Opsional) Daftar struktur tren yang masing-masing menentukan analisis tren untuk widget ini. Setiap struktur dalam daftar berisi informasi berikut:

type

Jenis garis tren. Pilih opsi berikut:

- `linear-regression`— Menampilkan garis regresi linier. SiteWise Monitor menggunakan metode [kuadrat terkecil](#) untuk menghitung regresi linier.

annotations

(Opsional) Struktur anotasi yang mendefinisikan ambang batas untuk widget. Untuk informasi selengkapnya, lihat [Mengonfigurasi ambang batas dalam Panduan](#) Aplikasi. AWS IoT SiteWise Monitor Anda dapat menambahkan hingga enam anotasi per widget. Struktur anotasi berisi informasi berikut:

y

(Opsional) Daftar struktur anotasi yang masing-masing menentukan ambang horizontal untuk widget ini. Setiap struktur dalam daftar berisi informasi berikut:

comparisonOperator

Operator perbandingan untuk ambang batas. Pilih salah satu cara berikut:

- `LT`— Sorot properti yang memiliki setidaknya satu titik data kurang dari `value`.
- `GT`— Sorot properti yang memiliki setidaknya satu titik data lebih besar dari `value`.
- `LTE`— Sorot properti yang memiliki setidaknya satu titik data kurang dari atau sama dengan `value`.
- `GTE`— Sorot properti yang memiliki setidaknya satu titik data lebih besar dari atau sama dengan `value`.
- `EQ`— Sorot properti yang memiliki setidaknya satu titik data yang sama dengan `value`.

value

Nilai ambang batas untuk membandingkan titik data dengan `comparisonOperator`.

color

(Opsional) Kode heksadesimal 6 digit dari warna ambang batas. Visualisasi menampilkan legenda properti dalam warna ini untuk properti dengan setidaknya satu titik data yang memenuhi aturan ambang batas. Default ke `black` (). `#000000`

showValue

(Opsional) Apakah atau tidak untuk menunjukkan nilai ambang batas di margin widget. Default ke `true`.

properties

(Opsional) Kamus datar properti untuk widget. Anggota struktur ini bergantung pada konteks. AWS IoT SiteWise menyediakan widget berikut yang menggunakan `properties`:

- [Bagan garis](#), [bagan sebar](#), dan [bagan batang](#) memiliki properti berikut:

colorDataAcrossThresholds

(Opsional) Apakah atau tidak untuk mengubah warna data yang melintasi ambang batas dalam widget ini. Saat Anda mengaktifkan opsi ini, data yang melewati ambang batas muncul dalam warna yang Anda pilih. Default ke `true`.

- [Status grid](#) memiliki properti berikut:

labels

(Opsional) Struktur yang mendefinisikan label untuk ditampilkan pada grid status. Struktur label berisi informasi berikut:

showValue

(Opsional) Apakah akan menampilkan unit dan nilai untuk setiap properti aset dalam widget ini atau tidak. Default ke `true`.

Example Contoh definisi dasbor

Contoh berikut mendefinisikan dasbor dari payload yang disimpan dalam file JSON.

```
aws iotsitewise create-dashboard \
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE \
```

```
--dashboard-name "Wind Farm Dashboard" \  
--dashboard-definition file://dashboard-definition.json
```

Contoh JSON berikut untuk `dashboard-definition.json` mendefinisikan dasbor dengan widget visualisasi berikut:

- Bagan garis yang memvisualisasikan total tenaga ladang angin di kiri atas dasbor. Bagan garis ini mencakup ambang batas yang menunjukkan kapan ladang angin menghasilkan daya yang lebih kecil daripada output minimum yang diharapkan. Grafik garis ini juga mencakup garis tren regresi linier.
- Diagram batang yang memvisualisasikan kecepatan angin untuk empat turbin di kanan atas dasbor.

Note

Contoh ini mewakili visualisasi diagram garis dan batang di dasbor. Dasbor ini mirip dengan [contoh dasbor ladang angin](#).

```
{  
  "widgets": [  
    {  
      "type": "sc-line-chart",  
      "title": "Total Average Power",  
      "x": 0,  
      "y": 0,  
      "height": 3,  
      "width": 3,  
      "metrics": [  
        {  
          "label": "Power",  
          "type": "iotsitewise",  
          "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
          "propertyId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",  
          "analysis": {  
            "trends": [  
              {  
                "type": "linear-regression"  
              }  
            ]  
          }  
        }  
      ]  
    }  
  ]  
}
```

```
    }
  }
],
"annotations": {
  "y": [
    {
      "comparisonOperator": "LT",
      "value": 20000,
      "color": "#D13212",
      "showValue": true
    }
  ]
}
},
{
  "type": "sc-bar-chart",
  "title": "Wind Speed",
  "x": 3,
  "y": 3,
  "height": 3,
  "width": 3,
  "metrics": [
    {
      "label": "Turbine 1",
      "type": "iotsitewise",
      "assetId": "a1b2c3d4-5678-90ab-cdef-2a2a2EXAMPLE",
      "propertyId": "a1b2c3d4-5678-90ab-cdef-55555EXAMPLE"
    },
    {
      "label": "Turbine 2",
      "type": "iotsitewise",
      "assetId": "a1b2c3d4-5678-90ab-cdef-2b2b2EXAMPLE",
      "propertyId": "a1b2c3d4-5678-90ab-cdef-55555EXAMPLE"
    },
    {
      "label": "Turbine 3",
      "type": "iotsitewise",
      "assetId": "a1b2c3d4-5678-90ab-cdef-2c2c2EXAMPLE",
      "propertyId": "a1b2c3d4-5678-90ab-cdef-55555EXAMPLE"
    },
    {
      "label": "Turbine 4",
      "type": "iotsitewise",
      "assetId": "a1b2c3d4-5678-90ab-cdef-2d2d2EXAMPLE",
```

```
        "propertyId": "a1b2c3d4-5678-90ab-cdef-55555EXAMPLE"  
      }  
    ]  
  }  
]  
}
```

Mengaktifkan alarm untuk portal Anda

Anda dapat mengaktifkan fitur alarm yang didukung oleh AWS IoT Events untuk portal Anda sehingga administrator portal dapat membuat, mengedit, dan menghapus model AWS IoT Events alarm di portal Monitor Anda SiteWise . Pemilik proyek dapat mengonfigurasi alarm. Pemirsa proyek dapat melihat detail alarm. Bagian ini menjelaskan bagaimana Anda dapat menggunakan AWS IoT SiteWise konsol untuk mengaktifkan fitur alarm untuk portal Anda.

Important

- Anda tidak dapat membuat alarm eksternal di portal Anda.
- Jika Anda ingin mengirim pemberitahuan alarm, Anda harus memilih Pusat Identitas IAM untuk layanan otentikasi pengguna.
- Fitur pemberitahuan alarm tidak tersedia di China (Beijing) Wilayah AWS.

Saat Anda mengonfigurasi dan membuat portal, Anda dapat mengaktifkan alarm dan pemberitahuan alarm di Langkah 2 Fitur tambahan. Berdasarkan layanan otentikasi pengguna, pilih salah satu opsi berikut:

IAM Identity Center

AWS IoT SiteWise > Monitor > Portals > Create portal

Step 1
Portal configuration

Step 2- optional
Additional features

Step 3
Invite administrators

Step 4
Assign users

Additional features - *optional*

Alarms

Your portal users can create alarms in the portal to monitor equipment or processes. They can also get notified when the equipment or processes perform outside specified range.

Enable alarms
If enabled, your portal users can define AWS IoT Events alarms in SiteWise Monitor.

AWS IoT SiteWise access role
Choose an IAM role that allows AWS IoT Events to send data to AWS IoT SiteWise. To edit the role, go to the [IAM console](#).

Create a role from an AWS managed template

Use an existing role

Enable alarm notifications
If enabled, alarms can send email or SMS notifications.

Sender
Specify the email address that sends alarm notifications. To edit or add a sender, go to the [Amazon SES console](#).

AWS Lambda role
Choose an IAM role that allows AWS Lambda to send data to Amazon SES and Amazon SNS. To edit the role, go to the [IAM console](#).

Create a role from an AWS managed template

Use an existing role

AWS Lambda function
Choose an AWS Lambda function to manage alarm notifications. To edit the function, go to the [AWS Lambda console](#).

Create a lambda from an AWS managed template

Use an existing lambda

[Previous](#) [Create](#)

Untuk mengaktifkan alarm untuk portal

1. (Opsional) Pilih Aktifkan alarm.
 - Untuk peran AWS IoT SiteWise akses, gunakan peran yang ada atau buat peran dengan izin yang diperlukan. Peran ini membutuhkan `iotevents:BatchPutMessage` izin dan hubungan kepercayaan yang memungkinkan `iot.amazonaws.com` dan `iotevents.amazonaws.com` untuk mengambil peran.
2. (Opsional) Pilih Aktifkan pemberitahuan alarm.
 - a. Untuk Pengirim, pilih pengirim.

⚠ Important

Anda harus memverifikasi alamat email pengirim di Amazon SES. Untuk informasi selengkapnya, lihat [Memverifikasi alamat email di Amazon SES](#), di Panduan Pengembang Layanan Email Sederhana Amazon.

- b. Untuk AWS Lambda peran, gunakan peran yang ada atau buat peran dengan izin yang diperlukan. Peran ini membutuhkan `lambda:InvokeFunction` dan `sso-directory:DescribeUser` izin dan hubungan kepercayaan yang memungkinkan `iotevents.amazonaws.com` dan `lambda.amazonaws.com` untuk mengambil peran.
- c. Untuk AWS Lambda fungsi, pilih fungsi Lambda yang ada atau buat fungsi yang mengelola notifikasi alarm. Untuk informasi selengkapnya, lihat [Mengelola notifikasi alarm](#) di Panduan AWS IoT Events Pengembang.

IAM

Untuk mengaktifkan alarm untuk portal

- (Opsional) Pilih Aktifkan alarm.

- Untuk peran AWS IoT SiteWise akses, gunakan peran yang ada atau buat peran dengan izin yang diperlukan. Peran ini membutuhkan `iotevents:BatchPutMessage` izin dan hubungan kepercayaan yang memungkinkan `iot.amazonaws.com` dan `iotevents.amazonaws.com` untuk mengambil peran.

Untuk informasi selengkapnya tentang alarm di SiteWise Monitor, lihat [Pemantauan dengan alarm](#) di Panduan AWS IoT SiteWise Aplikasi.

Mengaktifkan portal Anda di tepi

Setelah Anda mengaktifkan portal Anda di tepi, portal ini tersedia di semua gateway SiteWise Edge dengan paket pemrosesan data diaktifkan di akun Anda.

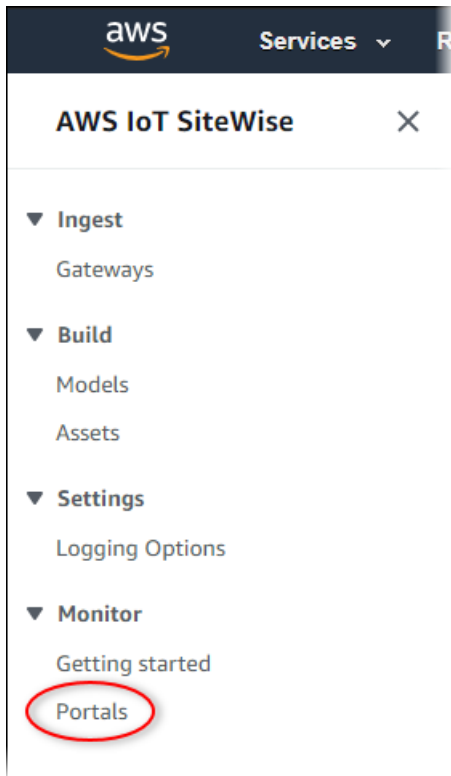
Untuk mengaktifkan portal di tepi

1. Di bagian konfigurasi Edge, nyalakan Aktifkan portal ini di tepi.
2. Pilih Buat.

Mengelola portal SiteWise Monitor Anda

Anda mungkin perlu memperbarui detail portal, mengubah administrator, atau menambahkan pengguna ke portal Anda. Bagian ini menjelaskan bagaimana Anda dapat menyelesaikan tugas-tugas administratif dasar ini untuk portal SiteWise Monitor Anda.

1. Masuk ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Monitor, Portal.



3. Pilih portal, lalu pilih Lihat detail (atau pilih Nama portal).
4. Anda dapat melakukan salah satu tugas administratif berikut:
 - [Mengubah nama portal, deskripsi, branding, email dukungan, dan izin](#)
 - [Menambahkan atau menghapus administrator portal](#)
 - [Mengirim undangan email ke administrator portal](#)
 - [Menambahkan atau menghapus pengguna portal](#)
 - [Menghapus portal](#)

Untuk informasi tentang cara membuat portal, lihat [Memulai dengan AWS IoT SiteWise Monitor](#).

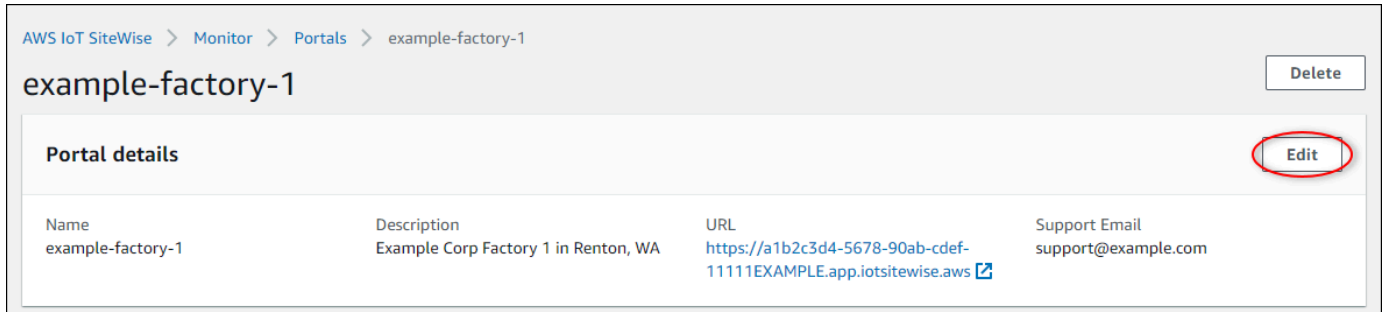
Topik

- [Mengubah nama portal, deskripsi, branding, email dukungan, dan izin](#)
- [Menambahkan atau menghapus administrator portal](#)
- [Mengirim undangan email ke administrator portal](#)
- [Menambahkan atau menghapus pengguna portal](#)
- [Menghapus portal](#)

Mengubah nama portal, deskripsi, branding, email dukungan, dan izin

Anda dapat mengubah nama portal, deskripsi, branding, email dukungan, dan izin.

1. Pada halaman detail portal, di bagian Detail portal, pilih Edit.

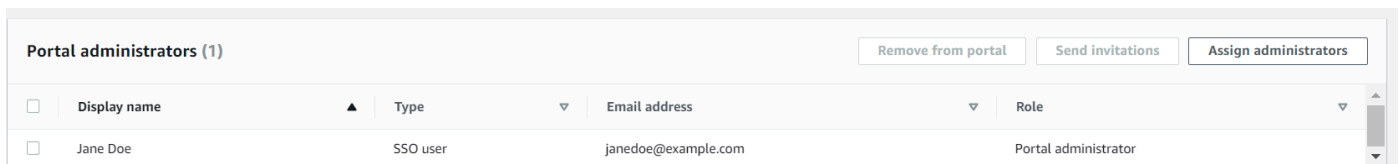


2. Perbarui Nama, Deskripsi, pencitraan merek Portal, email kontak Dukungan, atau Izin.
3. Setelah selesai, pilih Simpan.

Menambahkan atau menghapus administrator portal

Dalam beberapa langkah, Anda dapat menambah atau menghapus pengguna sebagai administrator untuk portal. Berdasarkan layanan otentikasi pengguna, pilih salah satu opsi berikut.

IAM Identity Center



Untuk menambahkan administrator portal

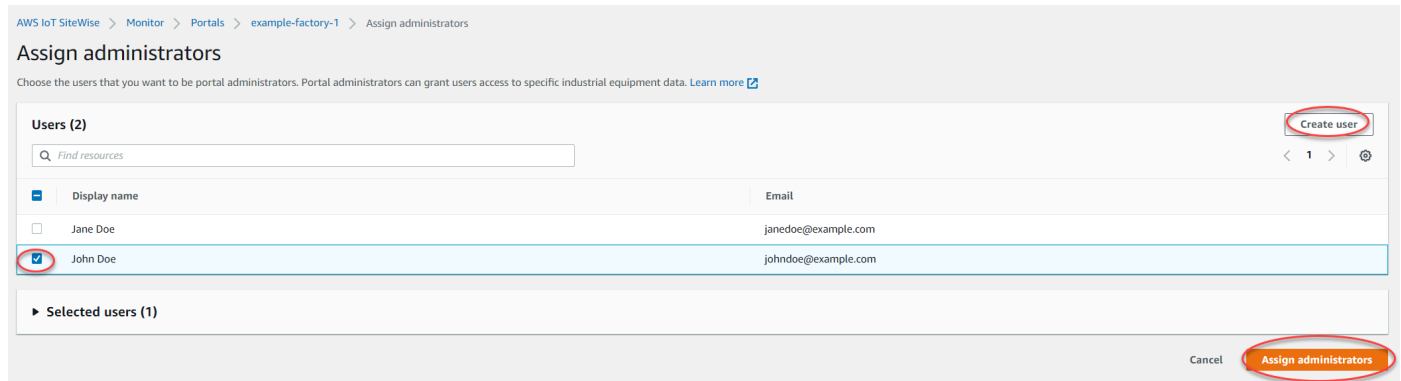
1. Pada halaman detail portal, di bagian Administrator portal, pilih Tetapkan administrator.
2. Pada halaman Tetapkan administrator, pilih kotak centang bagi pengguna untuk ditambahkan ke portal sebagai administrator.

Note

Jika Anda menggunakan Pusat Identitas IAM sebagai penyimpanan identitas, dan Anda masuk ke akun AWS Organizations manajemen, Anda dapat memilih Buat pengguna untuk membuat pengguna Pusat Identitas IAM. IAM Identity Center

mengirimkan email kepada pengguna baru untuk mengatur kata sandi mereka. Anda kemudian dapat menetapkan pengguna ke portal sebagai administrator. Untuk informasi selengkapnya, lihat [Mengelola identitas di Pusat Identitas IAM](#).

3. Pilih Tetapkan administrator.



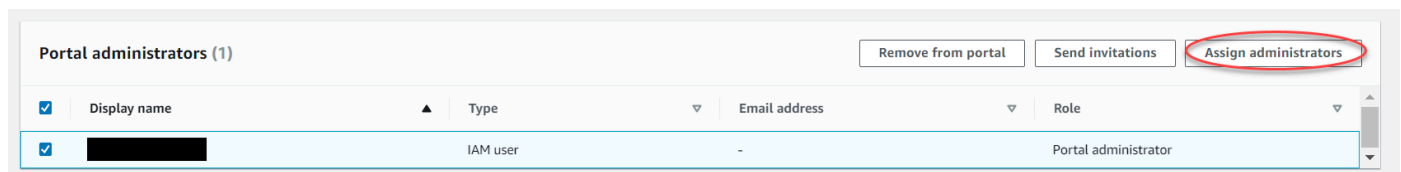
Untuk menghapus administrator portal

- Pada halaman detail portal, di bagian Administrator portal, pilih kotak centang untuk dihapus setiap pengguna, lalu pilih Hapus dari portal.

Note

Kami menyarankan Anda memilih setidaknya satu administrator portal.

IAM



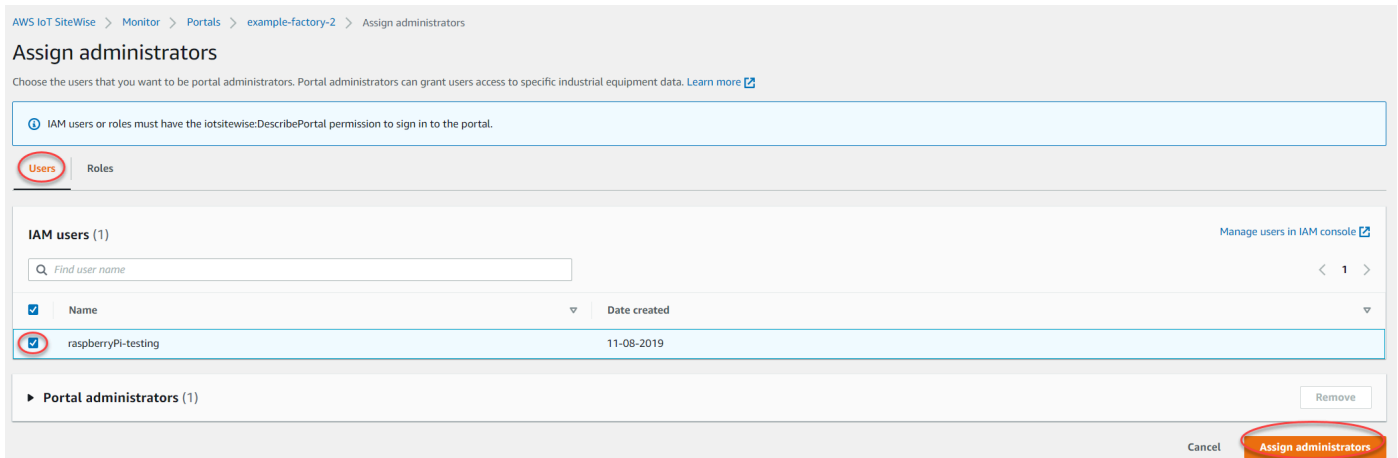
Untuk menambahkan administrator portal

1. Pada halaman detail portal, di bagian Administrator portal, pilih Tetapkan administrator.
2. Pada halaman Tetapkan administrator, lakukan hal berikut:
 - Pilih pengguna IAM, jika Anda ingin menambahkan pengguna IAM sebagai administrator portal Anda.

- Pilih peran IAM, jika Anda ingin menambahkan peran IAM sebagai administrator portal Anda.
3. Pilih kotak centang untuk pengguna atau peran yang Anda inginkan sebagai administrator portal Anda. Ini menambahkan pengguna atau peran ke daftar administrator Portal.
 4. Pilih Tetapkan administrator.

Important

Pengguna atau peran harus memiliki `iotsitewise:DescribePortal` izin untuk masuk ke portal.



AWS IoT SiteWise > Monitor > Portals > example-factory-2 > Assign administrators

Assign administrators

Choose the users that you want to be portal administrators. Portal administrators can grant users access to specific industrial equipment data. [Learn more](#)

Users Roles

IAM users (1) [Manage users in IAM console](#)

Find user name

<input checked="" type="checkbox"/>	Name	Date created
<input checked="" type="checkbox"/>	raspberrypi-testing	11-08-2019

Portal administrators (1) [Remove](#)

Cancel [Assign administrators](#)

AWS IoT SiteWise > Monitor > Portals > example-factory-2 > Assign administrators

Assign administrators

Choose the users that you want to be portal administrators. Portal administrators can grant users access to specific industrial equipment data. [Learn more](#)

ⓘ IAM users or roles must have the `lotsitewise:DescribePortal` permission to sign in to the portal.

Users **Roles**

IAM roles (66) [Manage roles in IAM console](#)

Find role name

<input type="checkbox"/>	Name	Date created
<input type="checkbox"/>	[REDACTED]	
<input checked="" type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_4wZigNpA1	03-16-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_ECKT-2Oar	03-11-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_GTnd0O4Wr	03-16-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_rHINLNC5-	03-11-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_XB330QUIO	03-10-2021
<input type="checkbox"/>	[REDACTED]	
<input type="checkbox"/>	[REDACTED]	
<input type="checkbox"/>	[REDACTED]	

► Portal administrators (2) [Remove](#)

Cancel **Assign administrators**

Untuk menghapus administrator portal

- Pada halaman detail portal, di bagian Administrator portal, pilih kotak centang untuk dihapus setiap pengguna, lalu pilih Hapus dari portal.

Note

Meninggalkan portal tanpa administrator portal tidak disarankan.

Mengirim undangan email ke administrator portal

Anda dapat mengirim undangan email ke administrator portal.

- Pada halaman detail portal, di bagian Administrator portal, pilih kotak centang untuk administrator portal.

Portal administrators (1) [Remove from portal](#) **Send invitations** [Assign users](#)

<input checked="" type="checkbox"/>	Display name	Email address	Role
<input checked="" type="checkbox"/>	John Doe	john.doe@example.com	Portal administrator

- Pilih Kirim undangan. Klien email Anda terbuka, dan undangan diisi di badan pesan.

Anda dapat menyesuaikan email sebelum mengirimkannya ke administrator portal Anda.

Menambahkan atau menghapus pengguna portal

Anda memilih pengguna mana yang memiliki akses ke portal Anda. Pengguna portal muncul dalam daftar pengguna dalam portal SiteWise Monitor. Dari daftar ini, administrator portal dapat menambahkan pemilik proyek, dan pemilik proyek dapat menambahkan pemirsa proyek.

Note

Administrator portal dan pengguna portal Anda mungkin menghubungi Anda melalui email dukungan portal jika mereka membutuhkan Anda untuk menambah atau menghapus pengguna.

Berdasarkan layanan otentikasi pengguna, pilih salah satu opsi berikut.

IAM Identity Center

Portal users (1)					Remove from portal	Assign users
<input type="checkbox"/>	Display name	Type	Email address	Role		
<input type="checkbox"/>	John Doe	SSO user	johndoe@example.com	Portal viewer		

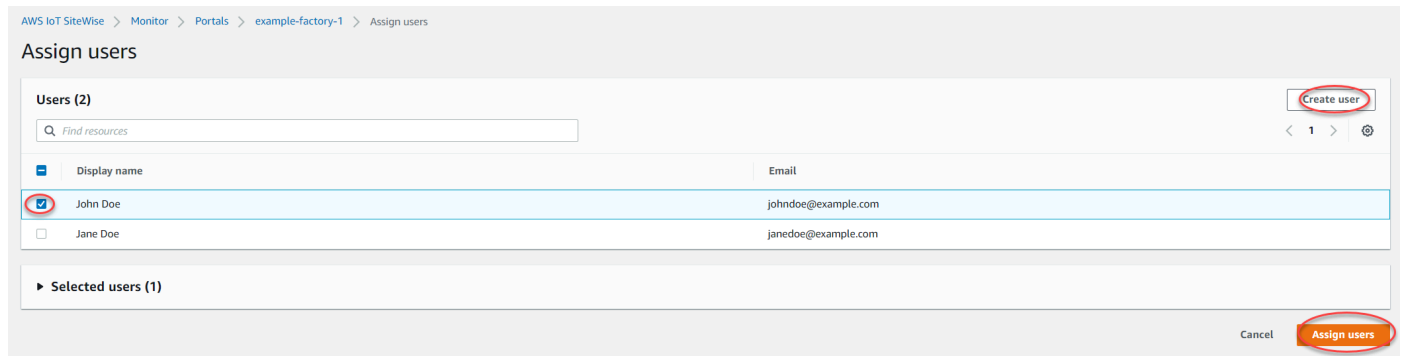
Untuk menambahkan pengguna portal

1. Pada halaman detail portal, di bagian Pengguna portal, pilih Tetapkan pengguna.
2. Pada halaman Tetapkan pengguna, pilih kotak centang untuk ditambahkan pengguna ke portal.

Note

Jika Anda menggunakan Pusat Identitas IAM sebagai penyimpanan identitas, dan Anda masuk ke akun AWS Organizations manajemen, Anda dapat memilih Buat pengguna untuk membuat pengguna Pusat Identitas IAM. IAM Identity Center mengirimkan email kepada pengguna baru untuk mengatur kata sandi mereka. Anda kemudian dapat menetapkan pengguna ke portal sebagai pengguna. Untuk informasi selengkapnya, lihat [Mengelola identitas di Pusat Identitas IAM](#).

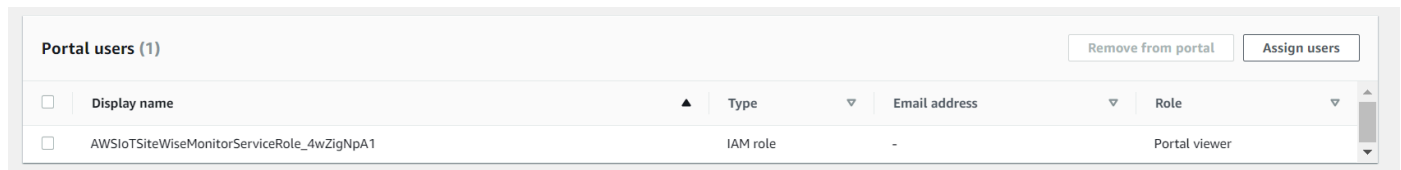
3. Pilih Tetapkan pengguna.



Untuk menghapus pengguna portal

- Pada halaman detail portal, di bagian Pengguna portal, pilih kotak centang untuk dihapus pengguna dari portal, lalu pilih Hapus dari portal.

IAM



Untuk menambahkan pengguna portal

1. Pada halaman detail portal, di bagian Pengguna portal, pilih Tetapkan pengguna.
2. Pada halaman Tetapkan pengguna, lakukan hal berikut:
 - Pilih pengguna IAM untuk menambahkan pengguna IAM sebagai pengguna portal Anda.
 - Pilih peran IAM untuk menambahkan peran IAM sebagai pengguna portal Anda.
3. Pilih kotak centang untuk pengguna atau peran yang ingin Anda tambahkan sebagai pengguna portal Anda. Ini menambahkan pengguna atau peran ke daftar pengguna Portal.
4. Pilih Tetapkan pengguna.

AWS IoT SiteWise > Monitor > Portals > example-factory-2 > Assign users

Assign users

Users Roles

IAM users (1) [Manage users in IAM console](#)

Find user name

<input checked="" type="checkbox"/>	Name	Date created
<input checked="" type="checkbox"/>	[REDACTED]	11-08-2019

Portal users (1) [Remove](#)

Cancel [Assign users](#)

AWS IoT SiteWise > Monitor > Portals > example-factory-2 > Assign users

Assign users

Users **Roles**

IAM roles (66) [Manage roles in IAM console](#)

Find role name

<input checked="" type="checkbox"/>	Name	Date created
<input type="checkbox"/>	[REDACTED]	
<input checked="" type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_4wZigNpA1	03-16-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_ECKT-2Oar	03-11-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_GTnd004Wr	03-16-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_rHINLNC5-	03-11-2021
<input type="checkbox"/>	AWSIoTSiteWiseMonitorServiceRole_XB330QUIO	03-10-2021
<input type="checkbox"/>	[REDACTED]	
<input type="checkbox"/>	[REDACTED]	
<input type="checkbox"/>	[REDACTED]	

Portal users (2) [Remove](#)

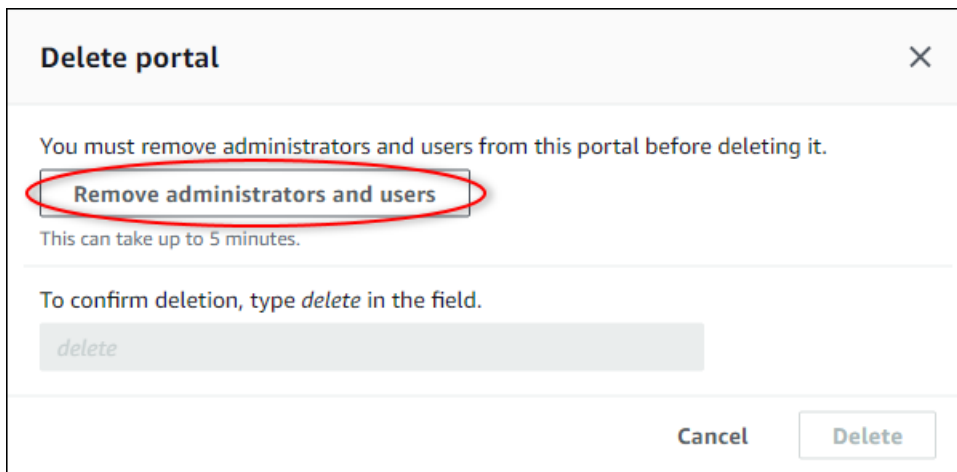
Cancel [Assign users](#)

Untuk menghapus pengguna portal

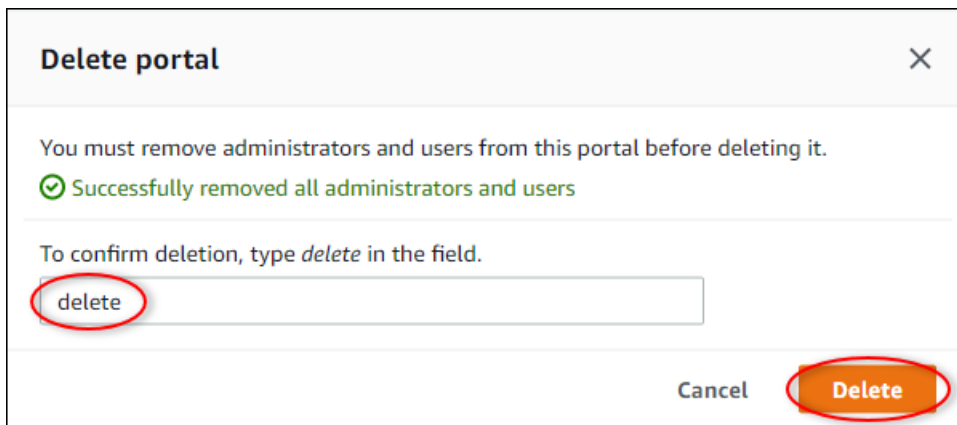
- Pada halaman detail portal, di bagian Pengguna portal, pilih kotak centang untuk dihapus pengguna dari portal, lalu pilih Hapus dari portal.

⚠ Important

Pengguna atau peran harus memiliki `iotsitewise:DescribePortal` izin untuk masuk ke portal.



3. Jika Anda yakin ingin menghapus seluruh portal, masukkan **delete** di bidang untuk mengonfirmasi penghapusan.



4. Pilih Hapus.

Memantau data dengan aplikasi dasbor IoT

Aplikasi dasbor IoT adalah aplikasi dasbor open source tempat Anda dapat memvisualisasikan dan berinteraksi dengan data operasional. Anda dapat memanfaatkan aplikasi dasbor IoT AWS Cloud Development Kit (AWS CDK) untuk menyebarkan.

Berikut ini adalah contoh fitur visualisasi data yang dapat disesuaikan dalam aplikasi dasbor IoT:

- Support untuk beberapa properti dalam satu bagan garis.
- Peningkatan pencarian aset dan properti.

Pelanggan dari manufaktur, logistik, energi, dan industri lainnya dapat menggunakan aplikasi dasbor IoT untuk mengatasi tantangan spesifik seperti melacak kinerja peralatan, mengoptimalkan efisiensi operasional, dan keputusan berbasis data. Untuk informasi selengkapnya, lihat [GitHub repositori untuk aplikasi dasbor IoT](#).

Data kueri dari AWS IoT SiteWise

Anda dapat menggunakan operasi AWS IoT SiteWise API untuk menanyakan nilai saat ini, nilai historis, dan agregat properti aset Anda selama interval waktu tertentu.

Gunakan fitur ini untuk mendapatkan wawasan tentang data Anda. Misalnya, temukan semua aset Anda dengan nilai properti tertentu atau buat representasi kustom data Anda. Anda juga dapat menggunakan operasi API untuk mengembangkan solusi perangkat lunak yang terintegrasi dengan data industri yang disimpan dalam AWS IoT SiteWise aset Anda. Anda juga dapat menjelajahi data aset Anda secara langsung AWS IoT SiteWise Monitor. Untuk mempelajari cara mengkonfigurasi SiteWise Monitor, lihat [Memantau data dengan AWS IoT SiteWise Monitor](#).

Operasi yang dijelaskan dalam bagian ini mengembalikan objek nilai properti yang berisi struktur stempel waktu, kualitas, nilai (TQV):

- `timestamp` Berisi waktu epoch Unix saat ini dalam hitungan detik dengan offset nanodetik.
- `quality` Berisi salah satu string berikut yang menunjukkan kualitas titik data:
 - `GOOD`— Data tidak terpengaruh oleh masalah apa pun.
 - `BAD`— Data dipengaruhi oleh masalah seperti kegagalan sensor.
 - `UNCERTAIN`— Data dipengaruhi oleh masalah seperti ketidakakuratan sensor.
- `value` Berisi salah satu bidang berikut, tergantung pada jenis properti:
 - `booleanValue`
 - `doubleValue`
 - `integerValue`
 - `stringValue`

Topik

- [Menanyakan nilai properti aset saat ini](#)
- [Menanyakan nilai properti aset historis](#)
- [Kueri agregat properti aset](#)
- [AWS IoT SiteWise bahasa query](#)

Menanyakan nilai properti aset saat ini

Tutorial ini menunjukkan dua cara untuk mendapatkan nilai saat ini dari properti aset. Anda dapat menggunakan AWS IoT SiteWise konsol atau menggunakan API di AWS Command Line Interface (AWS CLI).

Topik

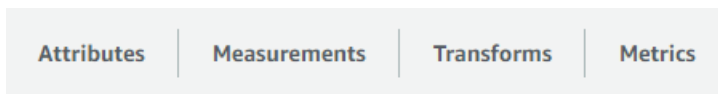
- [Kueri nilai saat ini properti aset \(konsol\)](#)
- [Kueri nilai saat ini properti aset \(AWS CLI\)](#)

Kueri nilai saat ini properti aset (konsol)

Anda dapat menggunakan AWS IoT SiteWise konsol untuk melihat nilai saat ini dari properti aset.

Untuk mendapatkan nilai saat ini dari properti aset (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset dengan properti untuk kueri.
4. Pilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.
5. Pilih tab untuk jenis properti. Misalnya, pilih Pengukuran untuk melihat nilai saat ini dari properti pengukuran.



6. Temukan properti untuk dilihat. Nilai saat ini muncul di kolom Nilai terbaru.

Kueri nilai saat ini properti aset (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk menanyakan nilai saat ini dari properti aset.

Gunakan [GetAssetPropertyValue](#) operasi untuk menanyakan nilai saat ini properti aset.

Untuk mengidentifikasi properti aset, tentukan salah satu dari berikut ini:

- Properti `propertyId` aset yang dikirimkan data ke. `assetId`

- The `propertyAlias`, yang merupakan alias aliran data (misalnya, `/company/windfarm/3/turbine/7/temperature`). Untuk menggunakan opsi ini, Anda harus terlebih dahulu menetapkan alias properti aset Anda. Untuk mengatur alias properti, lihat [Memetakan aliran data industri ke properti aset](#).

Untuk mendapatkan nilai saat ini dari properti aset (AWS CLI)

- Jalankan perintah berikut untuk mendapatkan nilai properti aset saat ini. Ganti `asset-id` dengan ID aset dan `property-id` dengan ID properti.

```
aws iotsitewise get-asset-property-value \  
  --asset-id asset-id \  
  --property-id property-id
```

Operasi mengembalikan respons yang berisi TQV saat ini dari properti dalam format berikut.

```
{  
  "propertyValue": {  
    "value": {  
      "booleanValue": Boolean,  
      "doubleValue": Number,  
      "integerValue": Number,  
      "stringValue": "String"  
    },  
    "timestamp": {  
      "timeInSeconds": Number,  
      "offsetInNanos": Number  
    },  
    "quality": "String"  
  }  
}
```

Menanyakan nilai properti aset historis

Anda dapat menggunakan [GetAssetPropertyValueHistory](#) operasi AWS IoT SiteWise API untuk menanyakan nilai historis properti aset.

Untuk mengidentifikasi properti aset, tentukan salah satu dari berikut ini:

- Properti `propertyId` aset yang dikirimkan data ke. `assetId`
- `ThepropertyAlias`, yang merupakan alias aliran data (misalnya, `/company/windfarm/3/turbine/7/temperature`). Untuk menggunakan opsi ini, Anda harus terlebih dahulu menetapkan alias properti aset Anda. Untuk mengatur alias properti, lihat [Memetakan aliran data industri ke properti aset](#).

Lulus parameter berikut untuk menyempurnakan hasil Anda:

- `startDate`— Awal eksklusif dari rentang dari mana untuk menanyakan data historis, dinyatakan dalam detik dalam waktu zaman Unix.
- `endDate`— Akhir inklusif dari rentang dari mana untuk menanyakan data historis, dinyatakan dalam detik dalam waktu zaman Unix.
- `maxResults`— Jumlah maksimum hasil untuk dikembalikan dalam satu permintaan. Default untuk hasil. 20
- `nextToken`— Token pagination dikembalikan dari panggilan sebelumnya dari operasi ini.
- `timeOrdering`— Urutan untuk diterapkan pada nilai yang dikembalikan: ASCENDING atau DESCENDING.
- `qualities`— Kualitas untuk memfilter hasil dengan: GOOD, BAD, atau UNCERTAIN.

Topik

- [Kueri riwayat nilai untuk properti aset \(AWS CLI\)](#)

Kueri riwayat nilai untuk properti aset (AWS CLI)

Untuk menanyakan riwayat nilai untuk properti aset (AWS CLI)

1. Jalankan perintah berikut untuk mendapatkan riwayat nilai untuk properti aset. Perintah ini menanyakan riwayat properti selama interval 10 menit tertentu. Ganti `asset-id` dengan ID aset dan `property-id dengan ID` properti. Ganti parameter tanggal dengan interval untuk kueri.

```
aws iotsitewise get-asset-property-value-history \  
  --asset-id asset-id \  
  --property-id property-id \  
  --start-date 1575216000 \  
  --end-date 1575216600
```

Operasi mengembalikan respons yang berisi TQV historis properti dalam format berikut:

```
{
  "assetPropertyValueHistory": [
    {
      "value": {
        "booleanValue": Boolean,
        "doubleValue": Number,
        "integerValue": Number,
        "stringValue": "String"
      },
      "timestamp": {
        "timeInSeconds": Number,
        "offsetInNanos": Number
      },
      "quality": "String"
    }
  ],
  "nextToken": "String"
}
```

2. Jika ada lebih banyak entri nilai, Anda dapat meneruskan token pagination dari `nextToken` bidang ke panggilan berikutnya ke operasi. [GetAssetPropertyValueHistory](#)

Kueri agregat properti aset

AWS IoT SiteWise secara otomatis menghitung nilai properti aset agregat, yang merupakan seperangkat metrik dasar yang dihitung selama beberapa interval waktu. AWS IoT SiteWise menghitung agregat berikut setiap menit, jam, dan hari untuk properti aset Anda:

- rata-rata — Rata-rata (mean) dari nilai properti selama interval waktu.
- count — Jumlah titik data untuk properti selama interval waktu.
- maksimum — Maksimum nilai properti selama interval waktu.
- minimum — Minimal nilai properti selama interval waktu.
- standar deviasi — Standar deviasi nilai properti selama interval waktu.
- jumlah — Jumlah nilai properti selama interval waktu.

Untuk properti non-numerik, seperti string dan Boolean, hanya menghitung agregat AWS IoT SiteWise hitungan.

Anda juga dapat menghitung metrik kustom untuk data aset Anda. Dengan properti metrik, Anda menentukan agregasi yang spesifik untuk operasi Anda. Properti metrik menawarkan fungsi agregasi tambahan dan interval waktu yang tidak dihitung sebelumnya untuk API. AWS IoT SiteWise Untuk informasi selengkapnya, lihat [Menggabungkan data dari properti dan aset lainnya \(metrik\)](#).

Topik

- [Agregat untuk properti aset \(API\)](#)
- [Agregat untuk properti aset \(AWS CLI\)](#)

Agregat untuk properti aset (API)

Anda dapat menggunakan AWS IoT SiteWise API untuk mendapatkan agregat untuk properti aset.

Gunakan [GetAssetPropertyAggregates](#) operasi untuk menanyakan agregat properti aset.

Untuk mengidentifikasi properti aset, tentukan salah satu dari berikut ini:

- Properti `propertyId` aset yang dikirimkan data ke. `assetId`
- `ThepropertyAlias`, yang merupakan alias aliran data (misalnya, `/company/windfarm/3/turbine/7/temperature`). Untuk menggunakan opsi ini, Anda harus terlebih dahulu menetapkan alias properti aset Anda. Untuk mengatur alias properti, lihat [Memetakan aliran data industri ke properti aset](#).

Anda juga harus melewati parameter yang diperlukan berikut:

- `aggregateTypes`— Daftar agregat untuk diambil. Anda dapat menentukan salah satu dari `AVERAGE`, `COUNT`, `MAXIMUM`, `MINIMUM`, `STANDARD_DEVIATION`, dan `SUM`.
- `resolution`— Interval waktu untuk mengambil metrik: `1m` (1 menit), `1h` (1 jam), atau `1d` (1 hari).
- `startDate`— Awal eksklusif dari rentang dari mana untuk menanyakan data historis, dinyatakan dalam detik dalam waktu zaman Unix.
- `endDate`— Akhir inklusif dari rentang dari mana untuk menanyakan data historis, dinyatakan dalam detik dalam waktu zaman Unix.

Anda juga dapat meneruskan salah satu parameter berikut untuk menyempurnakan hasil Anda:

- `maxResults`— Jumlah maksimum hasil untuk dikembalikan dalam satu permintaan. Default untuk hasil. 20
- `nextToken`— Token pagination dikembalikan dari panggilan sebelumnya dari operasi ini.
- `timeOrdering`— Urutan untuk diterapkan pada nilai yang dikembalikan: ASCENDING atau DESCENDING.
- `qualities`— Kualitas untuk memfilter hasil dengan: GOOD, BAD, atau UNCERTAIN.

Note

[GetAssetPropertyAggregates](#) Operasi mengembalikan TQV dengan format yang berbeda dari operasi lain yang dijelaskan dalam bagian ini. `valueStruktur` berisi bidang untuk masing-masing permintaan. `aggregateTypes timestamp` Berisi waktu agregasi terjadi, dalam detik dalam waktu epoch Unix.

Agregat untuk properti aset ()AWS CLI

Untuk menanyakan agregat untuk properti aset ()AWS CLI

1. Jalankan perintah berikut untuk mendapatkan agregat untuk properti aset. Perintah ini menanyakan rata-rata dan jumlah dengan resolusi 1 jam untuk interval 1 jam tertentu. Ganti *asset-id* dengan ID aset dan *property-id dengan ID* properti. Ganti parameter dengan agregat dan interval untuk kueri.

```
aws iotsitewise get-asset-property-aggregates \
  --asset-id asset-id \
  --property-id property-id \
  --start-date 1575216000 \
  --end-date 1575219600 \
  --aggregate-types AVERAGE SUM \
  --resolution 1h
```

Operasi mengembalikan respons yang berisi TQV historis properti dalam format berikut. Respons hanya mencakup agregat yang diminta.

```
{
  "aggregatedValues": [
```

```
{
  "timestamp": Number,
  "quality": "String",
  "value": {
    "average": Number,
    "count": Number,
    "maximum": Number,
    "minimum": Number,
    "standardDeviation": Number,
    "sum": Number
  }
},
"nextToken": "String"
}
```

2. Jika ada lebih banyak entri nilai, Anda dapat meneruskan token pagination dari nextToken bidang ke panggilan berikutnya ke operasi. [GetAssetPropertyAggregates](#)

AWS IoT SiteWise bahasa query

Dengan operasi [ExecuteQuery](#) API pengambilan AWS IoT SiteWise data, Anda dapat mengambil informasi tentang definisi struktural deklaratif, dan data timeseries yang terkait dengannya, dari berikut ini:

- model
- aset
- ukuran
- metrik
- berubah
- agregat

Hal ini dapat dilakukan dengan SQL seperti pernyataan query, dalam satu permintaan API.

Note

Fitur ini tersedia di semua Wilayah di mana keduanya AWS IoT SiteWise dan AWS IoT TwinMaker tersedia, kecuali AWS GovCloud (AS-Barat).

Topik

- [Prasyarat](#)
- [Referensi bahasa kueri](#)

Prasyarat

AWS IoT SiteWise memerlukan izin untuk berintegrasi AWS IoT TwinMaker sehingga dapat mengatur dan memodelkan data industri.

Sebelum Anda dapat mengambil informasi tentang model, aset, pengukuran, metrik, transformasi, dan agregat, pastikan prasyarat berikut terpenuhi:

- Peran terkait layanan untuk keduanya AWS IoT SiteWise dan AWS IoT TwinMaker disiapkan di Akun AWS. Untuk informasi selengkapnya tentang peran terkait layanan, lihat [Menggunakan peran terkait layanan](#) di Panduan Pengguna IAM.
- AWS IoT SiteWise Integrasi yang diaktifkan untuk peran IAM Anda. Untuk informasi selengkapnya, lihat [Mengintegrasikan dan AWS IoT SiteWise AWS IoT TwinMaker](#).
- AWS IoT TwinMaker Ruang kerja dengan ID `IoTSiteWiseDefaultWorkspace` di akun Anda di Wilayah. Untuk informasi selengkapnya, lihat [Menggunakan](#) Panduan AWS IoT TwinMaker Pengguna. `IoTSiteWiseDefaultWorkspace`
- Baik mode harga bundel standar atau berjenjang untuk AWS IoT TwinMaker diaktifkan. Untuk informasi selengkapnya, lihat [AWS IoT TwinMaker Mengganti mode harga](#) di Panduan AWS IoT TwinMaker Pengguna.

Referensi bahasa kueri

AWS IoT SiteWise mendukung bahasa kueri yang kaya untuk bekerja dengan data Anda. Tipe data, operator, fungsi, dan konstruksi yang tersedia dijelaskan dalam topik berikut.

Lihat [Kueri contoh](#) untuk menulis kueri dengan bahasa AWS IoT SiteWise kueri.

Topik

- [Memahami pandangan](#)
- [Jenis data yang didukung](#)
- [Mengambil data dengan pernyataan SELECT](#)
- [Operator logis](#)

- [Operator perbandingan](#)
- [Kueri contoh](#)

Memahami pandangan

Bagian ini memberikan informasi untuk membantu Anda memahami tampilan AWS IoT SiteWise, seperti metadata proses dan data telemetry.

Tabel berikut memberikan nama tampilan dan deskripsi tampilan.

Model data

Lihat nama	Lihat deskripsi
aset	Berisi informasi tentang aset dan model derivasi.
aset_property	Berisi informasi tentang struktur properti aset.
raw_time_seri	Berisi data historis deret waktu.
terbaru_value_time_series	Berisi nilai terbaru dari deret waktu.
precomputed_aggregates	Berisi nilai properti aset agregat yang dihitung secara otomatis. Mereka adalah satu set metrik dasar yang dihitung selama beberapa interval waktu.

Tampilan berikut mencantumkan nama kolom untuk kueri bersama dengan data sampel.

Lihat:aset

aset_id	asset_name	asset_description	asset_model_id
88898498-0b8b-42b5-bf57-16180bc3d3a0	WindTurbine A	WindTurbine Aset A	17847250-5bf0-4f74-b775-cc03f05e7cb8
17847250-5bf0-4f74-b775-cc03f05e7cb8	Model Aset Turbin Angin	Merupakan turbin di ladang angin.	

Lihat:ASET_PROPERTY

property_id	aset_id	property_name	property_data_type	property_alias	asset_composite_model_id
b29be434-b000-4d74-b809-75287d83bcd6	88898498-0b8b-42b5-bf57-16180bc3d3a0	suhu motor	Ganda	Rochester2/44///Line-5/Bus-2/Machine-5/Temperature	
3b458f00-24e7-458a-b4e8-c6026eff654a	88898498-0b8b-42b5-bf57-16180bc3d3a0	arah angin	Ganda	/company/windfarm/3/turbine/7/winddirection	2f458n00-56e7-458h-b4e8-c6026eff985g

Lihat:RAW_TIME_SERIES

aset_id	property_id	property_alias	event_timestamp	kualitas	boolean_nilai	int_nilai	nilai_ganda	string_nilai
88898498-0b8b-42b5-bf57-16180bc3d3a0	b29be434-b000-4d74-b809-75287d83bcd6	Rochester2/44///Line-5/Bus-2/Machine-5/Temperature	157521960	BAGUS			115.0	
88898498-0b8b-42b5-bf57-16180bc3d3a0	3b458f00-24e7-458a-b4e8-c6026eff654a	/company/windfarm/3/turbine/7/winddirection	157521937	BAGUS			348.75	

aset_id	property_id	property_alias	event_timestamp	kualitas	boolean_nilai	int_nilai	nilai_ganda	string_nilai
bf57-16180bc3d3a	c6026eff654a	windfarm/3/turbine/7/winddirection						

Note

Anda harus menyertakan klausa filter pada event_timestamp kolom untuk menanyakan raw_time_series tampilan. Ini adalah filter yang diperlukan, dan kueri akan gagal tanpanya.

Example query

```
SELECT event_timestamp, double_value FROM raw_time_series WHERE event_timestamp > 1234567890
```

Lihat:terbaru_Value_Time_Series

aset_id	property_id	property_alias	event_timestamp	kualitas	boolean_nilai	int_nilai	nilai_ganda	string_nilai
888984980b8b-42b1-b4e8-bf57-16180bc3d3a	3b458f00-24e7-4581-24e7-4581-b4e8-c6026eff654a	/company/windfarm/3/turbine/7/winddirection	15752196	BAGUS			355.39	

Lihat:precomputed_aggregates

aset_id	property_id	property_alias	event_timestamp	resolusi	nilai_sum	count_value	nilai_rata-rata	maksimum nilai	minimum nilai	stdev_nilai
8889840b8b-42- <td>b29be4b000-4c- <td>Roches 2/44// 0 Li ne-5/ Bus- 2/ Machin -5/ Temper ature</td> <td>1575210</td> <td>15m</td> <td>1105.48</td> <td>15</td> <td>73.4</td> <td>80,6</td> <td>68</td> <td>3,64</td> </td>	b29be4b000-4c- <td>Roches 2/44// 0 Li ne-5/ Bus- 2/ Machin -5/ Temper ature</td> <td>1575210</td> <td>15m</td> <td>1105.48</td> <td>15</td> <td>73.4</td> <td>80,6</td> <td>68</td> <td>3,64</td>	Roches 2/44// 0 Li ne-5/ Bus- 2/ Machin -5/ Temper ature	1575210	15m	1105.48	15	73.4	80,6	68	3,64

Jenis data yang didukung

AWS IoT SiteWise bahasa query mendukung tipe data berikut.

Lihat:aset

Tipe data	Deskripsi
STRING	Sebuah string panjang maksimum 1024 byte.
INTEGER	Sebuah integer 32-bit yang ditandatangani dengan rentang dari. -2,147,483,648 to 2,147,483,647
DOUBLE	Nomor floating point dengan rentang dari -10 ¹⁰⁰ to 10 ¹⁰⁰ dan presisi IEEE 754 ganda.
BOOLEAN	true atau false.

Note

Data presisi ganda tidak tepat. Beberapa nilai tidak dikonversi dengan tepat, dan tidak akan mewakili semua bilangan real karena presisi terbatas. Data floating-point dalam kueri mungkin bukan nilai yang sama yang diwakili secara internal. Nilai dibulatkan jika presisi nomor input terlalu tinggi.

Mengambil data dengan pernyataan SELECT

SELECTPernyataan ini digunakan untuk mengambil data dari satu atau lebih tampilan. AWS IoT SiteWise mendukung pandangan yang implisitJOIN. Anda dapat membuat daftar tampilan untuk bergabung (dalam FROM klausa SELECT pernyataan), menggunakan koma untuk memisahkannya.

Example

Gunakan SELECT pernyataan berikut:

```
SELECT select_expr [, ...]
[ FROM from_item [, ...] ]
[ WHERE [LIKE condition ESCAPE condition] ]
```

Pada contoh sebelumnya, LIKE klausa menentukan kondisi pencarian dan penyaringan menggunakan kartu liar. AWS IoT SiteWise mendukung percentage (%) sebagai karakter kartu liar.

Example untuk digunakan % dalam kondisi:

```
Prefix search: String%
Infix search: %String%
Suffix search: %String
```

Example untuk mencari aset:

```
SELECT asset_name, asset_description FROM asset WHERE asset_name LIKE 'Wind%'
```


Example untuk mencari aset menggunakan kondisi ESCAPE:

```
SELECT asset_name, asset_description FROM asset WHERE asset_name LIKE 'room\%' ESCAPE
'\'
```

Operator logis

AWS IoT SiteWise mendukung operator logis berikut.

Operator logis

Operator	Deskripsi	Contoh
AND	TRUE jika kedua nilai itu benar	a AND b

Jika salah satu a atau b adalah FALSE, ekspresi sebelumnya dievaluasi menjadi false. Agar AND operator dapat mengevaluasi menjadi benar, baik a dan b harus benar.

Example

```
SELECT a.asset_name
FROM asset as a, latest_value_time_series as t
WHERE t.int_value > 30 AND t.event_timestamp > 1234567890
```

Operator perbandingan

AWS IoT SiteWise mendukung operator perbandingan berikut.

Operator logis

Operator	Deskripsi
<	Kurang dari
>	Lebih besar dari
<=	Kurang dari atau sama dengan
>=	Lebih besar dari atau sama dengan

Operator	Deskripsi
=	Setara
!=	Tidak sama

Kueri contoh

Pemfilteran metadata

Contoh berikut adalah untuk pemfilteran metadata dengan SELECT pernyataan dengan bahasa query: AWS IoT SiteWise

```
SELECT a.asset_name, p.property_name
FROM asset a, asset_property p
WHERE a.asset_id = p.asset_id AND a.asset_name LIKE '%windmill%'
```

Penyaringan nilai

Berikut ini adalah contoh pemfilteran nilai menggunakan SELECT pernyataan dengan bahasa AWS IoT SiteWise query:

```
SELECT a.asset_name FROM asset a, raw_time_series r
WHERE a.asset_id = r.asset_id AND r.int_value > 30 AND r.event_timestamp > 1234567890
AND r.event_timestamp < 1234567891
```

Berinteraksi dengan layanan lain AWS

AWS IoT SiteWise dapat mempublikasikan data aset ke broker pesan berlangganan publikasi AWS IoT MQTT, sehingga Anda dapat berinteraksi dengan data aset Anda dari layanan lain. AWS IoT SiteWise menetapkan setiap properti aset topik MQTT unik yang dapat Anda gunakan untuk merutekan data aset Anda ke AWS layanan lain menggunakan aturan Inti. AWS IoT Misalnya, Anda dapat mengonfigurasi aturan AWS IoT Core untuk melakukan tugas-tugas berikut:

- Identifikasi kegagalan peralatan dan beri tahu personel yang sesuai dengan mengirimkan data ke [AWS IoT Events](#).
- Historisasi data aset tertentu untuk digunakan dalam solusi perangkat lunak eksternal dengan mengirimkan data ke Amazon [DynamoDB](#).
- Hasilkan laporan mingguan dengan memicu suatu [AWS Lambda](#) fungsi.

Anda dapat mengikuti tutorial yang berjalan melalui langkah-langkah yang diperlukan untuk mengatur aturan yang menyimpan nilai properti di DynamoDB. Untuk informasi selengkapnya, lihat [Menerbitkan pembaruan nilai properti ke Amazon DynamoDB](#).

Untuk informasi selengkapnya tentang cara mengonfigurasi aturan, lihat [Aturan](#) di Panduan AWS IoT Pengembang.

Anda juga dapat menggunakan data dari AWS layanan lain kembali ke AWS IoT SiteWise. Untuk menelan data melalui tindakan AWS IoT SiteWise aturan, lihat [Menelan data menggunakan aturan AWS IoT Core](#).

Topik

- [Memahami topik MQTT properti aset](#)
- [Bekerja dengan pemberitahuan properti aset](#)
- [Ekspor data ke Amazon S3 dengan pemberitahuan properti aset](#)
- [Integrasi dengan Grafana](#)
- [Mengintegrasikan dan AWS IoT SiteWise AWS IoT TwinMaker](#)
- [Mendeteksi anomali peralatan dengan Amazon Lookout for Equipment](#)

Memahami topik MQTT properti aset

Setiap properti aset memiliki jalur topik MQTT unik dalam format berikut.

```
$aws/sitewise/asset-models/assetModelId/assets/assetId/properties/propertyId
```

Note

AWS IoT SiteWise tidak mendukung wildcard filter topik # (multi-level) di mesin aturan AWS IoT Inti. Anda dapat menggunakan wildcard + (single-level). Misalnya, Anda dapat menggunakan filter topik berikut untuk mencocokkan semua pembaruan untuk model aset tertentu.

```
$aws/sitewise/asset-models/assetModelId/assets/+ /properties/+
```

Untuk mempelajari lebih lanjut tentang wildcard filter topik, lihat [Topik](#) di Panduan Pengembang AWS IoT Inti.

Bekerja dengan pemberitahuan properti aset

Anda dapat mengaktifkan pemberitahuan properti untuk mempublikasikan pembaruan data aset AWS IoT Core, lalu menjalankan kueri pada data Anda. Dengan pemberitahuan properti aset, AWS IoT SiteWise berikan AWS CloudFormation templat yang dapat Anda gunakan untuk mengekspor AWS IoT SiteWise data ke Amazon S3.

Note

Data aset dikirim ke AWS IoT Core setiap kali diterima oleh AWS IoT SiteWise, terlepas dari apakah nilainya telah berubah.

Topik

- [Mengaktifkan pemberitahuan properti aset \(konsol\)](#)
- [Mengaktifkan notifikasi properti aset \(\)AWS CLI](#)
- [Menanyakan pesan pemberitahuan properti aset](#)

Mengaktifkan pemberitahuan properti aset (konsol)

Secara default, AWS IoT SiteWise tidak mempublikasikan pembaruan nilai properti. Anda dapat menggunakan AWS IoT SiteWise konsol untuk mengaktifkan notifikasi untuk properti aset.

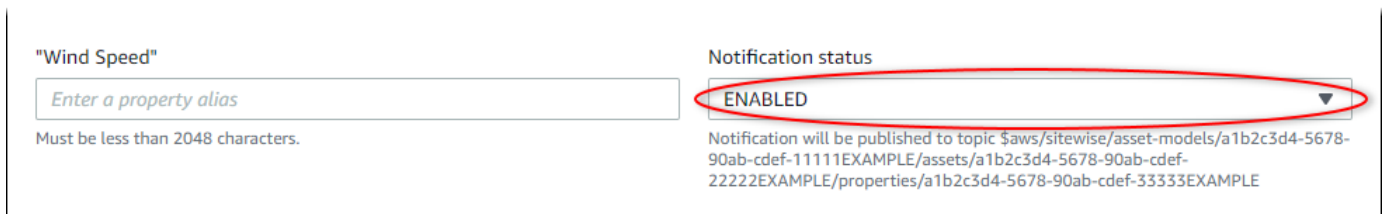
Untuk mengaktifkan atau menonaktifkan notifikasi untuk properti aset (konsol)

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset.
3. Pilih aset untuk mengaktifkan notifikasi properti.

Tip

Anda dapat memilih ikon panah untuk memperluas hierarki aset untuk menemukan aset Anda.

4. Pilih Edit.
5. Untuk status Pemberitahuan properti aset, pilih ENABLED.



The screenshot shows a form for editing a property named "Wind Speed". On the left, there is a text input field with the placeholder "Enter a property alias" and a note "Must be less than 2048 characters." On the right, there is a dropdown menu labeled "Notification status" which is currently set to "ENABLED". A red oval highlights the "ENABLED" option in the dropdown. Below the dropdown, there is a text string: "Notification will be published to topic \$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/assets/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE/properties/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE".

Anda juga dapat memilih DISABLED untuk menonaktifkan notifikasi untuk properti aset.

6. Pilih Simpan.

Mengaktifkan notifikasi properti aset ()AWS CLI

Secara default, AWS IoT SiteWise tidak mempublikasikan pembaruan nilai properti. Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk mengaktifkan atau menonaktifkan notifikasi untuk properti aset.

Anda harus mengetahui aset `assetId` dan properti Anda `propertyId` untuk menyelesaikan prosedur ini. Anda juga dapat menggunakan ID eksternal. Jika Anda membuat aset dan tidak mengetahuinya `assetId`, gunakan [ListAssets](#) API untuk mencantumkan semua aset untuk model tertentu. Gunakan [DescribeAsset](#) operasi untuk melihat properti aset Anda termasuk ID properti.

Gunakan [UpdateAssetProperty](#) operasi untuk mengaktifkan atau menonaktifkan notifikasi untuk properti aset. Tentukan parameter berikut:

- `assetId`— ID aset.
- `propertyId`— ID properti aset.
- `propertyNotificationState`— Status pemberitahuan nilai properti: `ENABLED` atau `DISABLED`.
- `propertyAlias`— Alias properti. Tentukan alias properti yang ada saat Anda memperbarui status notifikasi. Jika Anda menghilangkan parameter ini, alias properti yang ada akan dihapus.

Untuk mengaktifkan atau menonaktifkan pemberitahuan untuk properti aset (CLI)

1. Jalankan perintah berikut untuk mengambil alias properti aset. Ganti *asset-id* dengan ID aset dan *property-id dengan ID* properti.

```
aws iotsitewise describe-asset-property \  
  --asset-id asset-id \  
  --property-id property-id
```

Operasi mengembalikan respons yang berisi rincian properti aset dalam format berikut. Alias properti ada `assetProperty.alias` di objek JSON.

```
{  
  "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetName": "Wind Turbine 7",  
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
  "assetProperty": {  
    "id": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",  
    "name": "Wind Speed",  
    "alias": "/company/windfarm/3/turbine/7/windspeed",  
    "notification": {  
      "topic": "$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/  
assets/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE/properties/a1b2c3d4-5678-90ab-  
cdef-33333EXAMPLE",  
      "state": "DISABLED"  
    },  
    "dataType": "DOUBLE",  
    "unit": "m/s",  
    "type": {  
      "measurement": {}  
    }  
  }  
}
```

```
}  
}
```

2. Jalankan perintah berikut untuk mengaktifkan notifikasi untuk properti aset. Ganti *alias properti* dengan alias properti dari respons perintah sebelumnya, atau hilangkan `--property-alias` untuk memperbarui properti tanpa alias.

```
aws iotsitewise update-asset-property \  
  --asset-id asset-id \  
  --property-id property-id \  
  --property-notification-state ENABLED \  
  --property-alias property-alias
```

Anda juga dapat meneruskan `--property-notification-state DISABLED` untuk menonaktifkan notifikasi untuk properti aset.

Menanyakan pesan pemberitahuan properti aset

Untuk meminta pemberitahuan properti aset, buat AWS IoT Core aturan yang terdiri dari pernyataan SQL.

AWS IoT SiteWise menerbitkan pembaruan data properti aset ke AWS IoT Core dalam format berikut.

```
{  
  "type": "PropertyValueUpdate",  
  "payload": {  
    "assetId": "String",  
    "propertyId": "String",  
    "values": [  
      {  
        "timestamp": {  
          "timeInSeconds": Number,  
          "offsetInNanos": Number  
        },  
        "quality": "String",  
        "value": {  
          "booleanValue": Boolean,  
          "doubleValue": Number,  
          "integerValue": Number,  
          "stringValue": "String"  
        }  
      }  
    ]  
  }  
}
```

```

    }
  }
]
}
}

```

Setiap struktur dalam `values` daftar adalah struktur timestamp-quality-value (TQV).

- `timestamp` berisi waktu epoch Unix saat ini dalam hitungan detik dengan offset nanodetik.
- `quality` berisi salah satu string berikut yang menunjukkan kualitas titik data:
 - `GOOD`— Data tidak terpengaruh oleh masalah apa pun.
 - `BAD`— Data dipengaruhi oleh masalah seperti kegagalan sensor.
 - `UNCERTAIN`— Data dipengaruhi oleh masalah seperti ketidakakuratan sensor.
- `value` berisi salah satu bidang berikut, tergantung pada jenis properti:
 - `booleanValue`
 - `doubleValue`
 - `integerValue`
 - `stringValue`

Untuk mengurai nilai dari `values` array, Anda perlu menggunakan kueri objek bersarang yang kompleks dalam pernyataan SQL aturan Anda. Untuk informasi selengkapnya, lihat [Kueri objek bersarang](#) di Panduan AWS IoT Pengembang, atau lihat [Menerbitkan pembaruan nilai properti ke Amazon DynamoDB](#) tutorial untuk contoh spesifik penguraian pesan pemberitahuan properti aset.

Example Contoh query untuk mengekstrak array nilai

Pernyataan berikut menunjukkan bagaimana untuk query array nilai properti diperbarui untuk properti tipe ganda tertentu pada semua aset dengan properti tersebut.

```

SELECT
  (SELECT VALUE (value.doubleValue) FROM payload.values) AS windspeed
FROM
  '$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/assets/+/
properties/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE'
WHERE
  type = 'PropertyValueUpdate'

```

Pernyataan kueri aturan sebelumnya mengeluarkan data dalam format berikut.


```
{
  "windspeed": [
    26.32020195042838,
    26.282584572975477,
    26.352566977372508,
    26.283084346171442,
    26.571883739599322,
    26.60684140743005,
    26.628738636715045,
    26.273486932802125,
    26.436379105473964,
    26.600590095377303
  ]
}
```

Example Contoh kueri untuk mengekstrak satu nilai

Pernyataan berikut menunjukkan bagaimana untuk query nilai pertama dari array nilai properti untuk properti tipe ganda tertentu pada semua aset dengan properti tersebut.

```
SELECT
  get((SELECT VALUE (value.doubleValue) FROM payload.values), 0) AS windspeed
FROM
  '$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE/assets/+/
properties/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE'
WHERE
  type = 'PropertyValueUpdate'
```

Pernyataan kueri aturan sebelumnya mengeluarkan data dalam format berikut.

```
{
  "windspeed": 26.32020195042838
}
```

Important

Pernyataan kueri aturan ini mengabaikan pembaruan nilai selain yang pertama di setiap batch. Setiap batch dapat berisi hingga 10 nilai. Jika Anda perlu menyertakan nilai yang tersisa, Anda harus menyiapkan solusi yang lebih kompleks untuk menampilkan nilai properti aset ke layanan lain. Misalnya, Anda dapat mengatur aturan dengan AWS Lambda tindakan untuk mempublikasikan ulang setiap nilai dalam larik ke topik lain, dan menyiapkan aturan

lain untuk menanyakan topik tersebut dan mempublikasikan setiap nilai ke tindakan aturan yang diinginkan.

Ekspor data ke Amazon S3 dengan pemberitahuan properti aset

Anda dapat mengekspor data masuk dari AWS IoT SiteWise ke bucket Amazon S3 di akun Anda. Anda dapat mencadangkan data Anda dalam format yang dapat Anda gunakan untuk membuat laporan historis atau untuk menganalisis data Anda dengan metode yang kompleks.

Note

AWS IoT SiteWise juga mendukung penyimpanan tingkat dingin yang memungkinkan Anda menyimpan data dalam bucket Amazon S3 yang dikelola pelanggan. Untuk informasi selengkapnya tentang tingkatan penyimpanan yang didukung, lihat [Mengelola penyimpanan data](#).

AWS IoT SiteWise menyediakan fitur ini sebagai AWS CloudFormation template. Saat Anda membuat tumpukan dari template, AWS CloudFormation buat AWS sumber daya yang diperlukan untuk mengalirkan data yang masuk dari AWS IoT SiteWise bucket S3.

Kemudian, bucket S3 menerima semua data properti aset Anda yang dikirim dari pesan pembaruan nilai AWS IoT SiteWise properti. Bucket S3 juga menerima metadata aset Anda, yang mencakup nama aset dan properti serta informasi lainnya.

Untuk informasi selengkapnya tentang cara mengaktifkan pesan pembaruan nilai properti untuk properti aset yang akan diekspor ke Amazon S3, lihat [Berinteraksi dengan layanan lain AWS](#)

Fitur ini menyimpan data properti aset dan metadata aset Anda dalam format [Apache Parquet](#) di Amazon S3. Parquet adalah format data kolumnar yang menghemat ruang dan memungkinkan kueri lebih cepat dibandingkan dengan format berorientasi baris seperti JSON.


Note

Saat fitur ini mengambil metadata aset, fitur ini mendukung hingga sekitar 1.500 aset. Batasan ini hanya berlaku untuk metadata aset. Batasan ini tidak berlaku untuk jumlah aset yang didukung saat fitur mengekspor data properti aset.

Setiap nama sumber daya menyertakan awalan yang dapat Anda sesuaikan saat membuat tumpukan. Sumber daya meliputi:

- Bucket Amazon S3
- AWS Lambda fungsi
- AWS IoT Core Aturan
- AWS Identity and Access Management peran
- Aliran Amazon Data Firehose
- AWS Glue Database

Untuk daftar lengkap, lihat [Sumber daya yang dibuat dari template](#).

 Important

Anda akan dikenakan biaya untuk sumber daya yang dibuat dan dikonsumsi AWS CloudFormation template ini. Biaya ini termasuk penyimpanan data dan transfer data untuk beberapa AWS layanan.

Topik

- [Buat AWS CloudFormation tumpukan](#)
- [Lihat data Anda di Amazon S3](#)
- [Analisis data yang diekspor dengan Amazon Athena](#)
- [Sumber daya yang dibuat dari template](#)


Buat AWS CloudFormation tumpukan

Anda harus membuat tumpukan AWS CloudFormation untuk mengekspor data aset Anda ke Amazon S3.

Untuk mengekspor data ke Amazon S3

1. Buka [AWS CloudFormation template](#) dan masuk ke file AWS Management Console.
2. Pada halaman Buat tumpukan, pilih Berikutnya di bagian bawah halaman.

3. Pada halaman Tentukan detail tumpukan, masukkan bucket BucketName untuk S3 yang dibuat template ini untuk menerima data aset. Nama bucket ini harus unik secara global. Untuk informasi selengkapnya, lihat [Aturan untuk penamaan bucket](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.
4. (Opsional) Ubah salah satu parameter template lainnya:
 - GlobalResourcePrefix— Awalan untuk nama sumber daya global, seperti peran IAM, dibuat dari template ini.
 - LocalResourcePrefix— Awalan untuk nama sumber daya yang dibuat dari template ini di Wilayah saat ini.

 Note


Jika Anda membuat templat ini beberapa kali, Anda harus mengubah parameter nama bucket dan awalan sumber daya untuk menghindari konflik nama sumber daya.

5. Pilih Berikutnya.
6. Pada halaman Konfigurasi opsi tumpukan, pilih Berikutnya.
7. Di bagian bawah halaman, pilih kotak centang yang mengatakan saya mengakui bahwa AWS CloudFormation mungkin membuat sumber daya IAM.
8. Pilih Buat tumpukan.

Tumpukan membutuhkan waktu beberapa menit untuk membuatnya. Jika tumpukan gagal dibuat, akun Anda mungkin memiliki izin yang tidak mencukupi, atau Anda mungkin telah memasukkan nama bucket yang sudah ada. Gunakan langkah-langkah berikut untuk menghapus tumpukan dan coba lagi:

- a. Pilih Hapus di sudut kanan atas.

Tumpukan membutuhkan beberapa menit untuk dihapus.

 Note

AWS CloudFormation tidak menghapus bucket S3 atau grup CloudWatch log. Anda dapat menghapus sumber daya ini di konsol untuk layanan tersebut.

- b. Jika tumpukan gagal dihapus, pilih Hapus lagi.

- c. Jika tumpukan gagal dihapus lagi, ikuti langkah-langkah di AWS CloudFormation konsol untuk melewati sumber daya yang gagal dihapus, dan coba lagi.
9. Setelah AWS CloudFormation tumpukan berhasil dibuat, ikuti prosedur selanjutnya untuk menjelajahi data properti aset Anda di Amazon S3.

Important

Setelah Anda membuat tumpukan, Anda dapat melihat sumber daya baru di AWS akun Anda. Fitur ini mungkin berhenti bekerja dengan benar jika Anda menghapus atau memodifikasi sumber daya ini. Kami menyarankan agar Anda tidak mengubah sumber daya ini kecuali Anda ingin berhenti mengirim data ke bucket atau ingin menyesuaikan fitur ini.

Lihat data Anda di Amazon S3

Setelah membuat fitur, Anda dapat melihat data properti aset dan metadata aset di Amazon S3.

Note

Metadata aset diperbarui setiap enam jam. Anda mungkin perlu menunggu hingga enam jam untuk melihat metadata aset muncul di bucket S3.

Fitur ini menyimpan data properti aset di kolom berikut, di mana setiap baris berisi titik data:

- `type` — Jenis notifikasi properti (`PropertyValueUpdate`).
- `asset_id` — ID aset yang menerima titik data.
- `asset_property_id` — ID dari properti yang menerima titik data untuk aset.
- `time_in_seconds` — Waktu di mana data diterima, dinyatakan dalam detik dalam waktu epoch Unix.
- `offset_in_nanos` — Offset nanodetik dari `timeInSeconds`
- `asset_property_quality` — Kualitas titik data:,, atau. GOOD UNCERTAIN BAD
- `asset_property_value` — Nilai titik data.
- `asset_property_data_type` — Tipe data dari properti aset:,, atau. `boolean double integer string`

Fitur ini menyimpan metadata aset di kolom berikut, di mana setiap baris berisi properti aset:

- `asset_id` — ID aset.
- `asset_name` — Nama aset.
- `asset_model_id` — ID model aset.
- `asset_property_id` — ID dari properti aset.
- `asset_property_name` — Nama properti aset.
- `asset_property_data_type` — Tipe data dari properti aset:,,, atau. BOOLEAN DOUBLE INTEGER STRING
- `asset_property_unit` — Unit properti aset.
- `asset_property_alias` — Alias properti aset.

Untuk melihat AWS IoT SiteWise data Anda di Amazon S3

1. Arahkan ke [konsol Amazon S3](#).
2. Dari daftar bucket, pilih bucket dengan nama yang Anda pilih saat membuat template.
3. Di ember, pilih salah satu folder berikut:
 - `asset-property-updates`— Folder ini berisi data properti aset yang diekspor dari AWS IoT SiteWise.
 - `asset-metadata`— Folder ini berisi detail aset yang diekspor dari AWS IoT SiteWise.
4. Pilih objek yang ingin Anda lihat.
5. Pada halaman objek, lakukan hal berikut:
 - a. Pilih tab Select from.

Di panel ini, Anda dapat melihat pratinjau catatan dari file Parquet.
 - b. Untuk format File, pilih Parquet.
 - c. Untuk menampilkan isi file dalam format JSON, pilih Tampilkan pratinjau file.

Note

Jika data baru tidak muncul di bucket, periksa apakah Anda mengaktifkan notifikasi pembaruan nilai properti untuk properti aset Anda. Untuk informasi selengkapnya, lihat [Berinteraksi dengan layanan lain AWS](#).

Untuk informasi selengkapnya tentang cara menganalisis data aset yang disimpan di bucket S3, lihat [Analisis data yang diekspor dengan Amazon Athena](#).

Analisis data yang diekspor dengan Amazon Athena

Setelah memiliki data properti aset di Amazon S3, Anda dapat menggunakan beberapa AWS layanan untuk menghasilkan laporan atau menganalisis dan menanyakan data Anda:

- Jalankan kueri SQL pada data Anda menggunakan Amazon [Athena](#).
- Lakukan analisis data besar menggunakan [Amazon EMR](#).
- Cari dan analisis data Anda menggunakan [Amazon OpenSearch Service](#).

Anda dapat menemukan AWS layanan lain yang dapat berinteraksi dengan data Anda di Amazon S3 yang tercantum di bawah Analytics di [AWS Management Console](#)

Note

Tumpukan membuat AWS Glue database untuk memformat data properti aset. Anda tidak dapat menanyakan database ini untuk data aset. Ikuti langkah-langkah di bagian ini untuk membuat AWS Glue database yang dapat Anda kueri.

Dalam tutorial ini, Anda mempelajari cara mengonfigurasi prasyarat untuk menggunakan Amazon Athena dan cara menggunakan Athena untuk menjalankan kueri SQL pada data aset yang diekspor. AWS IoT SiteWise Untuk menanyakan data dengan Athena, Anda harus terlebih dahulu mengisinya AWS Glue Data Catalog dengan data aset Anda. Katalog Data berisi database dan tabel, dan Athena dapat mengakses data dalam Katalog Data. Anda dapat membuat AWS Glue crawler yang secara teratur memperbarui Katalog Data dengan data aset yang diekspor.

Topik

- [Mengkonfigurasi crawler untuk mengisi AWS Glue Data Catalog](#)
- [Meminta data dengan Athena](#)

Mengkonfigurasi crawler untuk mengisi AWS Glue Data Catalog

AWS Glue crawler merayapi penyimpanan data untuk mengisi tabel di AWS Glue Data Catalog. Dalam prosedur ini, Anda membuat dan menjalankan AWS Glue crawler untuk bucket S3 yang berisi data aset yang diekspor. Crawler membuat tabel untuk pembaruan properti aset dan tabel untuk metadata aset. Kemudian, Anda dapat melakukan query SQL pada tabel ini dengan Athena. Untuk informasi selengkapnya, lihat [Mengisi AWS Glue Data Catalog](#) dan [Mendefinisikan crawler di Panduan Pengembang](#).AWS Glue

Untuk membuat AWS Glue crawler

1. Navigasikan ke [konsol AWS Glue](#) tersebut.
2. Di panel navigasi, pilih Perayap.
3. Pilih Tambahkan crawler.
4. Pada halaman Add crawler, lakukan hal berikut:
 - a. Masukkan nama untuk crawler Anda, seperti **IoTSiteWiseDataCrawler**, lalu pilih Berikutnya.
 - b. Untuk tipe sumber Crawler, pilih Penyimpanan data, lalu pilih Berikutnya.
 - c. Pada halaman Tambahkan penyimpanan data, lakukan hal berikut:
 - i. Untuk Pilih penyimpanan data, pilih S3.
 - ii. Di jalur Sertakan, masukkan **s3://DOC-EXAMPLE-BUCKET1** untuk menambahkan bucket data aset Anda sebagai penyimpanan data. Ganti **DOC-EXAMPLE-BUCKET1** dengan nama bucket yang Anda pilih saat membuat tumpukan.
 - iii. Pilih Berikutnya.

Add a data store

Choose a data store

S3

Connection

Select a connection

Optionally include a Network connection to use with this S3 target. Note that each crawler is limited to one Network connection so any future S3 targets will also use the same connection (or none, if left blank).

Add connection

Crawl data in

Specified path in my account
 Specified path in another account

Include path

s3://AWSDOC-EXAMPLE-BUCKET1

All folders and files contained in the include path are crawled. For example, type s3://MyBucket/MyFolder/ to crawl all objects in MyFolder within MyBucket.

▶ Exclude patterns (optional)

Back Next

- d. Pada halaman Tambahkan penyimpanan data lain, pilih Tidak, lalu pilih Berikutnya.
- e. Pada halaman Pilih peran IAM, lakukan hal berikut:
 - i. Untuk membuat peran layanan baru yang memungkinkan AWS Glue untuk mengakses bucket S3, pilih Buat peran IAM.
 - ii. Masukkan akhiran untuk nama peran Anda, seperti **IoTSiteWiseDataCrawler**.
 - iii. Pilih Berikutnya.
- f. Untuk Frekuensi, pilih Per Jam, lalu pilih Berikutnya. Crawler memperbarui tabel dengan data baru setiap kali dijalankan, sehingga Anda dapat memilih frekuensi apa pun yang sesuai dengan kasus penggunaan Anda.
- g. Pada halaman Configure the crawler output, lakukan hal berikut:
 - i. Pilih Tambahkan database untuk membuat AWS Glue database untuk data aset Anda.
 - ii. Masukkan nama untuk database, seperti **iot_sitewise_asset_database**.
 - iii. Pilih Buat.
 - iv. Pilih Berikutnya.
- h. Tinjau detail crawler, lalu pilih Selesai.

Crawler info

Name IoTSiteWiseDataCrawler
Tags -

Data stores

Data store S3
Include path s3://AWSDOC-EXAMPLE-BUCKET1
Connection
Exclude patterns

IAM role

IAM role arn:aws:iam::123456789012:role/service-role/AWSGlueServiceRole-IoTSiteWiseDataCrawler

Schedule

Schedule At 00 minutes past the hour

Output

Database iot_sitewise_asset_database
Prefix added to tables (optional)
Create a single schema for each S3 path false
▶ Configuration options

Secara default, crawler baru Anda tidak segera berjalan. Anda harus menjalankannya secara manual atau menunggu hingga berjalan pada jadwal yang dikonfigurasi.

Untuk menjalankan crawler

1. Pada halaman Crawler, pilih kotak centang untuk crawler baru Anda, lalu pilih Jalankan crawler.

AWS Glue

Data catalog

Databases

Tables

Connections

Crawlers

Classifiers

Settings

Crawlers

A crawler connects to a data store, progresses through a prioritized list of classifiers to determine the schema for your data, and then creates metadata tables in your data catalog. [User preferences](#)

Showing: 1 - 1 < > ↻ ⓘ

<input checked="" type="checkbox"/>	Name	Schedule	Status	Logs	Last runtime	Median runtime	Tables updated	Tables added
<input checked="" style="border: 2px solid red;" type="checkbox"/>	IoTSiteWiseDataCrawler	At 00 minutes...	Ready		0 secs	0 secs	0	0

2. Tunggu hingga crawler selesai dan memiliki status Siap.

Crawler dapat memakan waktu beberapa menit untuk dijalankan, dan statusnya diperbarui secara otomatis.

3. Di panel navigasi, pilih Tabel.

Anda akan melihat dua tabel baru: `asset_metadatan` dan `asset_property_updates`.

Meminta data dengan Athena

Athena secara otomatis menemukan tabel data aset Anda di AWS Glue Data Catalog Untuk melakukan kueri di persimpangan tabel ini, Anda dapat membuat tampilan, yang merupakan tabel data logis. Untuk informasi selengkapnya, lihat [Bekerja dengan tampilan](#) di Panduan Pengguna Amazon Athena.

Setelah membuat tampilan yang menggabungkan data properti aset dan metadata, Anda dapat menjalankan kueri yang menampilkan nilai properti dengan nama aset dan properti yang dilampirkan. Untuk informasi selengkapnya, lihat [Menjalankan kueri SQL menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

Untuk menanyakan data aset dengan Athena

1. Arahkan ke konsol [Athena](#).

Jika halaman Memulai muncul, pilih Memulai.

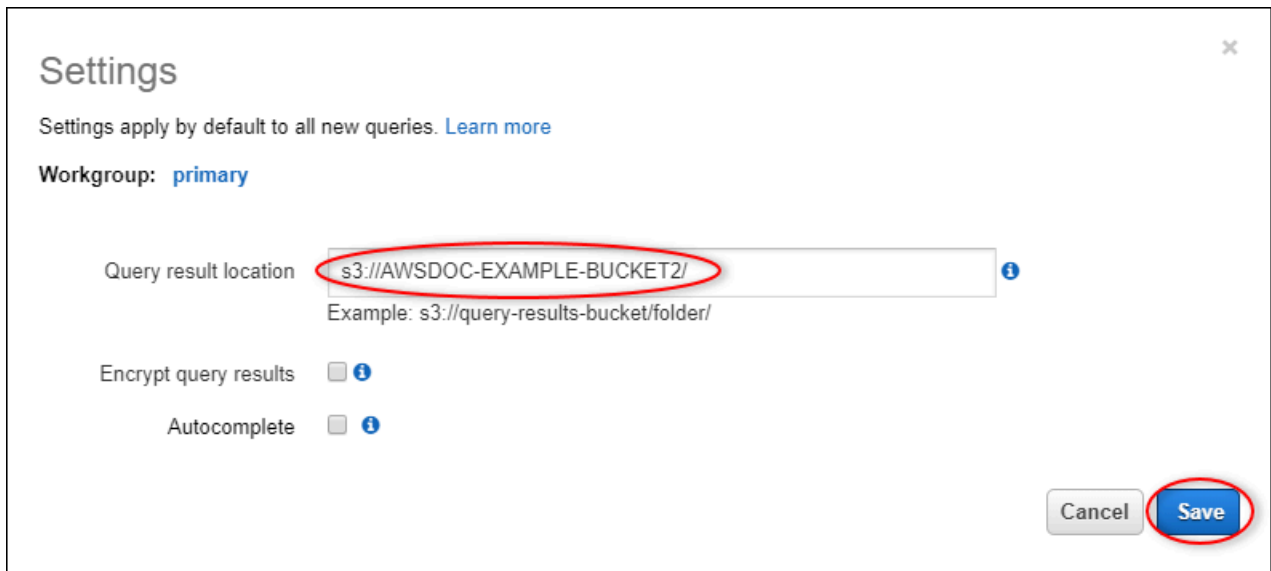
2. Jika Anda menggunakan Athena untuk pertama kalinya, selesaikan langkah-langkah berikut untuk mengonfigurasi bucket S3 untuk hasil kueri. Athena menyimpan hasil pertanyaan Anda di ember ini.

Important

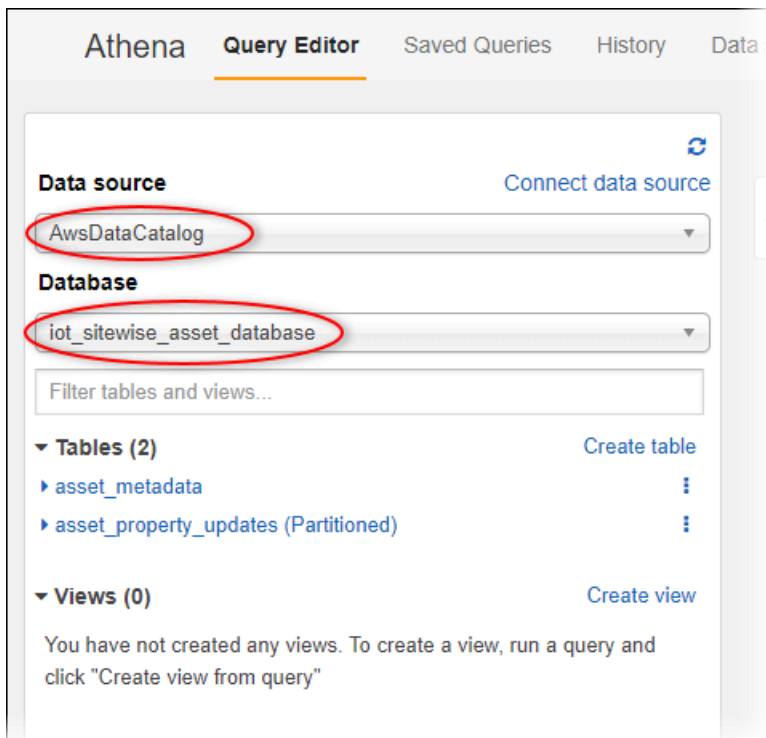
Gunakan bucket yang berbeda dari bucket data aset Anda, sehingga crawler yang Anda buat sebelumnya tidak meng-crawl hasil kueri. Kami menyarankan Anda membuat bucket untuk digunakan hanya untuk hasil kueri Athena. Untuk informasi selengkapnya, lihat [Bagaimana cara membuat bucket S3?](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

- a. Pilih Pengaturan.

- b. Di lokasi hasil Kueri, masukkan bucket S3 untuk hasil kueri Athena. Ember harus diakhiri dengan/.



- c. Pilih Simpan.
3. Panel kiri berisi sumber data untuk query. Lakukan hal-hal berikut:
 - a. Untuk sumber Data, pilih AwsDataCatalog untuk menggunakan file AWS Glue Data Catalog.
 - b. Untuk Database, pilih AWS Glue database yang Anda buat dengan crawler.



Anda akan melihat dua tabel: `asset_metadatan` dan `asset_property_updates`.

4. Untuk membuat tampilan dari kombinasi data properti aset dan metadata, masukkan kueri berikut, lalu pilih Jalankan kueri.

```
CREATE
    OR REPLACE VIEW iot_sitewise_asset_data AS
SELECT "from_unixtime"("time_in_seconds" + ("offset_in_nanos" / 1000000000))
    "timestamp",
    "metadata"."asset_name",
    "metadata"."asset_property_name",
    "data"."asset_property_value",
    "metadata"."asset_property_unit",
    "metadata"."asset_property_alias"
FROM ( "iot_sitewise_asset_database".asset_property_updates data
INNER JOIN "iot_sitewise_asset_database".asset_metadata metadata
    ON ( ("data"."asset_id" = "metadata"."asset_id")
        AND ("data"."asset_property_id" = "metadata"."asset_property_id") ) );
```

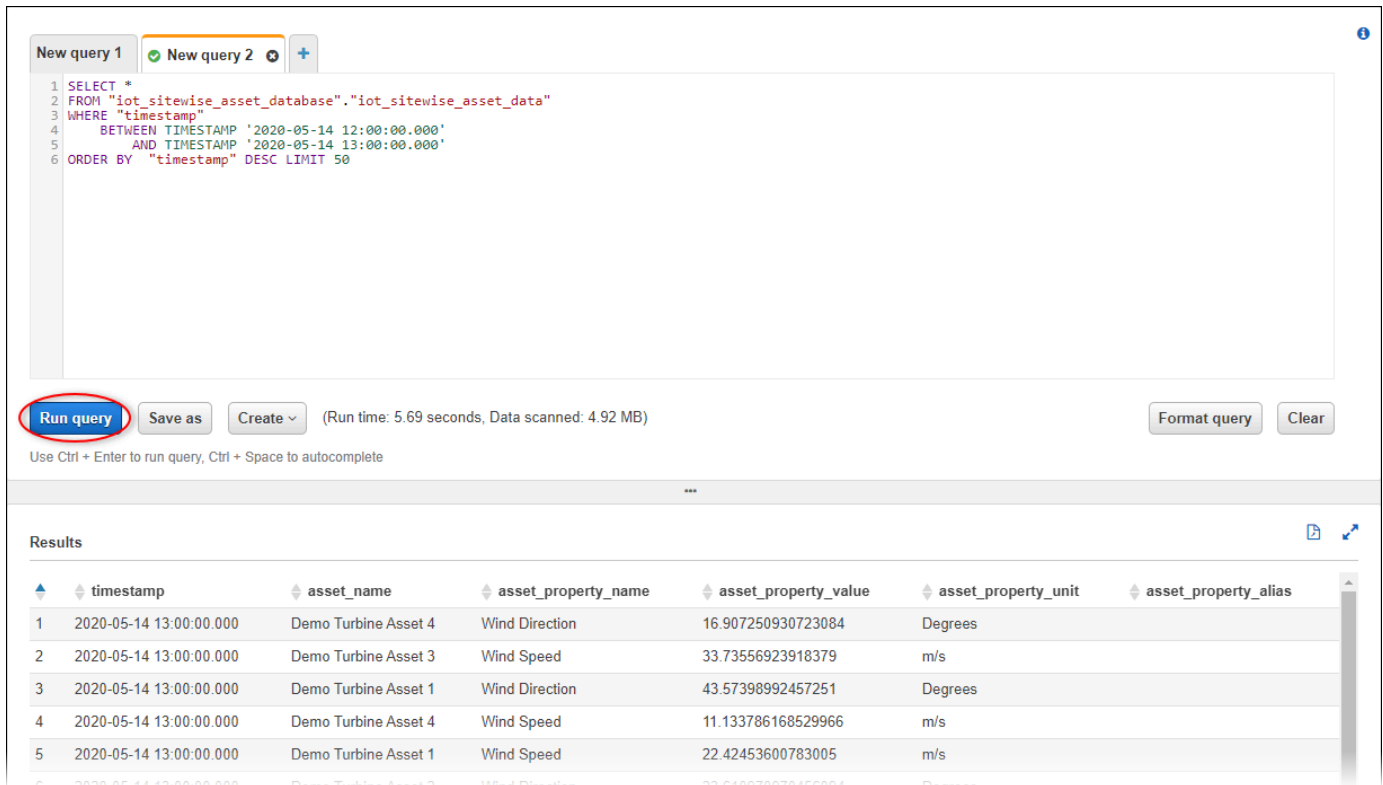
Kueri ini menggabungkan data properti aset dan tabel metadata pada ID aset dan ID properti untuk membuat tampilan. Anda dapat menjalankan kueri ini beberapa kali karena menggantikan tampilan yang ada jika tampilan sudah ada.

5. Untuk menambahkan kueri baru, pilih ikon +.
6. Untuk melihat sampel data aset, masukkan kueri berikut, lalu pilih Jalankan kueri. Ganti stempel waktu dengan interval di mana bucket Anda memiliki data.

```
SELECT *
FROM "iot_sitewise_asset_database"."iot_sitewise_asset_data"
WHERE "timestamp"
    BETWEEN TIMESTAMP '2020-05-14 12:00:00.000'
        AND TIMESTAMP '2020-05-14 13:00:00.000'
ORDER BY "timestamp" DESC LIMIT 50;
```

Kueri ini menghasilkan hingga 50 titik data antara dua stempel waktu, dengan entri terbaru ditampilkan terlebih dahulu.

Output kueri Anda mungkin terlihat mirip dengan hasil berikut.



The screenshot shows the AWS IoT SiteWise console interface. At the top, there are two tabs for 'New query 1' and 'New query 2'. The active query editor contains the following SQL code:

```

1 SELECT *
2 FROM "iot_sitewise_asset_database"."iot_sitewise_asset_data"
3 WHERE "timestamp"
4     BETWEEN TIMESTAMP '2020-05-14 12:00:00.000'
5     AND TIMESTAMP '2020-05-14 13:00:00.000'
6 ORDER BY "timestamp" DESC LIMIT 50

```

Below the query editor, there are buttons for 'Run query' (highlighted with a red circle), 'Save as', 'Create', 'Format query', and 'Clear'. A status bar indicates '(Run time: 5.69 seconds, Data scanned: 4.92 MB)'. Below the query editor, there is a 'Results' section with a table of data:

	timestamp	asset_name	asset_property_name	asset_property_value	asset_property_unit	asset_property_alias
1	2020-05-14 13:00:00.000	Demo Turbine Asset 4	Wind Direction	16.907250930723084	Degrees	
2	2020-05-14 13:00:00.000	Demo Turbine Asset 3	Wind Speed	33.73556923918379	m/s	
3	2020-05-14 13:00:00.000	Demo Turbine Asset 1	Wind Direction	43.57398992457251	Degrees	
4	2020-05-14 13:00:00.000	Demo Turbine Asset 4	Wind Speed	11.133786168529966	m/s	
5	2020-05-14 13:00:00.000	Demo Turbine Asset 1	Wind Speed	22.42453600783005	m/s	

Anda sekarang dapat menjalankan kueri yang berguna untuk AWS IoT SiteWise aplikasi Anda. Untuk informasi selengkapnya, lihat [referensi SQL untuk Amazon Athena](#) di Panduan Pengguna Amazon Athena.

Sumber daya yang dibuat dari template

Saat Anda membuat tumpukan dari template, AWS CloudFormation buat sumber daya berikut. Sebagian besar nama sumber daya menyertakan awalan yang dapat Anda sesuaikan saat membuat tumpukan.

Parameter nama sumber daya

- **BucketName**— Nama bucket S3 yang dibuat dari template ini yang menerima data aset.
- **GlobalResourcePrefix**- Sebuah awalan untuk nama-nama sumber daya global yang dibuat dari template ini. Default ke `sitewise-export-to-s3`.
- **LocalResourcePrefix**— Awalan untuk nama sumber daya yang dibuat dari template ini di Wilayah saat ini. Default ke `sitewise_export_to_s3`.

Sumber daya yang dibuat oleh AWS CloudFormation template

Sumber Daya	Deskripsi	Nama
Bucket S3 untuk data yang diproses	Bucket ini berisi dua folder. Satu folder menerima data yang diratakan dan diformat dari aliran pengiriman Firehose, dan folder lainnya menerima metadata aset.	<code>\${BucketName}</code>
AWS Glue basis data	Database ini berisi AWS Glue tabel yang dibuat oleh tumpukan ini.	<code>\${LocalResourcePrefix}_firehose_glue_database</code>
Tabel AWS Glue	Aliran pengiriman Firehose menggunakan tabel ini untuk memformat data ke format Parquet.	<code>\${LocalResourcePrefix}_firehose_glue_table</code>
AWS Lambda Fungsi yang mengubah data	Fungsi ini meratakan array nilai dalam pesan pemberitahuan nilai properti yang dikirim dari AWS IoT SiteWise	<code>\${LocalResourcePrefix}_lambda_transform_function</code>
Peran IAM untuk fungsi transformasi Lambda	Peran ini memungkinkan Lambda menyimpan log runtime untuk fungsi transformasi.	<code>\${GlobalResourcePrefix}-lambda-transform-role</code>
Kebijakan IAM untuk mengubah peran fungsi Lambda	Kebijakan ini memungkinkan Lambda menyimpan log eksekusi untuk fungsi transformasi.	<code>\${GlobalResourcePrefix}-lambda-transform-policy</code>
CloudWatch Grup log log untuk fungsi transformasi	Grup log ini berisi log untuk fungsi transformasi.	<code>/aws/lambda/\${LocalResourcePrefix}_lambda_transform_function</code>

Sumber Daya	Deskripsi	Nama
Fungsi Lambda yang mengumpulkan metadata aset	Fungsi ini mengambil detail tentang aset AWS IoT SiteWise dan menyimpan detail dalam bucket Amazon S3 yang dibuat oleh tumpukan ini.	<code>\${LocalResourcePrefix}_lambda_metadata_function</code>
Lapisan Lambda untuk fungsi metadata	Lapisan ini menyediakan AWS SDK yang berisi AWS IoT SiteWise operasi yang digunakan fungsi metadata.	<code>\${LocalResourcePrefix}_lambda_metadata_layer</code>
Peran IAM untuk fungsi Lambda metadata	Peran ini memungkinkan Lambda untuk mengambil detail tentang aset di. AWS IoT SiteWise	<code>\${GlobalResourcePrefix}-lambda-metadata-role</code>
Kebijakan IAM untuk peran fungsi Lambda metadata	Kebijakan ini memungkinkan Lambda untuk mengambil rincian tentang aset di. AWS IoT SiteWise	<code>\${GlobalResourcePrefix}-lambda-metadata-policy</code>
EventBridge acara terjadwal untuk fungsi metadata Lambda	Acara terjadwal ini menjalankan metadata Lambda setiap 6 jam untuk memperbarui bucket metadata aset.	<code>\${LocalResourcePrefix}-metadata-event</code>
CloudWatch Grup log log untuk fungsi metadata	Grup log ini berisi log untuk fungsi metadata.	<code>/aws/lambda/\${LocalResourcePrefix}_lambda_metadata_function</code>

Sumber Daya	Deskripsi	Nama
AWS IoT aturan	Aturan ini menanyakan pesan notifikasi nilai properti dan mengirimkan data aset ke aliran pengiriman Amazon Data Firehose.	<code>\${LocalResourcePrefix}_iot_topic_rule</code>
Peran IAM untuk aturan tersebut AWS IoT	Peran ini memungkinkan AWS IoT untuk mengirim data ke aliran pengiriman Firehose.	<code>\${GlobalResourcePrefix}-core-firehose-role</code>
Kebijakan IAM untuk peran AWS IoT aturan	Kebijakan ini memungkinkan AWS IoT untuk mengirim data ke aliran pengiriman Firehose.	<code>\${GlobalResourcePrefix}-core-firehose-policy</code>
Aliran pengiriman Firehose	Aliran pengiriman ini mengkonsumsi data dari AWS IoT aturan, meratakan data dengan fungsi Lambda, dan mengirimkan data ke Amazon S3.	<code>\${LocalResourcePrefix}_firehose_delivery_stream</code>
Peran IAM untuk aliran pengiriman	Peran ini memungkinkan Firehose untuk melakukan operasi pada bucket S3, tabel, fungsi Lambda AWS Glue , dan grup log Log. CloudWatch	<code>\${GlobalResourcePrefix}-firehose-delivery-role</code>
CloudWatch Grup log log untuk aliran pengiriman	Grup log ini berisi aliran logS3 Delivery, yang menerima log tentang aliran pengiriman Firehose.	<code>/aws/kinesisfirehose/\${LocalResourcePrefix}_firehose_delivery_stream</code>

Integrasi dengan Grafana

Grafana adalah platform visualisasi data yang dapat Anda gunakan untuk memvisualisasikan dan memantau data di dasbor. Di Grafana versi 7.3.0 dan yang lebih baru, Anda dapat menggunakan AWS IoT SiteWise plugin untuk memvisualisasikan AWS IoT SiteWise data aset Anda di dasbor Grafana. Anda dapat memvisualisasikan data dari berbagai AWS sumber (seperti AWS IoT SiteWise, Amazon Timestream, dan CloudWatch Amazon) dan sumber data lainnya dengan satu dasbor Grafana.

Anda memiliki dua opsi untuk menggunakan AWS IoT SiteWise plugin:

- Server Grafana lokal

Anda dapat mengatur AWS IoT SiteWise plugin di server Grafana yang Anda kelola. Untuk informasi lebih lanjut tentang cara menambahkan dan menggunakan plugin, lihat file [AWS IoT SiteWise Datasource README](#) di situs web. GitHub

- AWS Managed Service for Grafana

Anda dapat menggunakan AWS IoT SiteWise plugin di AWS Managed Service for Grafana (AMG). AMG mengelola server Grafana untuk Anda sehingga Anda dapat memvisualisasikan data Anda tanpa harus membangun, mengemas, atau menyebarkan perangkat keras apa pun atau infrastruktur Grafana lainnya. Untuk informasi selengkapnya, lihat topik berikut di AWS Managed Service for Grafana User Guide:

- [Apa itu Amazon Managed Service for Grafana \(AMG\)?](#)
- [Menggunakan sumber AWS IoT SiteWise data](#)

Example Contoh dasbor Grafana

Dasbor Grafana berikut memvisualisasikan [demo](#) ladang angin. Anda dapat mengakses dasbor demo ini di situs web [Grafana Play](#).



Mengintegrasikan dan AWS IoT SiteWiseAWS IoT TwinMaker

Mengintegrasikan dengan AWS IoT TwinMaker memberikan akses ke fungsionalitas yang kuat di AWS IoT SiteWise, seperti ExecuteQuery API pengambilan AWS IoT SiteWise data dan pencarian aset lanjutan di konsol. AWS IoT SiteWise Untuk mengintegrasikan layanan dan menggunakan fitur-fitur ini, Anda harus terlebih dahulu mengaktifkan integrasi.

Topik

- [Mengaktifkan integrasi](#)
- [Mengintegrasikan dan AWS IoT SiteWiseAWS IoT TwinMaker](#)

Mengaktifkan integrasi

Administrator dapat menggunakan kebijakan JSON AWS untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa. Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Untuk informasi selengkapnya tentang tindakan yang AWS IoT SiteWise didukung, lihat [Tindakan yang ditentukan oleh AWS IoT SiteWise](#) dalam Referensi Otorisasi Layanan.

Untuk informasi selengkapnya tentang peran AWS IoT TwinMaker terkait layanan, lihat Peran [terkait layanan AWS IoT TwinMaker di Panduan Pengguna](#). AWS IoT TwinMaker

Sebelum Anda dapat mengintegrasikan AWS IoT SiteWise dan AWS IoT TwinMaker, Anda harus memberikan izin berikut yang memungkinkan AWS IoT SiteWise untuk mengintegrasikan dengan ruang kerja AWS IoT TwinMaker tertaut:

- `iotsitewise:EnableSiteWiseIntegration`— Memungkinkan AWS IoT SiteWise untuk berintegrasi dengan AWS IoT TwinMaker ruang kerja yang ditautkan. Integrasi ini memungkinkan AWS IoT TwinMaker untuk membaca semua informasi pemodelan Anda AWS IoT SiteWise melalui peran AWS IoT TwinMaker terkait layanan. Untuk mengaktifkan izin ini, tambahkan kebijakan berikut ke peran IAM Anda:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:EnableSiteWiseIntegration"
      ],
      "Resource": "*"
    }
  ]
}
```

Mengintegrasikan dan AWS IoT SiteWise AWS IoT TwinMaker

Untuk mengintegrasikan AWS IoT SiteWise dan AWS IoT TwinMaker, Anda harus memiliki yang berikut:

- AWS IoT SiteWise pengaturan peran terkait layanan di akun Anda
- AWS IoT TwinMaker pengaturan peran terkait layanan di akun Anda
- AWS IoT TwinMaker ruang kerja dengan ID `IoTSiteWiseDefaultWorkspace` di akun Anda di Wilayah.

Untuk mengintegrasikan dengan menggunakan AWS IoT SiteWise konsol

Saat Anda melihat AWS IoT TwinMaker spanduk Integrasi dengan di konsol, pilih Berikan izin. Prasyarat dibuat di akun Anda.

Untuk mengintegrasikan dengan menggunakan AWS CLI

Untuk mengintegrasikan AWS IoT SiteWise dan AWS IoT TwinMaker dengan menggunakan AWS CLI, masukkan perintah berikut:

1. Panggil `CreateServiceLinkedRole` dengan sebuah `AWSServiceName` dari `iotsitewise.amazonaws.com`.

```
aws iam create-service-linked-role --aws-service-name iotsitewise.amazonaws.com
```

2. Panggil `CreateServiceLinkedRole` dengan sebuah `AWSServiceName` dari `iottwinmaker.amazonaws.com`.

```
aws iam create-service-linked-role --aws-service-name iottwinmaker.amazonaws.com
```

3. Panggil `CreateWorkspace` dengan sebuah ID dari `IoTSiteWiseDefaultWorkspace`.

```
aws iottwinmaker create-workspace --workspace-id IoTSiteWiseDefaultWorkspace
```

Mendeteksi anomali peralatan dengan Amazon Lookout for Equipment

Note

Deteksi anomali hanya tersedia di Wilayah di mana Amazon Lookout for Equipment tersedia.

Anda dapat berintegrasi AWS IoT SiteWise dengan Amazon Lookout for Equipment untuk mendapatkan wawasan tentang peralatan industri Anda melalui deteksi anomali dan pemeliharaan prediktif peralatan industri. Lookout for Equipment adalah layanan machine learning (ML) untuk memantau peralatan industri yang mendeteksi perilaku peralatan abnormal dan mengidentifikasi potensi kegagalan. Dengan Lookout for Equipment, Anda dapat menerapkan program pemeliharaan prediktif dan mengidentifikasi proses peralatan yang kurang optimal. Untuk informasi selengkapnya tentang Lookout for Equipment, lihat [Apa itu Amazon Lookout for Equipment?](#) di Panduan Pengguna Amazon Lookout for Equipment.

Saat Anda membuat prediksi untuk melatih model ML guna mendeteksi perilaku peralatan anomali, AWS IoT SiteWise kirimkan nilai properti aset ke Lookout for Equipment untuk melatih model ML guna mendeteksi perilaku peralatan anomali. Untuk menentukan definisi prediksi pada model aset, Anda menentukan peran IAM yang diperlukan untuk Lookout for Equipment untuk mengakses data dan properti yang akan dikirim ke Lookout for Equipment dan mengirim data yang diproses ke Amazon S3. Untuk informasi selengkapnya, lihat [Membuat model aset](#).

Untuk mengintegrasikan AWS IoT SiteWise dan Lookout for Equipment, Anda akan melakukan langkah-langkah tingkat tinggi berikut:

- Tambahkan definisi prediksi pada model aset yang menguraikan properti apa yang ingin Anda lacak. Definisi prediksi adalah kumpulan pengukuran, transformasi, dan metrik yang dapat digunakan kembali yang digunakan untuk membuat prediksi pada aset yang didasarkan pada model aset tersebut.
- Latih prediksi berdasarkan data historis yang Anda berikan.
- Jadwal inferensi, yang menceritakan AWS IoT SiteWise seberapa sering menjalankan prediksi tertentu.

Setelah inferensi dijadwalkan, model Lookout for Equipment memantau data yang diterimanya dari peralatan Anda dan mencari anomali dalam perilaku peralatan. Anda dapat melihat dan menganalisis hasil di SiteWise Monitor, menggunakan operasi AWS IoT SiteWise GET API, atau konsol Lookout for Equipment. Anda juga dapat membuat alarm menggunakan detektor alarm dari model aset untuk mengingatkan Anda tentang perilaku peralatan yang tidak normal.

Topik

- [Menambahkan definisi prediksi \(konsol\)](#)
- [Melatih prediksi \(konsol\)](#)
- [Memulai atau menghentikan inferensi pada prediksi \(konsol\)](#)

- [Menambahkan definisi prediksi \(CLI\)](#)
- [Melatih prediksi dan inferensi awal \(CLI\)](#)
- [Pelatihan prediksi \(CLI\)](#)
- [Memulai atau menghentikan inferensi pada prediksi \(CLI\)](#)

Menambahkan definisi prediksi (konsol)

Untuk mulai mengirim data yang dikumpulkan oleh AWS IoT SiteWise Lookout for Equipment, Anda harus menambahkan definisi prediksi ke model AWS IoT SiteWise aset.

Untuk menambahkan definisi prediksi ke model AWS IoT SiteWise aset

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Model dan pilih model aset yang ingin Anda tambahkan definisi prediksi.
3. Pilih Prediksi.
4. Pilih Tambahkan definisi prediksi.
5. Tentukan detail tentang definisi prediksi.
 - a. Masukkan Nama unik dan Deskripsi untuk definisi prediksi Anda. Pilih nama dengan serius karena setelah Anda membuat definisi prediksi, Anda tidak dapat mengubah namanya.
 - b. Buat atau pilih peran izin IAM yang memungkinkan Anda membagikan data aset Anda AWS IoT SiteWise dengan Amazon Lookout for Equipment. Peran tersebut harus memiliki IAM dan kebijakan kepercayaan berikut. Untuk bantuan membuat peran, lihat [Membuat peran menggunakan kebijakan kepercayaan khusus \(konsol\)](#).

Kebijakan IAM

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "L4EPermissions",
      "Effect": "Allow",
      "Action": [
        "lookoutequipment:CreateDataset",
        "lookoutequipment:CreateModel",
        "lookoutequipment:CreateInferenceScheduler",
        "lookoutequipment:DescribeDataset",
```

```

        "lookoutequipment:DescribeDataIngestionJob",
        "lookoutequipment:DescribeModel",
        "lookoutequipment:DescribeInferenceScheduler",
        "lookoutequipment:ListInferenceExecutions",
        "lookoutequipment:StartDataIngestionJob",
        "lookoutequipment:StartInferenceScheduler",
        "lookoutequipment:UpdateInferenceScheduler",
        "lookoutequipment:StopInferenceScheduler"
    ],
    "Resource": "*"
},
{
    "Sid": "S3Permissions",
    "Effect": "Allow",
    "Action": [
        "s3:CreateBucket",
        "s3:ListBucket",
        "s3:PutObject",
        "s3:GetObject"
    ],
    "Resource": ["arn:aws:s3:::iotsitewise-*"]
},
{
    "Sid": "IAMPermissions",
    "Effect": "Allow",
    "Action": [
        "iam:GetRole",
        "iam:PassRole"
    ],
    "Resource": "arn:aws:iam:::role/*"
}
]
}

```

Kebijakan kepercayaan

```

{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Principal": {
            "Service": "iotsitewise.amazonaws.com"
        }
    }],
}

```



```

        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "account_id"
            },
            "ArnEquals": {
                "aws:SourceArn":
"arn:aws:iotsitewise:region:account_id:asset/*"
            }
        }
    },
    {
        "Effect": "Allow",
        "Principal": {
            "Service": "lookoutequipment.amazonaws.com"
        },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "account_id"
            },
            "ArnEquals": {
                "aws:SourceArn":
"arn:aws:lookoutequipment:region:account_id:*"
            }
        }
    }
]
}

```

- c. Pilih Berikutnya.
6. Pilih atribut data (pengukuran, transformasi, dan metrik) yang ingin Anda kirim ke Lookout for Equipment.
 - a. (Opsional) Pilih pengukuran.
 - b. (Opsional) Pilih transformasi.
 - c. (Opsional) Pilih metrik.
 - d. Pilih Berikutnya.
 7. Tinjau pilihan Anda. Untuk menambahkan definisi prediksi ke model aset, pada halaman ringkasan, pilih Tambahkan definisi prediksi.

Anda juga dapat Mengedit atau Menghapus definisi prediksi yang ada yang memiliki prediksi aktif yang dilampirkan.

Melatih prediksi (konsol)

Setelah Anda menambahkan definisi prediksi ke model aset, Anda dapat melatih prediksi yang ada di aset Anda.

Untuk melatih prediksi di AWS IoT SiteWise

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset, dan pilih aset yang ingin dipantau.
3. Pilih Prediksi.
4. Pilih prediksi yang ingin Anda latih.
5. Di bawah Tindakan, pilih Mulai pelatihan, dan lakukan hal berikut:
 - a. Di bawah Detail Prediksi, pilih peran izin IAM yang memungkinkan AWS IoT SiteWise untuk berbagi data aset Anda dengan Lookout for Equipment. Jika Anda perlu membuat peran baru, pilih Buat peran baru.
 - b. Untuk pengaturan data Pelatihan, masukkan rentang waktu data Pelatihan untuk memilih data mana yang akan digunakan untuk melatih prediksi.
 - c. (Opsional) Untuk label Data, berikan bucket dan awalan Amazon S3 yang menyimpan data pelabelan Anda. Untuk informasi selengkapnya tentang pelabelan data, lihat [Melabeli data Anda di Panduan](#) Pengguna Amazon Lookout for Equipment.
 - d. Pilih Berikutnya.
6. (Opsional) Jika Anda ingin prediksi aktif segera setelah menyelesaikan pelatihan, di bawah Pengaturan lanjutan, pilih Aktifkan prediksi secara otomatis setelah pelatihan, lalu lakukan hal berikut:
 - a. Di bawah Input data, untuk frekuensi upload Data, tentukan seberapa sering data diunggah, dan untuk waktu tunda Offset, tentukan berapa banyak buffer yang akan digunakan.
 - b. Pilih Berikutnya.
7. Tinjau detail prediksi dan pilih Simpan dan mulai.

Memulai atau menghentikan inferensi pada prediksi (konsol)

Note

Biaya Lookout for Equipment berlaku untuk kesimpulan terjadwal dengan data yang ditransfer antara dan AWS IoT SiteWise Lookout for Equipment. Untuk informasi selengkapnya, lihat [harga Amazon Lookout](#) for Equipment.

Jika Anda menambahkan prediksi tetapi tidak memilih untuk mengaktifkannya setelah pelatihan, Anda harus mengaktifkannya agar dapat mulai memantau aset Anda.

Untuk memulai inferensi untuk prediksi

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset, dan pilih aset tempat prediksi ditambahkan.
3. Pilih Prediksi.
4. Pilih prediksi yang ingin Anda aktifkan.
5. Di bawah Tindakan, pilih Mulai inferensi, dan lakukan hal berikut:
 - a. Di bawah Input data, untuk frekuensi upload Data, tentukan seberapa sering data diunggah, dan untuk waktu tunda Offset, tentukan berapa banyak buffer yang akan digunakan.
 - b. Pilih Simpan dan mulai.

Untuk menghentikan inferensi untuk prediksi

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Aset, dan pilih aset tempat prediksi ditambahkan.
3. Pilih Prediksi.
4. Pilih prediksi yang ingin Anda hentikan.
5. Di bawah Tindakan, pilih Hentikan inferensi.

Menambahkan definisi prediksi (CLI)

Untuk menentukan definisi prediksi pada model aset baru atau yang sudah ada, Anda dapat menggunakan AWS Command Line Interface (AWS CLI). Setelah Anda menentukan definisi prediksi

pada model aset, Anda melatih, dan menjadwalkan inferensi untuk, prediksi pada aset AWS IoT SiteWise untuk melakukan deteksi anomali dengan Lookout for Equipment.

Prasyarat

Untuk menyelesaikan langkah-langkah ini, Anda harus memiliki model aset dan setidaknya satu aset yang dibuat. Lihat informasi yang lebih lengkap di [Membuat model aset \(AWS CLI\)](#) dan [Membuat aset \(AWS CLI\)](#).

Jika Anda baru AWS IoT SiteWise, Anda harus memanggil operasi `CreateBulkImportJob` API untuk mengimpor nilai properti aset ke dalam AWS IoT SiteWise, yang akan digunakan untuk melatih model. Untuk informasi selengkapnya, lihat [Buat pekerjaan impor massal \(AWS CLI\)](#).

Untuk menambahkan definisi prediksi

1. Buat file bernama `asset-model-payload.json`. Ikuti langkah-langkah di bagian lain ini untuk menambahkan detail model aset Anda ke file, tetapi jangan kirimkan permintaan untuk membuat atau memperbarui model aset.
 - Untuk informasi selengkapnya tentang cara membuat model aset, lihat [Membuat model aset \(AWS CLI\)](#)
 - Untuk informasi selengkapnya tentang cara memperbarui model aset yang ada, lihat [Memperbarui aset atau model komponen \(AWS CLI\)](#)
2. Tambahkan Lookout for Equipment composite model `assetModelCompositeModels()` ke model aset dengan menambahkan kode berikut.
 - Ganti *Property* dengan ID properti yang ingin Anda sertakan. Untuk mendapatkan ID tersebut, hubungi [DescribeAssetModel](#).
 - Ganti *RoleARN* dengan ARN dari peran IAM yang memungkinkan Lookout for Equipment mengakses data Anda. AWS IoT SiteWise

```
{
  ...
  "assetModelCompositeModels": [
    {
      "name": "L4Epredictiondefinition",
      "type": "AWS/L4E_ANOMALY",
      "properties": [
        {
```

```

    "name": "AWS/L4E_ANOMALY_RESULT",
    "dataType": "STRUCT",
    "dataTypeSpec": "AWS/L4E_ANOMALY_RESULT",
    "unit": "none",
    "type": {
      "measurement": {}
    }
  },
  {
    "name": "AWS/L4E_ANOMALY_INPUT",
    "dataType": "STRUCT",
    "dataTypeSpec": "AWS/L4E_ANOMALY_INPUT",
    "type": {
      "attribute": {
        "defaultValue": "{\"properties\": [\"Property1\", \"Property2\"]}"
      }
    }
  },
  {
    "name": "AWS/L4E_ANOMALY_PERMISSIONS",
    "dataType": "STRUCT",
    "dataTypeSpec": "AWS/L4E_ANOMALY_PERMISSIONS",
    "type": {
      "attribute": {
        "defaultValue": "{\"roleArn\": \"RoleARN\"}"
      }
    }
  },
  {
    "name": "AWS/L4E_ANOMALY_DATASET",
    "dataType": "STRUCT",
    "dataTypeSpec": "AWS/L4E_ANOMALY_DATASET",
    "type": {
      "attribute": {}
    }
  },
  {
    "name": "AWS/L4E_ANOMALY_MODEL",
    "dataType": "STRUCT",
    "dataTypeSpec": "AWS/L4E_ANOMALY_MODEL",
    "type": {
      "attribute": {}
    }
  },
},

```

```

    {
      "name": "AWS/L4E_ANOMALY_INFERENCE",
      "dataType": "STRUCT",
      "dataTypeSpec": "AWS/L4E_ANOMALY_INFERENCE",
      "type": {
        "attribute": {}
      }
    },
    {
      "name": "AWS/L4E_ANOMALY_TRAINING_STATUS",
      "dataType": "STRUCT",
      "dataTypeSpec": "AWS/L4E_ANOMALY_TRAINING_STATUS",
      "type": {
        "attribute": {
          "defaultValue": "{}"
        }
      }
    },
    {
      "name": "AWS/L4E_ANOMALY_INFERENCE_STATUS",
      "dataType": "STRUCT",
      "dataTypeSpec": "AWS/L4E_ANOMALY_INFERENCE_STATUS",
      "type": {
        "attribute": {
          "defaultValue": "{}"
        }
      }
    }
  ]
}

```

3. Buat model aset atau perbarui model aset yang ada. Lakukan salah satu hal berikut:

- Untuk membuat model aset, jalankan perintah berikut:

```
aws iotsitewise create-asset-model --cli-input-json file://asset-model-payload.json
```

- Untuk memperbarui model aset yang ada, jalankan perintah berikut. Ganti *asset-model-id* dengan ID model aset yang ingin Anda perbarui.

```
aws iotsitewise update-asset-model \
  --asset-model-id asset-model-id \
```

```
--cli-input-json file://asset-model-payload.json
```

Setelah Anda menjalankan perintah, perhatikan `assetModelId` dalam respon.

Melatih prediksi dan inferensi awal (CLI)

Sekarang setelah definisi prediksi ditentukan, Anda dapat melatih aset berdasarkan itu dan memulai inferensi. Jika Anda ingin melatih prediksi Anda tetapi tidak memulai inferensi, lompat ke [Pelatihan prediksi \(CLI\)](#). Untuk melatih prediksi dan memulai inferensi pada aset, Anda memerlukan sumber `assetId` daya target.

Untuk melatih dan memulai inferensi prediksi

1. Jalankan perintah berikut untuk menemukan bagian `assetModelCompositeModelId` bawah `assetModelCompositeModelSummaries`. Ganti `asset-model-id` dengan ID model aset yang Anda buat [Memperbarui aset atau model komponen \(AWS CLI\)](#).

```
aws iotsitewise describe-asset-model \  
  --asset-model-id asset-model-id \  
  --cli-input-json file://asset-model-payload.json
```

2. Jalankan perintah berikut untuk menemukan `TrainingWithInference` tindakan. `actionDefinitionId` Ganti `asset-model-id` dengan ID yang digunakan pada langkah sebelumnya dan ganti `asset-model-composite-model-id` dengan ID yang dikembalikan pada langkah sebelumnya.

```
aws iotsitewise describe-asset-model-composite-model \  
  --asset-model-id asset-model-id \  
  --asset-model-composite-model-id asset-model-composite-model-id \  
  --cli-input-json file://asset-model-payload.json
```

3. Buat file bernama `train-start-inference-prediction.json` dan tambahkan kode berikut, ganti yang berikut ini:

- `asset-id` dengan ID aset target
- `action-definition-id` dengan ID `TrainingWithInference` tindakan
- `StartTime` dengan dimulainya data pelatihan, disediakan dalam hitungan detik
- `EndTime` dengan berakhirnya data pelatihan, disediakan dalam hitungan detik

```
{
```

```

"targetResource": {
  "assetId": "asset-id"
},
"actionDefinitionId": "action-definition-Id",
"actionPayload":{
  "stringValue": "{\"l4ETrainingWithInference\":{\"trainingWithInferenceMode
\": \"START\", \"trainingPayload\": {\"exportDataStartTime\": StartTime,
\"exportDataEndTime\": EndTime}, \"inferencePayload\": {\"dataDelayOffsetInMinutes
\": 0, \"dataUploadFrequency\": \"PT5M\"}}}"
}
}

```

4. Jalankan perintah berikut untuk memulai pelatihan dan inferensi:

```

aws iotsitewise execute-action --cli-input-json file://train-start-inference-
prediction.json

```

Pelatihan prediksi (CLI)

Sekarang setelah definisi prediksi ditentukan, Anda dapat melatih aset berdasarkan itu. Untuk melatih prediksi aset, Anda memerlukan sumber daya target. `assetId`

Untuk melatih prediksi

1. Jalankan perintah berikut untuk menemukan bagian `assetModelCompositeModelId` bawah `assetModelCompositeModelSummaries`. Ganti *asset-model-id* dengan ID model aset yang Anda buat [Memperbarui aset atau model komponen \(AWS CLI\)](#).

```

aws iotsitewise describe-asset-model \
  --asset-model-id asset-model-id \

```


2. Jalankan perintah berikut untuk menemukan Training tindakan. `actionDefinitionId` Ganti *asset-model-id* dengan ID yang digunakan pada langkah sebelumnya dan ganti *asset-model-composite-model-id* dengan ID yang dikembalikan pada langkah sebelumnya.

```

aws iotsitewise describe-asset-model-composite-model \
  --asset-model-id asset-model-id \
  --asset-model-composite-model-id asset-model-composite-model-id \

```


3. Buat file bernama `train-prediction.json` dan tambahkan kode berikut, ganti yang berikut ini:
 - *asset-id* dengan ID aset target
 - *action-definition-id* dengan ID tindakan pelatihan
 - *StartTime* dengan dimulainya data pelatihan, disediakan dalam hitungan detik
 - *EndTime* dengan berakhirnya data pelatihan, disediakan dalam hitungan detik
 - (Opsional) *BucketName* dengan nama bucket Amazon S3 yang menyimpan data label Anda
 - (Opsional) *Prefix* dengan awalan yang terkait dengan bucket Amazon S3.

 Note

Sertakan nama bucket dan awalan atau keduanya.

```
{
  "targetResource": {
    "assetId": "asset-id"
  },
  "actionDefinitionId": "action-definition-Id",
  "actionPayload": { "stringValue": "{\"l4ETraining\": {\"trainingMode\":
  \\\"START\\\", \\\"exportDataStartTime\\\": StartTime, \\\"exportDataEndTime\\\": EndTime,
  \\\"labelInputConfiguration\\\": {\"bucketName\\\": \\\"BucketName\\\", \\\"prefix\\\":
  \\\"Prefix\\\"}}}"
  }
}
```

4. Jalankan perintah berikut untuk memulai pelatihan:

```
aws iotsitewise execute-action --cli-input-json file://train-prediction.json
```

Sebelum Anda dapat memulai inferensi, pelatihan harus diselesaikan. Untuk memeriksa status pelatihan, lakukan salah satu hal berikut:

- Dari konsol, arahkan ke aset tempat prediksi aktif.
- Dari AWS CLI, hubungi `BatchGetAssetPropertyValue` menggunakan `trainingStatus` properti. `propertyId`

Memulai atau menghentikan inferensi pada prediksi (CLI)

Setelah prediksi dilatih, Anda dapat memulai inferensi untuk memberi tahu Lookout for Equipment untuk mulai memantau aset Anda. Untuk memulai atau menghentikan inferensi, Anda memerlukan `assetId` sumber daya target.

Untuk memulai inferensi

1. Jalankan perintah berikut untuk menemukan bagian `assetModelCompositeModelId` bawah `assetModelCompositeModelSummaries`. Ganti `asset-model-id` dengan ID model aset yang Anda buat [Memperbarui aset atau model komponen \(AWS CLI\)](#).

```
aws iotsitewise describe-asset-model \
  --asset-model-id asset-model-id \
```

2. Jalankan perintah berikut untuk menemukan Inference tindakan. `actionDefinitionId` Ganti `asset-model-id` dengan ID yang digunakan pada langkah sebelumnya dan ganti `asset-model-composite-model-id` dengan ID yang dikembalikan pada langkah sebelumnya.

```
aws iotsitewise describe-asset-model-composite-model \
  --asset-model-id asset-model-id \
  --asset-model-composite-model-id asset-model-composite-model-id \
```

3. Buat file bernama `start-inference.json` dan tambahkan kode berikut, ganti yang berikut ini:

- `asset-id` dengan ID aset target
- `action-definition-id` dengan ID dari tindakan inferensi awal
- `Offset` dengan jumlah buffer yang akan digunakan
- `Frequency` dengan seberapa sering data diunggah

```
{
  "targetResource": {
    "assetId": "asset-id"
  },
  "actionDefinitionId": "action-definition-Id",
  "actionPayload": { "stringValue": "{\"l4EInference\": {\"inferenceMode\": \"START\", \"dataDelayOffsetInMinutes\": Offset, \"dataUploadFrequency\": \"Frequency\"}}"
```



```
aws iotsitewise execute-action --cli-input-json file://stop-inference.json
```

Mengelola penyimpanan data

Anda dapat mengonfigurasi AWS IoT SiteWise untuk menyimpan data Anda di tingkatan penyimpanan berikut:

Tingkat panas

Tingkat penyimpanan panas adalah penyimpanan deret waktu yang AWS IoT SiteWise dikelola. Hot tier paling efektif untuk data yang sering diakses, dengan write-to-read latensi rendah. Data yang disimpan di hot tier digunakan oleh aplikasi industri yang membutuhkan akses cepat ke nilai pengukuran terbaru di peralatan Anda. Ini termasuk aplikasi yang memvisualisasikan metrik real-time dengan dasbor interaktif, atau aplikasi yang memantau operasi dan meluncurkan alarm untuk mengidentifikasi masalah kinerja.

Secara default, data yang tertelan AWS IoT SiteWise disimpan di hot tier. Anda dapat menentukan periode retensi untuk tingkat panas, setelah itu AWS IoT SiteWise memindahkan data di tingkat panas ke penyimpanan tingkat hangat atau dingin, berdasarkan konfigurasi Anda. Untuk kinerja terbaik dan efisiensi biaya, atur periode retensi hot tier Anda menjadi lebih lama dari waktu yang dibutuhkan untuk sering mengambil data. Ini digunakan untuk metrik waktu nyata, alarm, dan skenario pemantauan. Jika periode retensi tidak ditetapkan, data Anda disimpan tanpa batas waktu di tingkat panas.

Tingkat hangat

Tingkat penyimpanan hangat adalah tingkat AWS IoT SiteWise terkelola yang efektif untuk penyimpanan data historis yang hemat biaya. Ini paling baik digunakan untuk mengambil volume besar data dengan karakteristik write-to-read latensi sedang. Gunakan tingkat hangat untuk menyimpan data historis yang diperlukan untuk beban kerja yang besar. Misalnya, ini digunakan untuk pengambilan data untuk analitik, aplikasi intelijen bisnis (BI), alat pelaporan, dan pelatihan model pembelajaran mesin (ML). Jika Anda mengaktifkan tingkat penyimpanan dingin, Anda dapat menentukan periode retensi tingkat hangat. Setelah periode retensi berakhir, AWS IoT SiteWise menghapus data dari tingkat hangat.

Tingkat dingin

Tingkat penyimpanan dingin menggunakan bucket Amazon S3 untuk menyimpan data yang jarang digunakan. Dengan tingkat dingin diaktifkan, AWS IoT SiteWise mereplikasi deret waktu, termasuk pengukuran, metrik, transformasi dan agregat, dan definisi model aset setiap 6 jam. Tingkat dingin digunakan untuk menyimpan data yang mentolerir latensi baca tinggi untuk laporan historis dan cadangan.

Topik

- [Konfigurasi pengaturan penyimpanan](#)
- [Memecahkan masalah pengaturan penyimpanan](#)
- [Jalur file dan skema data yang disimpan di tingkat dingin](#)

Konfigurasi pengaturan penyimpanan

Anda dapat mengonfigurasi pengaturan penyimpanan untuk ikut serta dalam layanan penyimpanan tingkat hangat yang dikelola, dan juga untuk mereplikasi data ke tingkat dingin. Untuk mempelajari lebih lanjut tentang periode retensi untuk tingkat hangat dan panas, lihat [Dampak retensi data](#). Saat mengonfigurasi pengaturan penyimpanan, lakukan hal berikut:

- Retensi tingkat panas — Tetapkan periode retensi untuk berapa lama data Anda disimpan di hot tier sebelum dihapus, dan dipindahkan ke layanan penyimpanan tingkat hangat atau penyimpanan tingkat dingin yang dikelola berdasarkan pengaturan penyimpanan Anda. AWS IoT SiteWise akan menghapus data apa pun di hot tier yang ada sebelum periode retensi berakhir. Jika Anda tidak menetapkan periode retensi, data Anda disimpan tanpa batas waktu di tingkat panas.
- Retensi tingkat hangat — Tetapkan periode retensi untuk berapa lama data Anda disimpan di tingkat hangat sebelum dihapus dari AWS IoT SiteWise penyimpanan dan dipindahkan ke penyimpanan tingkat dingin yang dikelola pelanggan. AWS IoT SiteWise menghapus data apa pun dari tingkat hangat yang ada sebelum periode retensi berakhir. Jika periode retensi tidak diatur, data Anda disimpan tanpa batas waktu di tingkat hangat.

Note

Untuk meningkatkan kinerja kueri, tetapkan periode retensi tingkat panas dengan penyimpanan tingkat hangat.

Dampak retensi data dalam penyimpanan tingkat panas dan hangat

- Saat Anda mengurangi periode retensi penyimpanan tingkat panas, data dipindahkan secara permanen dari tingkat panas ke tingkat hangat atau dingin. Saat Anda mengurangi periode retensi tingkat hangat, data dipindahkan ke tingkat dingin, dan dihapus secara permanen dari tingkat hangat.

- Saat Anda meningkatkan periode retensi penyimpanan tingkat panas atau hangat, perubahan memengaruhi data yang dikirim AWS IoT SiteWise sejak saat itu. AWS IoT SiteWise tidak mengambil data dari penyimpanan hangat atau dingin untuk mengisi tingkat panas. Misalnya, jika periode retensi penyimpanan hot tier awalnya ditetapkan selama 30 hari dan kemudian ditingkatkan menjadi 60 hari, dibutuhkan 30 hari untuk penyimpanan hot tier berisi data senilai 60 hari.

Topik

- [Konfigurasi pengaturan penyimpanan untuk tingkat hangat \(konsol\)](#)
- [Konfigurasi pengaturan penyimpanan untuk tingkat hangat \(AWS CLI\)](#)
- [Konfigurasi pengaturan penyimpanan untuk tingkat dingin \(konsol\)](#)
- [Konfigurasi pengaturan penyimpanan untuk tingkat dingin \(AWS CLI\)](#)

Konfigurasi pengaturan penyimpanan untuk tingkat hangat (konsol)

Prosedur berikut menunjukkan cara mengonfigurasi pengaturan penyimpanan untuk mereplikasi data ke tingkat hangat di AWS IoT SiteWise konsol.

Untuk mengonfigurasi pengaturan penyimpanan di konsol

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, di bawah Pengaturan, pilih Penyimpanan.
3. Di sudut kanan atas, pilih Edit.
4. Pada halaman Edit penyimpanan, lakukan hal berikut:
5. Untuk pengaturan Hot tier, lakukan hal berikut:
 - Jika Anda ingin menetapkan periode retensi untuk berapa lama data disimpan di hot tier sebelum dihapus, dan dipindahkan ke penyimpanan tingkat hangat yang dikelola layanan, pilih Aktifkan periode retensi.
 - Untuk mengonfigurasi periode retensi, masukkan bilangan bulat dan pilih unit. Periode retensi harus lebih besar dari atau sama dengan 30 hari.

AWS IoT SiteWise menghapus data apa pun di tingkat panas yang lebih lama dari periode retensi. Jika Anda tidak menetapkan periode retensi, data Anda disimpan tanpa batas waktu.

6. (Disarankan) Untuk pengaturan tingkat hangat, lakukan hal berikut:

- Untuk memilih penyimpanan tingkat hangat, pilih Saya mengonfirmasi keikutsertaan penyimpanan tingkat hangat untuk memilih penyimpanan tingkat hangat.
- (Opsional) Untuk mengonfigurasi periode retensi, masukkan bilangan bulat dan pilih unit. Periode retensi harus lebih besar dari atau sama dengan 365 hari.

AWS IoT SiteWise menghapus data di tingkat hangat yang ada lebih awal dari periode retensi. Jika Anda tidak menetapkan periode retensi, data Anda disimpan tanpa batas waktu.

Note

- Saat Anda memilih tingkat hangat, konfigurasi hanya akan ditampilkan sekali.
- Untuk mengatur retensi tingkat panas, Anda harus memiliki penyimpanan tingkat hangat atau dingin. Untuk efisiensi biaya dan pengambilan data historis, AWS IoT SiteWise merekomendasikan agar Anda menyimpan data jangka panjang di tingkat hangat.
- Untuk mengatur retensi tingkat hangat, Anda harus memiliki penyimpanan tingkat dingin.

7. Pilih Simpan untuk menyimpan pengaturan penyimpanan Anda.

Di bagian AWS IoT SiteWise penyimpanan, penyimpanan tingkat hangat berada di salah satu status berikut:

- Diaktifkan — Jika data Anda ada sebelum periode retensi tingkat AWS IoT SiteWise panas, pindahkan data ke tingkat hangat.
- Dinonaktifkan - Penyimpanan tingkat hangat dinonaktifkan.

Konfigurasi pengaturan penyimpanan untuk tingkat hangat (AWS CLI)

Anda dapat mengonfigurasi pengaturan penyimpanan untuk memindahkan data ke tingkat hangat dengan menggunakan perintah AWS CLI dan perintah berikut.

Untuk mencegah penggantian konfigurasi yang ada, ambil informasi konfigurasi penyimpanan saat ini dengan menjalankan perintah berikut:


```
aws iotsitewise describe-storage-configuration
```

Example respons tanpa konfigurasi tingkat dingin yang ada

```
{
  "storageType": "SITEWISE_DEFAULT_STORAGE",
  "disassociatedDataStorage": "ENABLED",
  "configurationStatus": {
    "state": "ACTIVE"
  },
  "lastUpdateDate": "2021-10-14T15:53:35-07:00",
  "warmTier": "DISABLED"
}
```

Example respons dengan konfigurasi tingkat dingin yang ada

```
{
  "storageType": "MULTI_LAYER_STORAGE",
  "multiLayerStorage": {
    "customerManagedS3Storage": {
      "s3ResourceArn": "arn:aws:s3:::bucket-name/prefix/",
      "roleArn": "arn:aws:iam::aws-account-id:role/role-name"
    }
  },
  "disassociatedDataStorage": "ENABLED",
  "retentionPeriod": {
    "numberOfDays": retention-in-days
  },
  "configurationStatus": {
    "state": "ACTIVE"
  },
  "lastUpdateDate": "2023-10-25T15:59:46-07:00",
  "warmTier": "DISABLED"
}
```

Konfigurasi pengaturan penyimpanan untuk tingkat hangat dengan AWS CLI

Jalankan perintah berikut untuk mengkonfigurasi pengaturan penyimpanan. Ganti `file-name` dengan nama file yang berisi konfigurasi AWS IoT SiteWise penyimpanan.

```
aws iotsitewise put-storage-configuration --cli-input-json file://file-name.json
```

Example AWS IoT SiteWise konfigurasi dengan tingkat panas dan hangat

```
{
    "storageType": "SITEWISE_DEFAULT_STORAGE",
    "disassociatedDataStorage": "ENABLED",
    "warmTier": "ENABLED",
    "retentionPeriod": {
        "numberOfDays": hot-tier-retention-in-days
    }
}
```

hot-tier-retention-in-day harus bilangan bulat yang lebih besar dari atau sama dengan 30 hari.

Example response

```
{
    "storageType": "SITEWISE_DEFAULT_STORAGE",
    "configurationStatus": {
        "state": "UPDATE_IN_PROGRESS"
    }
}
```

Jika Anda mengaktifkan penyimpanan tingkat dingin, lihat [Konfigurasi pengaturan penyimpanan dengan AWS CLI dan tingkat dingin yang ada](#).

Konfigurasi pengaturan penyimpanan dengan AWS CLI dan tingkat dingin yang ada

Konfigurasi pengaturan penyimpanan menggunakan AWS CLI penyimpanan tingkat dingin yang ada


- Jalankan perintah berikut untuk mengkonfigurasi pengaturan penyimpanan. Ganti *nama file* dengan nama file yang berisi konfigurasi AWS IoT SiteWise penyimpanan.

```
aws iotsitewise put-storage-configuration --cli-input-json file://file-name.json
```

Example AWS IoT SiteWise konfigurasi penyimpanan

- Ganti nama *ember* dengan *nama bucket* Amazon S3 Anda.

- Ganti *awalan* dengan *awalan* Amazon S3 Anda.
- Ganti *aws-account-id* dengan ID AWS akun Anda.
- Ganti *role-name* dengan nama peran akses Amazon S3 yang AWS IoT SiteWise memungkinkan untuk mengirim data ke Amazon S3.
- Ganti *hot-tier-retention-in-hari* dengan bilangan bulat yang lebih besar dari atau sama dengan 30 hari.
- Ganti *warm-tier-retention-in-hari* dengan bilangan bulat yang lebih besar dari atau sama dengan 365 hari.

 Note

AWS IoT SiteWise akan menghapus data apa pun di tingkat hangat yang lebih tua dari periode retensi tingkat dingin. Jika Anda tidak menetapkan periode retensi, data Anda disimpan tanpa batas waktu.

```
{
  "storageType": "MULTI_LAYER_STORAGE",
  "multiLayerStorage": {
    "customerManagedS3Storage": {
      "s3ResourceArn": "arn:aws:s3:::bucket-name/prefix/",
      "roleArn": "arn:aws:iam::aws-account-id:role/role-name"
    }
  },
  "disassociatedDataStorage": "ENABLED",
  "retentionPeriod": {
    "numberOfDays": hot-tier-retention-in-days
  },
  "warmTier": "ENABLED",
  "warmTierRetentionPeriod": {
    "numberOfDays": warm-tier-retention-in-days
  }
}
```

Example response

```
{
  "storageType": "MULTI_LAYER_STORAGE",
```

```
"configurationStatus": {  
  "state": "UPDATE_IN_PROGRESS"  
}  
}
```

Konfigurasi pengaturan penyimpanan untuk tingkat dingin (konsol)

Prosedur berikut menunjukkan cara mengonfigurasi pengaturan penyimpanan untuk mereplikasi data ke tingkat dingin di AWS IoT SiteWise konsol.

Untuk mengkonfigurasi pengaturan penyimpanan di konsol

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, di bawah Pengaturan, pilih Penyimpanan.
3. Di sudut kanan atas, pilih Edit.
4. Pada halaman Edit penyimpanan, lakukan hal berikut:
 - a. Untuk pengaturan Penyimpanan, pilih Aktifkan penyimpanan tingkat dingin. Penyimpanan tingkat dingin dinonaktifkan secara default.
 - b. Untuk lokasi bucket S3, masukkan nama bucket Amazon S3 yang ada dan awalan.

Note

- Amazon S3 menggunakan awalan sebagai nama folder di bucket Amazon S3. Awalan harus memiliki 1-255 karakter dan diakhiri dengan garis miring (/). AWS IoT SiteWise Data Anda disimpan di folder ini.
- Jika Anda tidak memiliki bucket Amazon S3, pilih View, lalu buat bucket di konsol Amazon S3. Untuk informasi selengkapnya, lihat [Membuat bucket S3 pertama Anda](#) di Panduan Pengguna Amazon S3.

- c. Untuk peran akses S3, lakukan salah satu hal berikut:
 - Pilih Buat peran dari templat AWS terkelola, AWS secara otomatis membuat peran IAM yang memungkinkan AWS IoT SiteWise untuk mengirim data ke Amazon S3.
 - Pilih Gunakan peran yang ada, lalu pilih peran yang Anda buat dari daftar.

Note

- Anda harus menggunakan nama bucket Amazon S3 yang sama untuk lokasi bucket S3 yang Anda gunakan pada langkah sebelumnya dan dalam kebijakan IAM Anda.
- Pastikan bahwa peran Anda memiliki izin yang ditunjukkan dalam contoh berikut.

Example kebijakan izin:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject",
        "s3:GetBucketLocation",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::bucket-name",
        "arn:aws:s3:::bucket-name/*"
      ]
    }
  ]
}
```

Ganti nama *ember* dengan nama bucket Amazon S3 Anda.

- d. Untuk mengatur hot tier, lihat Langkah 5 di [Konfigurasi pengaturan penyimpanan untuk tingkat hangat \(konsol\)](#).
- e. (Opsional) Untuk AWS IoT Analytics integrasi, lakukan hal berikut.
 - i. Jika Anda AWS IoT Analytics ingin menggunakan kueri data Anda, pilih Penyimpanan AWS IoT Analytics data yang diaktifkan.

- ii. AWS IoT SiteWise menghasilkan nama untuk penyimpanan data Anda atau Anda dapat memasukkan nama yang berbeda.

AWS IoT SiteWise secara otomatis membuat penyimpanan data AWS IoT Analytics untuk menyimpan data Anda. Untuk menanyakan data, Anda dapat menggunakan AWS IoT Analytics untuk membuat kumpulan data. Untuk informasi selengkapnya, lihat [Bekerja dengan AWS IoT SiteWise data](#) di Panduan AWS IoT Analytics Pengguna.

- f. Pilih Simpan.

Di bagian AWS IoT SiteWise penyimpanan, penyimpanan tingkat dingin dapat menjadi salah satu dari nilai berikut:

- Diaktifkan - AWS IoT SiteWise mereplikasi data Anda ke bucket Amazon S3 yang ditentukan.
- Mengaktifkan — AWS IoT SiteWise memproses permintaan Anda untuk mengaktifkan penyimpanan tingkat dingin. Proses ini bisa memakan waktu beberapa menit untuk menyelesaikannya.
- Enable_Failed — AWS IoT SiteWise tidak dapat memproses permintaan Anda untuk mengaktifkan penyimpanan tingkat dingin. Jika Anda mengaktifkan AWS IoT SiteWise untuk mengirim log ke Amazon CloudWatch Logs, Anda dapat menggunakan log ini untuk memecahkan masalah. Untuk informasi selengkapnya, lihat [Pemantauan dengan Amazon CloudWatch Logs](#).
- Dinonaktifkan - Penyimpanan tingkat dingin dinonaktifkan.

Konfigurasi pengaturan penyimpanan untuk tingkat dingin (AWS CLI)

Prosedur berikut menunjukkan cara mengkonfigurasi pengaturan penyimpanan untuk mereplikasi data ke tingkat dingin menggunakan AWS CLI.

Untuk mengkonfigurasi pengaturan penyimpanan menggunakan AWS CLI

1. Untuk mengeksport data ke bucket Amazon S3 di akun Anda, jalankan perintah berikut untuk mengonfigurasi pengaturan penyimpanan. Ganti *nama file* dengan nama file yang berisi konfigurasi AWS IoT SiteWise penyimpanan.

```
aws iotsitewise put-storage-configuration --cli-input-json file://file-name.json
```

Example AWS IoT SiteWise konfigurasi penyimpanan

- Ganti nama *ember* dengan *nama bucket* Amazon S3 Anda.
- Ganti *awalan* dengan *awalan* Amazon S3 Anda.
- Ganti *aws-account-id* dengan ID AWS akun Anda.
- Ganti *role-name* dengan nama peran akses Amazon S3 yang AWS IoT SiteWise memungkinkan untuk mengirim data ke Amazon S3.
- Ganti *retention-in-days* dengan bilangan bulat lebih besar dari atau sama dengan 30 hari.

```
{
  "storageType": "MULTI_LAYER_STORAGE",
  "multiLayerStorage": {
    "customerManagedS3Storage": {
      "s3ResourceArn": "arn:aws:s3:::bucket-name/prefix",
      "roleArn": "arn:aws:iam::aws-account-id:role/role-name"
    }
  },
  "retentionPeriod": {
    "numberOfDays": retention-in-days,
    "unlimited": false
  }
}
```

Note

- Anda harus menggunakan nama bucket Amazon S3 yang sama dalam konfigurasi AWS IoT SiteWise penyimpanan dan kebijakan IAM.
- Pastikan bahwa peran Anda memiliki izin yang ditunjukkan dalam contoh berikut.

Example kebijakan izin:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
```

```

        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject",
        "s3:GetBucketLocation",
        "s3:ListBucket"
    ],
    "Resource": [
        "arn:aws:s3:::bucket-name",
        "arn:aws:s3:::bucket-name/*"
    ]
}
]
}

```

Ganti nama *ember* dengan nama *bucket* Amazon S3 Anda.

Example response

```

{
  "storageType": "MULTI_LAYER_STORAGE",
  "retentionPeriod": {
    "numberOfDays": 100,
    "unlimited": false
  },
  "configurationStatus": {
    "state": "UPDATE_IN_PROGRESS"
  }
}

```

Note

Diperlukan beberapa menit AWS IoT SiteWise untuk memperbarui konfigurasi penyimpanan.

2. Untuk mengambil informasi konfigurasi penyimpanan, jalankan perintah berikut.

```
aws iotsitewise describe-storage-configuration
```


Example response

```
{
  "storageType": "MULTI_LAYER_STORAGE",
  "multiLayerStorage": {
    "customerManagedS3Storage": {
      "s3ResourceArn": "arn:aws:s3::DOC-EXAMPLE-BUCKET/torque/",
      "roleArn": "arn:aws:iam::123456789012:role/SWAccessS3Role"
    }
  },
  "retentionPeriod": {
    "numberOfDays": 100,
    "unlimited": false
  },
  "configurationStatus": {
    "state": "ACTIVE"
  },
  "lastUpdateDate": "2021-03-30T15:54:14-07:00"
}
```

3. Untuk berhenti mengekspor data ke bucket Amazon S3, jalankan perintah berikut untuk mengonfigurasi pengaturan penyimpanan.

```
aws iotsitewise put-storage-configuration --storage-type SITEWISE_DEFAULT_STORAGE
```

Note

Secara default, data Anda hanya disimpan di tingkat panas AWS IoT SiteWise.

Example response

```
{
  "storageType": "SITEWISE_DEFAULT_STORAGE",
  "configurationStatus": {
    "state": "UPDATE_IN_PROGRESS"
  }
}
```

4. Untuk mengambil informasi konfigurasi penyimpanan, jalankan perintah berikut.

```
aws iotsitewise describe-storage-configuration
```

Example response

```
{
  "storageType": "SITEWISE_DEFAULT_STORAGE",
  "configurationStatus": {
    "state": "ACTIVE"
  },
  "lastUpdateDate": "2021-03-30T15:57:14-07:00"
}
```

(Opsional) Buat penyimpanan AWS IoT Analytics data (AWS CLI)

Penyimpanan AWS IoT Analytics data adalah repositori yang dapat diskalakan dan dapat dikueri yang menerima dan menyimpan data. Anda dapat menggunakan AWS IoT SiteWise konsol atau AWS IoT Analytics API untuk membuat penyimpanan AWS IoT Analytics data untuk menyimpan AWS IoT SiteWise data Anda. Untuk menanyakan data, Anda membuat kumpulan data dengan menggunakan [AWS IoT Analytics](#). Untuk informasi selengkapnya, lihat [Bekerja dengan AWS IoT SiteWise data](#) di Panduan AWS IoT Analytics Pengguna.

Langkah-langkah berikut digunakan AWS CLI untuk membuat penyimpanan data di AWS IoT Analytics.

Untuk membuat penyimpanan data, jalankan perintah berikut. Ganti *nama file* dengan nama file yang berisi konfigurasi penyimpanan data.

```
aws iotanalytics create-datastore --cli-input-json file://file-name.json
```

Note

- Anda harus menentukan nama bucket Amazon S3 yang ada. Jika Anda tidak memiliki ember Amazon S3, buat ember terlebih dahulu. Untuk informasi selengkapnya, lihat [Membuat bucket S3 pertama Anda](#) di Panduan Pengguna Amazon S3.

- Anda harus menggunakan nama bucket Amazon S3 yang sama dalam konfigurasi AWS IoT SiteWise penyimpanan, kebijakan IAM, dan konfigurasi penyimpanan AWS IoT Analytics data.

Example AWS IoT Analytics konfigurasi penyimpanan data

Ganti *data-store-name* dan *s3-bucket-name* dengan nama penyimpanan data dan AWS IoT Analytics nama bucket Amazon S3 Anda.

```
{
  "datastoreName": "data-store-name",
  "datastoreStorage": {
    "iotSiteWiseMultiLayerStorage": {
      "customerManagedS3Storage": {
        "bucket": "s3-bucket-name"
      }
    }
  },
  "retentionPeriod": {
    "numberOfDays": 90
  }
}
```

Example response

```
{
  "datastoreName": "datastore_IoTSiteWise_demo",
  "datastoreArn": "arn:aws:iotanalytics:us-west-2:123456789012:datastore/
datastore_IoTSiteWise_demo",
  "retentionPeriod": {
    "numberOfDays": 90,
    "unlimited": false
  }
}
```

Memecahkan masalah pengaturan penyimpanan

Gunakan informasi berikut untuk memecahkan masalah dan menyelesaikan masalah dengan konfigurasi penyimpanan.

Masalah

- [Kesalahan: Bucket tidak ada](#)
- [Kesalahan: Akses ditolak ke jalur Amazon S3](#)
- [Kesalahan: Peran ARN tidak dapat diasumsikan](#)
- [Kesalahan: Gagal mengakses bucket Amazon S3 lintas wilayah](#)

Kesalahan: Bucket tidak ada

Solusi: AWS IoT SiteWise tidak dapat menemukan ember Amazon S3 Anda. Pastikan Anda memasukkan nama bucket Amazon S3 yang ada di Wilayah saat ini.

Kesalahan: Akses ditolak ke jalur Amazon S3

Solusi: AWS IoT SiteWise tidak dapat mengakses bucket Amazon S3 Anda. Lakukan hal-hal berikut:

- Pastikan Anda menggunakan bucket Amazon S3 yang sama dengan yang Anda tentukan dalam kebijakan IAM.
- Pastikan bahwa peran Anda memiliki izin yang ditunjukkan dalam contoh berikut.

Example kebijakan izin

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject",
        "s3:GetBucketLocation",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::bucket-name",
        "arn:aws:s3:::bucket-name/*"
      ]
    }
  ]
}
```

```
}
```

Ganti nama *ember* dengan nama *bucket* Amazon S3 Anda.

Kesalahan: Peran ARN tidak dapat diasumsikan

Solusi: AWS IoT SiteWise tidak dapat mengambil peran IAM atas nama Anda. Pastikan peran Anda mempercayai layanan berikut: `iotsitewise.amazonaws.com`. Untuk informasi selengkapnya, lihat [Saya tidak dapat mengambil peran](#), lihat Panduan Pengguna IAM.

Kesalahan: Gagal mengakses bucket Amazon S3 lintas wilayah

Solusi: Bucket Amazon S3 yang Anda tentukan berada di Wilayah yang berbeda AWS . Pastikan bucket dan AWS IoT SiteWise aset Amazon S3 Anda berada di Wilayah yang sama.

Jalur file dan skema data yang disimpan di tingkat dingin

AWS IoT SiteWise menyimpan data Anda di tingkat dingin dengan mereplikasi deret waktu, termasuk pengukuran, metrik, transformasi dan agregat, serta definisi model aset dan aset. Berikut ini menjelaskan jalur file dan skema data yang dikirim ke tingkat dingin.

Topik

- [Data peralatan \(pengukuran\)](#)
- [Metrik, transformasi, dan agregat](#)
- [Metadata aset](#)
- [Metadata hierarki aset](#)
- [File indeks data penyimpanan](#)

Data peralatan (pengukuran)

AWS IoT SiteWise mengekspor data peralatan (pengukuran) ke tingkat dingin setiap enam jam sekali. Data mentah disimpan dalam tingkat dingin dalam format [Apache AVRO](#) (.avro).

Jalur berkas

AWS IoT SiteWise menyimpan data peralatan (pengukuran) di tingkat dingin menggunakan templat berikut.

```
{keyPrefix}/raw/startYear={startYear}/startMonth={startMonth}/startDay={startDay}/
seriesBucket={seriesBucket}/raw_{timeseriesId}_{startTimestamp}_{quality}.avro
```

Setiap path file ke data mentah di Amazon S3 berisi komponen-komponen berikut.

Komponen jalur	Deskripsi
keyPrefix	Awalan Amazon S3 yang Anda tentukan dalam konfigurasi penyimpanan. AWS IoT SiteWise Amazon S3 menggunakan awalan sebagai nama folder di bucket.
raw	Folder yang menyimpan data deret waktu dari peralatan (pengukuran). rawFolder disimpan di folder awalan.
seriesBucket	<p>Angka heksadesimal antara 00 dan ff. Angka ini berasal dari <code>timeSeriesId</code>. Partisi ini digunakan untuk meningkatkan throughput saat AWS IoT SiteWise menulis ke tingkat dingin. Saat Anda menggunakan Amazon Athena untuk menjalankan kueri, Anda dapat menggunakan partisi untuk partisi butir halus guna meningkatkan kinerja kueri.</p> <p><code>seriesBucket</code> dan <code>timeSeriesBucket</code> dalam metadata aset adalah angka yang sama.</p>
startYear	Tahun waktu mulai eksklusif yang terkait dengan data deret waktu.
startMonth	Bulan waktu mulai eksklusif yang terkait dengan data deret waktu.
startDay	Hari bulan dari waktu mulai eksklusif yang terkait dengan data deret waktu.

Komponen jalur	Deskripsi
fileName	<p>Nama file menggunakan karakter garis bawah (_) sebagai pembatas untuk memisahkan yang berikut:</p> <ul style="list-style-type: none"> rawAwalan. timeSeriesId Nilainya. Stempel waktu zaman dari waktu mulai eksklusif yang terkait dengan data deret waktu. Kualitas data. Nilai-nilai yang valid: GOOD, BAD, dan UNCERTAIN . Untuk informasi selengkapnya, lihat AssetPropertyValuedi Referensi AWS IoT SiteWise API. <p>File disimpan dalam .avro format dengan menggunakan kompresi Snappy.</p>

Example jalur file ke data mentah di tingkat dingin

```
keyPrefix/raw/startYear=2021/startMonth=1/startDay=2/seriesBucket=a2/
raw_7020c8e2-e6db-40fa-9845-ed0ddddd4c77d_95e63da7-d34e-43e1-
bc6f-1b490154b07a_1609577700_GOOD.avro
```

Bidang

Skema data mentah yang diekspor ke tingkat dingin berisi bidang-bidang berikut.

Nama bidang	Jenis yang didukung	Jenis default	Deskripsi
seriesId	string	N/A	ID yang mengidentifikasi data deret waktu dari peralatan (pengukuran). Anda dapat menggunakan

Nama bidang	Jenis yang didukung	Jenis default	Deskripsi
			an bidang ini untuk menggabungkan data mentah dan metadata aset dalam kueri.
<code>timeInSeconds</code>	<code>long</code>	N/A	Tanggal stempel waktu, dalam hitungan detik, dalam format epoch Unix. Data nanodetik pecahan disediakan oleh. <code>offsetInNanos</code>
<code>offsetInNanos</code>	<code>long</code>	N/A	Nanodetik offset dari. <code>timeInSeconds</code>
<code>quality</code>	<code>string</code>	N/A	Kualitas nilai deret waktu.
<code>doubleValue</code>	<code>double</code> atau <code>null</code>	<code>null</code>	Data deret waktu tipe ganda (nomor floating point).
<code>stringValue</code>	<code>string</code> atau <code>null</code>	<code>null</code>	Data deret waktu dari tipe string (urutan karakter).
<code>integerValue</code>	<code>int</code> atau <code>null</code>	<code>null</code>	Data deret waktu tipe integer (bilangan bulat).
<code>booleanValue</code>	<code>boolean</code> atau <code>null</code>	<code>null</code>	Data deret waktu tipe Boolean (benar atau salah).

Nama bidang	Jenis yang didukung	Jenis default	Deskripsi
jsonValue	string atau null	null	Data deret waktu tipe JSON (tipe data kompleks yang disimpan sebagai string).
recordVersion	long atau null	null	Nomor versi untuk catatan. Anda dapat menggunakan nomor versi untuk memilih catatan terbaru. Catatan yang lebih baru memiliki nomor versi yang lebih besar.

Example data mentah di tingkat dingin

```
{
  "seriesId": "e9687d2a-0dbe-4f65-9ed6-6f443cba41f7_95e63da7-d34e-43e1-bc6f-1b490154b07a",
  "timeInSeconds": 1625675887,
  "offsetInNanos": 0,
  "quality": "GOOD",
  "doubleValue": {"double": 0.75},
  "stringValue": null,
  "integerValue": null,
  "booleanValue": null,
  "jsonValue": null,
  "recordVersion": 1
},
{
  "seriesId": "e9687d2a-0dbe-4f65-9ed6-6f443cba41f7_95e63da7-d34e-43e1-bc6f-1b490154b07a",
  "timeInSeconds": 1625675889,
  "offsetInNanos": 0,
  "quality": "GOOD",
  "doubleValue": {"double": 0.69},
  "stringValue": null,
  "integerValue": null,
  "booleanValue": null,
  "jsonValue": null,
  "recordVersion": 2
},
{
  "seriesId": "e9687d2a-0dbe-4f65-9ed6-6f443cba41f7_95e63da7-d34e-43e1-bc6f-1b490154b07a",
  "timeInSeconds": 1625675890,
  "offsetInNanos": 0,
  "quality": "GOOD",
  "doubleValue": {"double": 0.66},
  "stringValue": null,
  "integerValue": null,
  "booleanValue": null,
  "jsonValue": null,
  "recordVersion": 3
},
{
  "seriesId": "e9687d2a-0dbe-4f65-9ed6-6f443cba41f7_95e63da7-d34e-43e1-bc6f-1b490154b07a",
  "timeInSeconds": 1625675891,
  "offsetInNanos": 0,
  "quality": "GOOD",
  "doubleValue": {"double": 0.92},
  "stringValue": null,
  "integerValue": null,
  "booleanValue": null,
  "jsonValue": null,
  "recordVersion": 4
},
{
  "seriesId": "e9687d2a-0dbe-4f65-9ed6-6f443cba41f7_95e63da7-d34e-43e1-bc6f-1b490154b07a",
  "timeInSeconds": 1625675892,
  "offsetInNanos": 0,
  "quality": "GOOD",
  "doubleValue": {"double": 0.73},
  "stringValue": null,
  "integerValue": null,
  "booleanValue": null,
  "jsonValue": null,
  "recordVersion": 5
}
```

Metrik, transformasi, dan agregat

AWS IoT SiteWise mengekspor metrik, transformasi, dan agregat ke tingkat dingin setiap enam jam sekali. Metrik, transformasi, dan agregat disimpan dalam tingkat dingin dalam format [Apache AVRO](#) (). `.avro`

Jalur berkas

AWS IoT SiteWise menyimpan metrik, transformasi, dan agregat di tingkat dingin menggunakan templat berikut.

```
{keyPrefix}/agg/startYear={startYear}/startMonth={startMonth}/startDay={startDay}/
seriesBucket={seriesBucket}/agg_{timeseriesId}_{startTimestamp}_{quality}.avro
```

Setiap jalur file ke metrik, transformasi, dan agregat di Amazon S3 berisi komponen berikut.

Komponen jalur	Deskripsi
keyPrefix	Awalan Amazon S3 yang Anda tentukan dalam konfigurasi penyimpanan. AWS IoT SiteWise Amazon S3 menggunakan awalan sebagai nama folder di bucket.
agg	Folder yang menyimpan data deret waktu dari metrik. <code>aggFolder</code> disimpan di folder awalan.
seriesBucket	Angka heksadesimal antara 00 dan ff. Angka ini berasal dari <code>timeSeriesId</code> . Partisi ini digunakan untuk meningkatkan throughput saat AWS IoT SiteWise menulis ke tingkat dingin. Saat Anda menggunakan Amazon Athena untuk menjalankan kueri, Anda dapat menggunakan partisi untuk partisi butir halus guna meningkatkan kinerja kueri. <code>seriesBucket</code> dan <code>timeSeriesBucket</code> dalam metadata aset adalah angka yang sama.

Komponen jalur	Deskripsi
<code>startYear</code>	Tahun waktu mulai eksklusif yang terkait dengan data deret waktu.
<code>startMonth</code>	Bulan waktu mulai eksklusif yang terkait dengan data deret waktu.
<code>startDay</code>	Hari bulan dari waktu mulai eksklusif yang terkait dengan data deret waktu.
<code>fileName</code>	<p>Nama file menggunakan karakter garis bawah (<code>_</code>) sebagai pembatas untuk memisahkan yang berikut:</p> <ul style="list-style-type: none"> <code>rawAwalan</code>. <code>timeSeriesId</code> Nilainya. Stempel waktu zaman dari waktu mulai eksklusif yang terkait dengan data deret waktu. Kualitas data. Nilai-nilai yang valid: <code>GOOD</code>, <code>BAD</code>, dan <code>UNCERTAIN</code> . Untuk informasi selengkapnya, lihat AssetPropertyValuedi Referensi AWS IoT SiteWise API. <p>File disimpan dalam <code>.avro</code> format dengan menggunakan kompresi Snappy.</p>

Example jalur file ke metrik di tingkat dingin

```
keyPrefix/agg/startYear=2021/startMonth=1/startDay=2/seriesBucket=a2/agg_7020c8e2-e6db-40fa-9845-ed0dddd4c77d_95e63da7-d34e-43e1-bc6f-1b490154b07a_1609577700_GOOD.avro
```

Bidang

Skema metrik, transformasi, dan agregat yang diekspor ke tingkat dingin berisi bidang berikut.

Nama bidang	Jenis yang didukung	Jenis default	Deskripsi
<code>seriesId</code>	<code>string</code>	N/A	ID yang mengidentifikasi data deret waktu dari peralatan, metrik, atau transformasi. Anda dapat menggunakan bidang ini untuk menggabungkan data mentah dan metadata aset dalam kueri.
<code>timeInSeconds</code>	<code>long</code>	N/A	Tanggal stempel waktu, dalam hitungan detik, dalam format epoch Unix. Data nanodetik pecahan disediakan oleh <code>offsetInNanos</code>
<code>offsetInNanos</code>	<code>long</code>	N/A	Nanodetik offset dari <code>timeInSeconds</code>
<code>quality</code>	<code>string</code>	N/A	Kualitas yang digunakan untuk memfilter data aset.
<code>resolution</code>	<code>string</code>	N/A	Interval waktu untuk mengumpulkan data.
<code>count</code>	<code>double</code> atau <code>null</code>	<code>null</code>	Jumlah total titik data untuk variabel yang diberikan selama interval waktu saat ini.

Nama bidang	Jenis yang didukung	Jenis default	Deskripsi
average	double atau null	null	Rata-rata nilai variabel yang diberikan selama interval waktu saat ini.
min	double atau null	null	Minimum nilai variabel yang diberikan selama interval waktu saat ini.
max	boolean atau null	null	Maksimum nilai variabel yang diberikan selama interval waktu saat ini.
sum	string atau null	null	Jumlah nilai variabel yang diberikan selama interval waktu saat ini.
recordVersion	long atau null	null	Nomor versi untuk catatan. Anda dapat menggunakan nomor versi untuk memilih catatan terbaru. Catatan yang lebih baru memiliki nomor versi yang lebih besar.

Example Data metrik di tingkat dingin

```
{
  "seriesId": "f74c2828-5317-4df3-ba16-6d41b5bcb531",
  "timeInSeconds": 1637334060,
  "offsetInNanos": 0,
  "quality": "GOOD",
  "resolution": "1s",
  "value": {
    "double": 16.0,
    "min": {"double": 1.0},
    "max": {"double": 31.0},
    "sum": {"double": 496.0},
    "recordVersion": null
  }
}
```

```

{"seriesId":"f74c2828-5317-4df3-
ba16-6d41b5bcb531","timeInSeconds":1637334120,"offsetInNanos":0,"quality":"GOOD","resolution":
{"double":46.0},"min":{"double":32.0},"max":{"double":60.0},"sum":
{"double":1334.0},"recordVersion":null}
{"seriesId":"f74c2828-5317-4df3-
ba16-6d41b5bcb531","timeInSeconds":1637334540,"offsetInNanos":0,"quality":"GOOD","resolution":
{"double":16.0},"min":{"double":1.0},"max":{"double":31.0},"sum":
{"double":496.0},"recordVersion":null}
{"seriesId":"f74c2828-5317-4df3-
ba16-6d41b5bcb531","timeInSeconds":1637334600,"offsetInNanos":0,"quality":"GOOD","resolution":
{"double":46.0},"min":{"double":32.0},"max":{"double":60.0},"sum":
{"double":1334.0},"recordVersion":null}
{"seriesId":"f74c2828-5317-4df3-
ba16-6d41b5bcb531","timeInSeconds":1637335020,"offsetInNanos":0,"quality":"GOOD","resolution":
{"double":16.0},"min":{"double":1.0},"max":{"double":31.0},"sum":
{"double":496.0},"recordVersion":null}

```

Metadata aset

Saat Anda mengaktifkan AWS IoT SiteWise untuk mengekspor data ke tingkat dingin untuk pertama kalinya, metadata aset diekspor ke tingkat dingin. Setelah konfigurasi awal, AWS IoT SiteWise ekspor metadata aset ke tingkat hanya jika Anda mengubah definisi model aset atau definisi aset. Metadata aset disimpan dalam tingkat dingin dalam format JSON () yang dibatasi baris baru. .ndjson

Jalur berkas

AWS IoT SiteWise menyimpan metadata aset di tingkat dingin menggunakan templat berikut.

```
{keyPrefix}/asset_metadata/asset_{assetId}.ndjson
```

Setiap jalur file ke metadata aset di tingkat dingin berisi komponen berikut.

Komponen jalur	Deskripsi
keyPrefix	Awalan Amazon S3 yang Anda tentukan dalam konfigurasi penyimpanan AWS IoT SiteWise s. Amazon S3 menggunakan awalan sebagai nama folder di bucket.

Komponen jalur	Deskripsi
asset_metadata	Folder yang menyimpan metadata aset. asset_metadata Folder disimpan di folder awalan.
fileName	Nama file menggunakan karakter garis bawah (_) sebagai pembatas untuk memisahkan yang berikut: <ul style="list-style-type: none"> assetAwalan. assetIdNilainya. <p>File disimpan dalam .ndjson format.</p>

Example jalur file ke metadata aset di tingkat yang lebih dingin

keyPrefix/asset_metadata/asset_35901915-d476-4dca-8637-d9ed4df939ed.ndjson

Bidang

Skema metadata aset yang diekspor ke tingkat dingin berisi bidang-bidang berikut.

Nama bidang	Deskripsi
assetId	ID aset.
assetName	Nama aset.
assetExternalId	ID eksternal aset.
assetModelId	ID model aset yang digunakan untuk membuat aset ini.
assetModelName	Nama model aset.
assetModelExternalId	ID eksternal dari model aset.
assetPropertyId	ID dari properti aset.

Nama bidang	Deskripsi
<code>assetPropertyName</code>	Nama properti aset.
<code>assetPropertyExternalId</code>	ID eksternal dari properti aset.
<code>assetPropertyDataType</code>	Tipe data properti aset.
<code>assetPropertyUnit</code>	Unit properti aset (misalnya, Newtons dan RPM).
<code>assetPropertyAlias</code>	Alias yang mengidentifikasi properti aset, seperti jalur aliran data server OPC-UA (misalnya, <code>/company/windfarm/3/turbine/7/temperature</code>)
<code>timeSeriesId</code>	ID yang mengidentifikasi data deret waktu dari peralatan, metrik, atau transformasi. Anda dapat menggunakan bidang ini untuk menggabungkan data mentah dan metadata aset dalam kueri.
<code>timeSeriesBucket</code>	<p>Angka heksadesimal antara 00 dan ff. Angka ini berasal dari <code>timeSeriesId</code>. Partisi ini digunakan untuk meningkatkan throughput saat AWS IoT SiteWise menulis ke tingkat dingin. Saat Anda menggunakan Amazon Athena untuk menjalankan kueri, Anda dapat menggunakan partisi untuk partisi butir halus guna meningkatkan kinerja kueri.</p> <p><code>timeSeriesBucket</code> dan <code>seriesBucket</code> di jalur file ke data mentah adalah nomor yang sama.</p>
<code>assetCompositeModelId</code>	ID dari model komposit.
<code>assetCompositeModelExternalId</code>	ID eksternal dari model komposit.

Nama bidang	Deskripsi
assetCompositeModelDescription	Deskripsi dari model komposit.
assetCompositeModelName	Nama dari model komposit.
assetCompositeModelType	Jenis dari model komposit. Untuk model komposit alarm, jenis ini berupa AWS/ALARM .
assetCreationDate	Tanggal aset dibuat, dalam waktu zaman Unix.
assetLastUpdateDate	Tanggal aset terakhir diperbarui, dalam waktu zaman Unix.
assetStatusErrorCode	Kode kesalahan.
assetStatusErrorMessage	Pesan kesalahan.
assetStatusState	Status aset saat ini.

Example metadata aset di tingkat dingin

```

{"assetId":"7020c8e2-e6db-40fa-9845-ed0dddd4c77d","assetExternalId":null,"assetName":"Wind Turbine Asset 2","assetModelId":"ec1d924f-f07d-444f-b072-e2994c165d35","assetModelExternalId":null,"assetModelName":"Wind Turbine Asset Model","assetPropertyId":"95e63da7-d34e-43e1-bc6f-1b490154b07a","assetPropertyExternalId":null,"assetPropertyName":"Temperature","assetPropertyExternalId":null,"assetPropertyExternalName":"Washington/Seattle/WT2/temp","timeSeriesId":"7020c8e2-e6db-40fa-9845-ed0dddd4c77d_95e63da7-d34e-43e1-bc6f-1b490154b07a","timeSeriesBucket":"f6","assetArn":null,"assetCompositeModelDescription":null}
{"assetId":"7020c8e2-e6db-40fa-9845-ed0dddd4c77d","assetExternalId":null,"assetName":"Wind Turbine Asset 2","assetModelId":"ec1d924f-f07d-444f-b072-e2994c165d35","assetModelExternalId":null,"assetModelName":"Wind Turbine Asset Model","assetPropertyId":"c706d54d-4c11-42dc-9a01-63662fc697b4","assetPropertyExternalId":null,"assetPropertyExternalName":"Washington/Seattle/WT2/pressure","timeSeriesId":"7020c8e2-e6db-40fa-9845-ed0dddd4c77d_c706d54d-4c11-42dc-9a01-63662fc697b4","timeSeriesBucket":"1e","assetArn":null,"assetCompositeModelDescription":null}
{"assetId":"7020c8e2-e6db-40fa-9845-ed0dddd4c77d","assetExternalId":null,"assetName":"Wind Turbine Asset 2","assetModelId":"ec1d924f-f07d-444f-b072-e2994c165d35","assetModelExternalId":null,"assetModelName":"Wind Turbine Asset Model","assetPropertyId":"95e63da7-d34e-43e1-bc6f-1b490154b07a","assetPropertyExternalId":null,"assetPropertyName":"Temperature","assetPropertyExternalId":null,"assetPropertyExternalName":"Washington/Seattle/WT2/temp","timeSeriesId":"7020c8e2-e6db-40fa-9845-ed0dddd4c77d_95e63da7-d34e-43e1-bc6f-1b490154b07a","timeSeriesBucket":"f6","assetArn":null,"assetCompositeModelDescription":null}

```

```

2", "assetModelId": "ec1d924f-f07d-444f-b072-
e2994c165d35", "assetModelExternalId": null, "assetModelName": "Wind
Turbine Asset Model", "assetPropertyId": "8cf1162f-dead-4fbe-b468-
c8e24cde9f50", "assetPropertyExternalId": null, "assetPropertyName": "Max
Temperature", "assetPropertyDataType": "DOUBLE", "assetPropertyUnit": null, "assetPropertyAlias": nu
e6db-40fa-9845-ed0ddddd4c77d_8cf1162f-dead-4fbe-b468-
c8e24cde9f50", "timeSeriesBucket": "d7", "assetArn": null, "assetCompositeModelDescription": null, "as

{"assetId": "3a5f2a22-3b37-4332-9c1c-404ea1d73fab", "assetExternalId": null, "assetName": "BatchAss
ebc75e75e827", "assetModelExternalId": null, "assetModelName": "FlashTestAssetModelDouble", "assetPr
b410-
ab401a9176ed", "assetPropertyExternalId": null, "assetPropertyName": "measurementProperty", "assetPr
ae89-
ff316f5ff8aa", "timeSeriesBucket": "af", "assetArn": null, "assetCompositeModelDescription": null, "as

```

Metadata hierarki aset

Saat Anda mengaktifkan AWS IoT SiteWise untuk menyimpan data dalam tingkat dingin untuk pertama kalinya, metadata hierarki aset diekspor ke tingkat dingin. Setelah konfigurasi awal, AWS IoT SiteWise mengekspor metadata hierarki aset ke tingkat dingin hanya jika Anda membuat perubahan pada model aset atau definisi aset. Metadata hierarki aset disimpan dalam tingkat dingin dalam format JSON () yang dibatasi baris baru. .ndjson

Pengidentifikasi eksternal untuk hierarki, aset target, atau aset sumber diambil dengan memanggil API. [DescribeAsset](#)

Jalur berkas

AWS IoT SiteWise menyimpan metadata hierarki aset di tingkat dingin menggunakan templat berikut.

```
{keyPrefix}/asset_hierarchy_metadata/{parentAssetId}_{hierarchyId}.ndjson
```

Setiap jalur file ke metadata hierarki aset di tingkat dingin berisi komponen berikut.

Komponen jalur	Deskripsi
keyPrefix	Awalan Amazon S3 yang Anda tentukan dalam konfigurasi penyimpanan. AWS IoT SiteWise

Komponen jalur	Deskripsi
	Amazon S3 menggunakan awalan sebagai nama folder di bucket.
<code>asset_hierarchy_metadata</code>	Folder yang menyimpan metadata hierarki aset. <code>asset_hierarchy_metadata</code> Folder disimpan di folder awalan.
<code>fileName</code>	<p>Nama file menggunakan karakter garis bawah (<code>_</code>) sebagai pembatas untuk memisahkan yang berikut:</p> <ul style="list-style-type: none"> <code>parentAssetId</code> Nilainya. <code>hierarchyId</code> Nilainya. <p>File disimpan dalam <code>.ndjson</code> format.</p>

Example jalur file ke metadata hierarki aset di tingkat dingin

```
keyPrefix/asset_hierarchy_metadata/35901915-d476-4dca-8637-d9ed4df939ed_c5b3ced8-589a-48c7-9998-cdcccfc9747a0.ndjson
```

Bidang

Skema metadata hierarki aset yang diekspor ke tingkat dingin berisi bidang berikut.

Nama bidang	Deskripsi
<code>sourceAssetId</code>	ID aset sumber dalam hubungan aset ini.
<code>targetAssetId</code>	ID aset target dalam hubungan aset ini.
<code>hierarchyId</code>	ID hierarki.
<code>associationType</code>	<p>Jenis asosiasi dari hubungan aset ini.</p> <p>Nilainya harusCHILD. Aset target adalah aset anak dari aset sumber.</p>

Example metadata hierarki aset di tingkat dingin

```
{"sourceAssetId":"80388e72-2284-44fb-9c89-bfbaf0dfedd2","targetAssetId":"2b866c25-0c74-4750-bdf5-b73683c8a2a2","hierarchyId":"bbed9f59-0412-4585-a61d-6044db526aee","associationType":"CHILD"}
{"sourceAssetId":"80388e72-2284-44fb-9c89-bfbaf0dfedd2","targetAssetId":"6b51246e-984d-460d-bc0b-470ea47d1e31","hierarchyId":"bbed9f59-0412-4585-a61d-6044db526aee","associationType":"CHILD"}
```

Untuk melihat data Anda di tingkat dingin

1. Arahkan ke [konsol Amazon S3](#).
2. Di panel navigasi, pilih Bucket, lalu pilih bucket Amazon S3 Anda.
3. Arahkan ke folder yang berisi data mentah, metadata aset, atau metadata hierarki aset.
4. Pilih file, dan kemudian dari Tindakan, pilih Unduh.

File indeks data penyimpanan

AWS IoT SiteWise menggunakan file-file ini untuk mengoptimalkan kinerja kueri data. Mereka muncul di ember Amazon S3 Anda, tetapi Anda tidak perlu menggunakannya.

Jalur berkas

AWS IoT SiteWise menyimpan file indeks data di tingkat dingin menggunakan template berikut.

```
keyPrefix/index/series=timeseriesId/startYear=startYear/startMonth=startMonth/  
startDay=startDay/index_timeseriesId_startTimestamp_quality
```

Example jalur file ke file indeks penyimpanan data

```
keyPrefix/index/series=7020c8e2-e6db-40fa-9845-ed0dddd4c77d_95e63da7-d34e-43e1-bc6f-1b490154b07a/startYear=2022/startMonth=02/startDay=03/index_7020c8e2-e6db-40fa-9845-ed0dddd4c77d_95e63da7-d34e-43e1-bc6f-1b490154b07a_1643846400_G00D
```

Keamanan di AWS IoT SiteWise

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. [Model tanggung jawab bersama](#) menjelaskan hal ini sebagai keamanan dari cloud dan keamanan dalam cloud:

- Keamanan cloud — AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan AWS layanan di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari [program AWS kepatuhan program AWS](#) . Untuk mempelajari tentang program kepatuhan yang berlaku AWS IoT SiteWise, lihat [AWS layanan dalam cakupan berdasarkan AWS layanan program kepatuhan](#) .
- Keamanan di cloud — Tanggung jawab Anda ditentukan oleh AWS layanan yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain, yang mencakup sensitivitas data Anda, persyaratan perusahaan Anda, serta undang-undang dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan AWS IoT SiteWise. Topik berikut menunjukkan cara mengonfigurasi AWS IoT SiteWise untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga belajar cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan AWS IoT SiteWise sumber daya Anda.

Topik

- [Perlindungan data di AWS IoT SiteWise](#)
- [Enkripsi data](#)
- [Identitas dan manajemen akses untuk AWS IoT SiteWise](#)
- [Validasi kepatuhan untuk AWS IoT SiteWise](#)
- [Ketahanan di AWS IoT SiteWise](#)
- [Keamanan infrastruktur di AWS IoT SiteWise](#)
- [Konfigurasi dan analisis kerentanan](#)
- [Titik akhir VPC](#)

- [Praktik terbaik keamanan untuk AWS IoT SiteWise](#)

Perlindungan data di AWS IoT SiteWise

[Model tanggung jawab AWS bersama model](#) berlaku untuk perlindungan data di AWS IoT SiteWise. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam [Pertanyaan Umum Privasi Data](#). Lihat informasi tentang perlindungan data di Eropa di pos blog [Model Tanggung Jawab Bersama dan GDPR AWS](#) di Blog Keamanan AWS .

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredensial dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan sumber daya. AWS Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-2 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi yang lebih lengkap tentang titik akhir FIPS yang tersedia di [Standar Pemrosesan Informasi Federal \(FIPS\) 140-2](#).

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan AWS IoT SiteWise atau lainnya Layanan AWS menggunakan konsol, API AWS CLI, atau AWS SDK. Data apa pun yang Anda masukkan ke dalam

tanda atau bidang isian bebas yang digunakan untuk nama dapat digunakan untuk log penagihan atau log diagnostik. Saat Anda memberikan URL ke server eksternal, kami sangat menganjurkan supaya Anda tidak menyertakan informasi kredensial di dalam URL untuk memvalidasi permintaan Anda ke server itu.

Topik

- [Privasi lalu lintas jaringan internet](#)

Privasi lalu lintas jaringan internet

Koneksi antara AWS IoT SiteWise dan aplikasi lokal, seperti gateway SiteWise Edge, diamankan melalui koneksi Transport Layer Security (TLS). Untuk informasi selengkapnya, lihat [Enkripsi dalam bergerak](#).

AWS IoT SiteWise tidak mendukung koneksi antara Availability Zone dalam AWS Wilayah atau koneksi antar AWS akun.

Anda dapat mengonfigurasi Pusat Identitas IAM hanya dalam satu Wilayah pada satu waktu. SiteWise Monitor terhubung ke Wilayah yang Anda konfigurasi untuk Pusat Identitas IAM. Ini berarti Anda menggunakan satu Wilayah untuk akses Pusat Identitas IAM, tetapi Anda dapat membuat portal di Wilayah mana pun.

Enkripsi data

Enkripsi data mengacu pada perlindungan data saat dalam perjalanan (saat bepergian ke dan dari AWS IoT SiteWise, dan antara gateway dan server SiteWise Edge), dan saat istirahat (saat disimpan di perangkat lokal atau di AWS layanan). Anda dapat melindungi data saat transit menggunakan Keamanan Lapisan Pengangkutan (TLS) atau saat istirahat menggunakan enkripsi di sisi klien.

Note

AWS IoT SiteWise pemrosesan tepi mengekspos API yang di-host dalam gateway SiteWise Edge dan dapat diakses melalui jaringan lokal. API ini diekspos melalui koneksi TLS yang didukung oleh server-sertifikat yang dimiliki oleh konektor Edge. AWS IoT SiteWise Untuk otentikasi klien, API ini menggunakan kata sandi kontrol akses. Server-certificate private-key dan password akses-kontrol keduanya disimpan pada disk. AWS IoT SiteWise pemrosesan tepi bergantung pada enkripsi sistem file untuk keamanan kredensial ini saat istirahat.

Untuk informasi lebih lanjut tentang enkripsi sisi server dan enkripsi sisi klien, tinjau topik yang tercantum di bawah ini.

Topik

- [Enkripsi diam](#)
- [Enkripsi dalam bergerak](#)
- [Manajemen kunci](#)

Enkripsi diam

AWS IoT SiteWise menyimpan data Anda di AWS Cloud dan di gateway AWS IoT SiteWise Edge.

Data saat istirahat di AWS Cloud

AWS IoT SiteWise menyimpan data di AWS layanan lain yang mengenkripsi data saat istirahat secara default. Enkripsi saat istirahat terintegrasi dengan AWS Key Management Service (AWS KMS) untuk mengelola kunci enkripsi yang digunakan untuk mengenkripsi nilai properti aset Anda dan nilai agregat di. AWS IoT SiteWise Anda dapat memilih untuk menggunakan kunci terkelola pelanggan untuk mengenkripsi nilai properti aset dan nilai agregat. AWS IoT SiteWise Anda dapat membuat, mengelola, dan melihat kunci enkripsi Anda melalui AWS KMS.

Anda dapat memilih Kunci milik AWS untuk mengenkripsi data Anda, atau memilih kunci yang dikelola pelanggan untuk mengenkripsi nilai properti aset dan nilai agregat Anda:

Cara kerjanya

Enkripsi saat istirahat terintegrasi dengan AWS KMS untuk mengelola kunci enkripsi yang digunakan untuk mengenkripsi data Anda.

- Kunci milik AWS — Kunci enkripsi default. AWS IoT SiteWise memiliki kunci ini. Anda tidak dapat melihat kunci ini di AWS akun Anda. Anda juga tidak dapat melihat operasi pada kunci di AWS CloudTrail log. Anda dapat menggunakan kunci ini tanpa biaya tambahan.
- Kunci yang dikelola pelanggan — Kunci disimpan di akun Anda, yang Anda buat, miliki, dan kelola. Anda memiliki kontrol penuh atas tombol KMS. AWS KMS Biaya tambahan berlaku.

Kunci milik AWS

Kunci milik AWS tidak disimpan di akun Anda. Mereka adalah bagian dari kumpulan kunci KMS yang AWS memiliki dan mengelola untuk digunakan di beberapa AWS akun. AWS Layanan dapat digunakan Kunci milik AWS untuk melindungi data Anda.

Anda tidak dapat melihat, mengelola, menggunakan Kunci milik AWS, atau mengaudit penggunaannya. Namun, Anda tidak perlu melakukan pekerjaan apa pun atau mengubah program apa pun untuk melindungi kunci yang mengenkripsi data Anda.

Anda tidak dikenakan biaya bulanan atau biaya penggunaan jika Anda menggunakan Kunci milik AWS, dan mereka tidak dihitung terhadap AWS KMS kuota untuk akun Anda.

Kunci yang dikelola pelanggan

Kunci yang dikelola pelanggan adalah kunci KMS di akun Anda yang Anda buat, miliki, dan kelola. Anda memiliki kontrol penuh atas kunci KMS ini, seperti berikut ini:

- Menetapkan dan memelihara kebijakan utama mereka, kebijakan IAM, dan hibah
- Mengaktifkan dan menonaktifkannya
- Memutar materi kriptografi mereka
- Menambahkan tanda
- Membuat alias yang merujuk kepada mereka
- Menjadwalkan mereka untuk dihapus

Anda juga dapat menggunakan CloudTrail dan Amazon CloudWatch Logs untuk melacak permintaan yang AWS IoT SiteWise dikirimkan AWS KMS atas nama Anda.

Jika Anda menggunakan kunci yang dikelola pelanggan, Anda harus memberikan AWS IoT SiteWise akses ke kunci KMS yang disimpan di akun Anda. AWS IoT SiteWise menggunakan enkripsi amplop dan hierarki kunci untuk mengenkripsi data. Kunci AWS KMS enkripsi Anda digunakan untuk mengenkripsi kunci root hierarki kunci ini. Untuk informasi lebih lanjut, lihat [Enkripsi amplop](#) di Panduan Developer AWS Key Management Service .

Contoh kebijakan berikut memberikan AWS IoT SiteWise izin untuk membuat kunci terkelola pelanggan atas nama Anda. Ketika Anda membuat kunci Anda, Anda harus mengizinkan `kms:CreateGrant` dan `kms:DescribeKey` tindakan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1603902045292",
      "Action": [
        "kms:CreateGrant",
        "kms:DescribeKey"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Konteks enkripsi untuk hibah yang Anda buat menggunakan ID Anda `aws:iotsitewise:subscriberId` dan akun.

Data diam di gateway SiteWise Edge

AWS IoT SiteWise gateway menyimpan data berikut pada sistem file lokal:

- Informasi konfigurasi sumber OPC-UA
- Kumpulan jalur aliran data OPC-UA dari sumber OPC-UA yang terhubung
- Data industri di-cache saat gateway SiteWise Edge kehilangan koneksi ke internet

SiteWise Gerbang tepi berjalan. AWS IoT Greengrass AWS IoT Greengrass bergantung pada izin file Unix dan enkripsi full-disk (jika diaktifkan) untuk melindungi data pada intinya. Anda bertanggung jawab untuk mengamankan sistem file dan perangkat.

Namun, AWS IoT Greengrass mengenkripsi salinan lokal rahasia server OPC-UA Anda yang diambil dari Secrets Manager. Untuk informasi selengkapnya, lihat [Enkripsi rahasia](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

Untuk informasi selengkapnya tentang enkripsi saat istirahat pada AWS IoT Greengrass inti, lihat [Enkripsi saat istirahat](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

Enkripsi dalam bergerak

AWS IoT SiteWise memiliki tiga mode komunikasi di mana data dalam perjalanan:

- [Melalui internet](#) — Komunikasi antara perangkat lokal (termasuk gateway SiteWise Edge) dan AWS IoT SiteWise dienkripsi.
- [Melalui jaringan lokal](#) — Komunikasi antara OpsHub untuk SiteWise aplikasi dan gateway SiteWise Edge selalu dienkripsi. Komunikasi antara aplikasi SiteWise monitor yang berjalan di dalam browser Anda dan gateway SiteWise Edge selalu dienkripsi. Komunikasi antara gateway SiteWise Edge dan sumber OPC-UA dapat dienkripsi.
- [Antara komponen di gateway SiteWise Edge](#) — Komunikasi antar AWS IoT Greengrass komponen di gateway SiteWise Edge tidak dienkripsi.

Topik

- [Data dalam transit melalui internet](#)
- [Data dalam transit melalui jaringan lokal](#)
- [Data dalam transit antar komponen lokal di gateway SiteWise Edge](#)

Data dalam transit melalui internet

AWS IoT SiteWise Menggunakan Transport Layer Security (TLS) untuk mengenkripsi semua komunikasi melalui internet. Semua data yang dikirim ke AWS Cloud dikirim melalui koneksi TLS menggunakan protokol MQTT atau HTTPS, sehingga aman secara default. SiteWise Edge gateway, yang berjalan AWS IoT Greengrass, dan pemberitahuan nilai properti menggunakan model keamanan AWS IoT transportasi. Untuk informasi lebih lanjut, lihat [Keamanan angkutan](#) di dalam AWS IoT Panduan Developer.

Data dalam transit melalui jaringan lokal

SiteWise Edge gateway mengikuti spesifikasi OPC-UA untuk komunikasi dengan sumber OPC-UA lokal. Anda bertanggung jawab untuk mengonfigurasi sumber Anda agar menggunakan mode keamanan pesan yang mengenkripsi data dalam perjalanan.

Jika Anda memilih mode keamanan pesan tanda tangan, data yang sedang transit antara gateway SiteWise Edge dan sumber ditandatangani tetapi tidak dienkripsi. Jika Anda memilih mode keamanan pesan tanda dan enkripsi, data yang sedang transit antara gateway dan sumber SiteWise Edge ditandatangani dan dienkripsi. Untuk informasi selengkapnya tentang mengonfigurasi sumber, lihat [Mengkonfigurasi sumber data](#).

Komunikasi antara aplikasi konsol tepi dan gateway SiteWise Edge selalu dienkripsi oleh TLS. Konektor SiteWise SiteWise Edge pada gateway Edge menghasilkan dan menyimpan sertifikat

yang ditandatangani sendiri untuk dapat membuat koneksi TLS dengan konsol tepi untuk AWS IoT SiteWise aplikasi. Anda harus menyalin sertifikat ini dari gateway SiteWise Edge Anda ke konsol tepi untuk AWS IoT SiteWise aplikasi sebelum Anda menghubungkan aplikasi ke gateway SiteWise Edge. Ini memastikan bahwa konsol tepi untuk AWS IoT SiteWise aplikasi dapat memverifikasi bahwa ia telah terhubung ke gateway SiteWise Edge tepercaya Anda.

Selain TLS untuk kerahasiaan dan keaslian server, SiteWise Edge menggunakan protokol SiGv4 untuk menetapkan keaslian aplikasi konsol tepi. Konektor SiteWise Edge pada gateway SiteWise Edge menerima dan menyimpan kata sandi untuk dapat memverifikasi koneksi masuk dari aplikasi konsol tepi, aplikasi SiteWise Monitor yang berjalan di dalam browser, dan klien lain berdasarkan SDK. AWS IoT SiteWise

Untuk informasi selengkapnya tentang membuat kata sandi dan sertifikat server, lihat [the section called “Mengelola SiteWise gateway Edge”](#).

Data dalam transit antar komponen lokal di gateway SiteWise Edge

SiteWise Edge gateway berjalan AWS IoT Greengrass, yang tidak mengenkripsi data yang dipertukarkan secara lokal pada AWS IoT Greengrass inti karena data tidak meninggalkan perangkat. Ini termasuk komunikasi antar AWS IoT Greengrass komponen seperti AWS IoT SiteWise konektor. Untuk informasi selengkapnya, lihat [Data pada perangkat inti](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

Manajemen kunci

AWS IoT SiteWise manajemen kunci cloud

Secara default, AWS IoT SiteWise gunakan Kunci yang dikelola AWS untuk melindungi data Anda di AWS Cloud. Anda dapat memperbarui pengaturan Anda untuk menggunakan kunci yang dikelola pelanggan untuk mengenkripsi beberapa data. AWS IoT SiteWise Anda dapat membuat, mengelola, dan melihat kunci enkripsi Anda melalui AWS Key Management Service (AWS KMS).

AWS IoT SiteWise mendukung enkripsi sisi server dengan kunci terkelola pelanggan yang disimpan AWS KMS untuk mengenkripsi data berikut:

- Nilai properti aset
- Nilai agregat

Note

Data dan sumber daya lainnya dienkripsi menggunakan enkripsi default dengan kunci yang dikelola oleh AWS IoT SiteWise Kunci ini disimpan di AWS IoT SiteWise akun.

Untuk informasi lebih lanjut, lihat [Apa itu AWS Key Management Service?](#) di Panduan AWS Key Management Service Pengembang.

Aktifkan enkripsi menggunakan kunci yang dikelola pelanggan

Untuk menggunakan kunci yang dikelola pelanggan AWS IoT SiteWise, Anda perlu memperbarui AWS IoT SiteWise pengaturan Anda.

Untuk mengaktifkan enkripsi menggunakan kunci KMS

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Pilih Pengaturan Akun dan pilih Edit untuk membuka halaman Edit pengaturan akun.
3. Untuk jenis kunci Enkripsi, pilih Pilih AWS KMS kunci yang berbeda. Ini memungkinkan enkripsi dengan kunci terkelola pelanggan yang disimpan di AWS KMS.

Note

Saat ini, Anda hanya dapat menggunakan enkripsi kunci terkelola pelanggan untuk nilai properti aset dan nilai agregat.

4. Pilih kunci KMS Anda dengan salah satu opsi berikut:
 - Untuk menggunakan kunci KMS yang ada — Pilih alias kunci KMS Anda dari daftar.
 - Untuk membuat kunci KMS baru — Pilih Buat AWS KMS kunci.

Note

Ini membuka AWS KMS dasbor. Untuk informasi selengkapnya tentang membuat kunci KMS, lihat [Membuat kunci](#) di Panduan AWS Key Management Service Pengembang.

5. Pilih Simpan untuk memperbarui pengaturan Anda.

SiteWise Manajemen kunci gateway tepi

SiteWise Edge gateway berjalan AWS IoT Greengrass, dan perangkat AWS IoT Greengrass inti menggunakan kunci publik dan pribadi untuk mengautentikasi dengan AWS Cloud dan mengenkripsi rahasia lokal, seperti rahasia otentikasi OPC-UA. Untuk informasi selengkapnya, lihat [Manajemen kunci](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

Identitas dan manajemen akses untuk AWS IoT SiteWise

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya. AWS IoT SiteWise IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

- [Audiens](#)
- [Mengautentikasi dengan identitas](#)
- [Bagaimana AWS IoT SiteWise bekerja dengan IAM](#)
- [AWS kebijakan terkelola untuk AWS IoT SiteWise](#)
- [Menggunakan peran tertaut layanan untuk AWS IoT SiteWise](#)
- [Menyiapkan izin untuk alarm AWS IoT Events](#)
- [Pencegahan confused deputy lintas layanan](#)
- [Memecahkan masalah AWS IoT SiteWise identitas dan akses](#)

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan. AWS IoT SiteWise

Pengguna layanan — Jika Anda menggunakan AWS IoT SiteWise layanan untuk melakukan pekerjaan Anda, maka administrator Anda memberi Anda kredensial dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak AWS IoT SiteWise fitur untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara mengelola akses dapat membantu Anda meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur di AWS IoT SiteWise, lihat [Memecahkan masalah AWS IoT SiteWise identitas dan akses](#).

Administrator layanan — Jika Anda bertanggung jawab atas AWS IoT SiteWise sumber daya di perusahaan Anda, Anda mungkin memiliki akses penuh ke AWS IoT SiteWise. Tugas Anda adalah menentukan AWS IoT SiteWise fitur dan sumber daya mana yang harus diakses pengguna layanan Anda. Kemudian, Anda harus mengirimkan permintaan kepada administrator IAM Anda untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep Basic IAM. Untuk mempelajari lebih lanjut tentang bagaimana perusahaan Anda dapat menggunakan IAM AWS IoT SiteWise, lihat [Bagaimana AWS IoT SiteWise bekerja dengan IAM](#).

Administrator IAM – Jika Anda adalah administrator IAM, Anda mungkin ingin belajar dengan lebih detail tentang cara Anda menulis kebijakan untuk mengelola akses ke AWS IoT SiteWise. Untuk melihat contoh kebijakan AWS IoT SiteWise berbasis identitas yang dapat Anda gunakan di IAM, lihat [AWS IoT SiteWise contoh kebijakan berbasis identitas](#)

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi (masuk ke AWS) sebagai Pengguna root akun AWS, sebagai pengguna IAM, atau dengan mengasumsikan peran IAM.

Anda dapat masuk AWS sebagai identitas federasi dengan menggunakan kredensial yang disediakan melalui sumber identitas. AWS IAM Identity Center Pengguna (IAM Identity Center), autentikasi masuk tunggal perusahaan Anda, dan kredensial Google atau Facebook Anda adalah contoh identitas federasi. Saat Anda masuk sebagai identitas gabungan, administrator Anda sebelumnya menyiapkan federasi identitas menggunakan peran IAM. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil peran.

Bergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal AWS akses. Untuk informasi selengkapnya tentang masuk AWS, lihat [Cara masuk ke Panduan AWS Sign-In Pengguna Anda Akun AWS](#).

Jika Anda mengakses AWS secara terprogram, AWS sediakan kit pengembangan perangkat lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensial Anda. Jika Anda tidak menggunakan AWS alat, Anda harus menandatangani permintaan sendiri. Untuk informasi selengkapnya tentang penggunaan metode yang disarankan untuk menandatangani permintaan sendiri, lihat [Menandatangani permintaan AWS API](#) di Panduan Pengguna IAM.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Misalnya, AWS merekomendasikan agar Anda menggunakan otentikasi multi-

faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari selengkapnya, lihat [Autentikasi multi-faktor](#) dalam Panduan Pengguna AWS IAM Identity Center dan [Menggunakan autentikasi multi-faktor \(MFA\) di AWS](#) dalam Panduan Pengguna IAM.

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya di akun. Identitas ini disebut pengguna Akun AWS root dan diakses dengan masuk dengan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari Anda. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, lihat [Tugas yang memerlukan kredensial pengguna root](#) dalam Panduan Pengguna IAM.

Pengguna dan grup IAM

[Pengguna IAM](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, sebaiknya andalkan kredensial sementara daripada membuat pengguna IAM yang memiliki kredensial jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan khusus yang memerlukan kredensial jangka panjang dengan pengguna IAM, kami sarankan Anda merotasi kunci akses. Untuk informasi selengkapnya, lihat [Rotasikan kunci akses secara rutin untuk kasus penggunaan yang memerlukan kredensial jangka panjang](#) dalam Panduan Pengguna IAM.

[Grup IAM](#) adalah identitas yang menentukan kumpulan pengguna IAM. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin untuk beberapa pengguna sekaligus. Grup membuat izin lebih mudah dikelola untuk sekelompok besar pengguna. Misalnya, Anda dapat memiliki grup yang bernama IAMAdmins dan memberikan izin kepada grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran dimaksudkan untuk dapat digunakan oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari selengkapnya, silakan lihat [Kapan harus membuat pengguna IAM \(bukan peran\)](#) dalam Panduan Pengguna IAM.

Peran IAM

[Peran IAM](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus. Peran ini mirip dengan pengguna IAM, tetapi tidak terkait dengan orang tertentu. Anda dapat mengambil peran IAM untuk sementara AWS Management Console dengan [beralih peran](#). Anda dapat mengambil peran dengan memanggil operasi AWS CLI atau AWS API atau dengan menggunakan URL kustom. Untuk informasi selengkapnya tentang cara menggunakan peran, lihat [Menggunakan peran IAM](#) dalam Panduan Pengguna IAM.

Peran IAM dengan kredensial sementara berguna dalam situasi berikut:

- Akses pengguna gabungan – Untuk menetapkan izin ke sebuah identitas gabungan, Anda dapat membuat peran dan menentukan izin untuk peran tersebut. Saat identitas terfederasi mengautentikasi, identitas tersebut akan dikaitkan dengan peran dan diberi izin yang ditentukan oleh peran tersebut. Untuk informasi tentang peran-peran untuk federasi, lihat [Membuat peran untuk Penyedia Identitas pihak ketiga](#) dalam Panduan Pengguna IAM. Jika Anda menggunakan Pusat Identitas IAM, Anda perlu mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah identitas tersebut diautentikasi, Pusat Identitas IAM mengorelasikan izin yang diatur ke peran dalam IAM. Untuk informasi tentang rangkaian izin, lihat [Rangkaian izin](#) dalam Panduan Pengguna AWS IAM Identity Center .
- Izin pengguna IAM sementara – Pengguna atau peran IAM dapat mengambil peran IAM guna mendapatkan berbagai izin secara sementara untuk tugas tertentu.
- Akses lintas akun – Anda dapat menggunakan peran IAM untuk mengizinkan seseorang (pengguna utama tepercaya) dengan akun berbeda untuk mengakses sumber daya yang ada di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, dengan beberapa Layanan AWS, Anda dapat melampirkan kebijakan langsung ke sumber daya (alih-alih menggunakan peran sebagai proxy). Untuk mempelajari perbedaan antara kebijakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat [Bagaimana peran IAM berbeda dari kebijakan berbasis sumber daya](#) dalam Panduan Pengguna IAM.
- Akses lintas layanan — Beberapa Layanan AWS menggunakan fitur lain Layanan AWS. Sebagai contoh, ketika Anda melakukan panggilan dalam suatu layanan, biasanya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Sebuah layanan mungkin melakukannya menggunakan izin prinsipal yang memanggil, menggunakan peran layanan, atau peran terkait layanan.
 - Sesi akses teruskan (FAS) — Saat Anda menggunakan pengguna atau peran IAM untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Saat Anda menggunakan beberapa layanan, Anda mungkin melakukan tindakan yang kemudian memulai tindakan lain di layanan

yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan saat membuat permintaan FAS, lihat [Teruskan sesi akses](#).

- Peran layanan – Peran layanan adalah [peran IAM](#) yang diambil oleh layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat [Membuat peran untuk mendelegasikan izin ke Layanan AWS](#) dalam Panduan pengguna IAM.
- Peran terkait layanan — Peran terkait layanan adalah jenis peran layanan yang ditautkan ke Layanan AWS. Layanan dapat menggunakan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.
- Aplikasi yang berjalan di Amazon EC2 — Anda dapat menggunakan peran IAM untuk mengelola kredensial sementara untuk aplikasi yang berjalan pada instans EC2 dan membuat atau permintaan API. AWS CLI AWS Cara ini lebih dianjurkan daripada menyimpan kunci akses dalam instans EC2. Untuk menetapkan AWS peran ke instans EC2 dan membuatnya tersedia untuk semua aplikasinya, Anda membuat profil instance yang dilampirkan ke instance. Profil instans berisi peran dan memungkinkan program yang berjalan di instans EC2 mendapatkan kredensial sementara. Untuk informasi selengkapnya, lihat [Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di instans Amazon EC2](#) dalam Panduan Pengguna IAM.

Untuk mempelajari apakah kita harus menggunakan peran IAM atau pengguna IAM, lihat [Kapan harus membuat peran IAM \(bukan pengguna\)](#) dalam Panduan Pengguna IAM.

Bagaimana AWS IoT SiteWise bekerja dengan IAM

Sebelum Anda menggunakan AWS Identity and Access Management (IAM) untuk mengelola akses AWS IoT SiteWise, Anda harus memahami fitur IAM apa yang tersedia untuk digunakan. AWS IoT SiteWise

Fitur IAM	Didukung oleh AWS IoT SiteWise
Kebijakan berbasis identitas dengan izin tingkat sumber daya	Ya
Tindakan kebijakan	Ya
Sumber daya kebijakan	Ya
Kunci persyaratan kebijakan	Ya
Kebijakan berbasis sumber daya	Tidak
Daftar kontrol akses (ACL)	Tidak
Otorisasi berbasis tag (ABAC)	Ya
Kredensial sementara	Ya
Sesi akses teruskan (FAS)	Ya
Peran terkait layanan	Ya
Peran layanan	Ya

Untuk mendapatkan pandangan tingkat tinggi tentang bagaimana AWS IoT SiteWise dan AWS layanan lain bekerja dengan IAM, lihat [AWS layanan yang bekerja dengan IAM di Panduan Pengguna IAM](#).

Daftar Isi

- [AWS IoT SiteWise Peran IAM](#)
 - [Menggunakan kredensial sementara dengan AWS IoT SiteWise](#)
 - [Sesi akses teruskan \(FAS\) untuk AWS IoT SiteWise](#)
 - [Peran terkait layanan](#)
 - [Peran layanan](#)

- [Memilih IAM role dalam AWS IoT SiteWise](#)
- [Otorisasi berdasarkan tanda AWS IoT SiteWise](#)
- [AWS IoT SiteWise kebijakan berbasis identitas](#)
 - [Tindakan kebijakan](#)
 - [BatchPutAssetPropertyValue otorisasi](#)
 - [Sumber daya kebijakan](#)
 - [Kunci persyaratan kebijakan](#)
 - [Contoh-contoh](#)
- [AWS IoT SiteWise contoh kebijakan berbasis identitas](#)
 - [Praktik terbaik kebijakan](#)
 - [Menggunakan konsol AWS IoT SiteWise](#)
 - [Memungkinkan pengguna untuk melihat izin mereka sendiri](#)
 - [Memungkinkan pengguna untuk menyerap data ke aset dalam satu hierarki](#)
 - [Melihat AWS IoT SiteWise aset berdasarkan tag](#)
- [Mengelola akses menggunakan kebijakan](#)
 - [Kebijakan berbasis identitas](#)
 - [Kebijakan berbasis sumber daya](#)
 - [Daftar kontrol akses \(ACL\)](#)
 - [Tipe kebijakan lain](#)
 - [Beberapa jenis kebijakan](#)

AWS IoT SiteWise Peran IAM

[IAM role](#) adalah entitas dalam Akun AWS Anda yang memiliki izin khusus.

Menggunakan kredensial sementara dengan AWS IoT SiteWise

Anda dapat menggunakan kredensial sementara untuk masuk dengan gabungan, menjalankan IAM role, atau menjalankan peran lintas akun. Anda memperoleh kredensial keamanan sementara dengan memanggil operasi AWS STS API seperti [AssumeRole](#) atau [GetFederationToken](#).

AWS IoT SiteWise mendukung menggunakan kredensial sementara.

SiteWise Monitor mendukung pengguna federasi untuk mengakses portal. Pengguna portal mengautentikasi dengan Pusat Identitas IAM atau kredensial IAM mereka.

⚠ Important

Pengguna atau peran harus memiliki `iotsitewise:DescribePortal` izin untuk masuk ke portal.

Saat pengguna masuk ke portal, SiteWise Monitor akan membuat kebijakan sesi yang memberikan izin berikut:

- Akses hanya-baca ke aset dan data aset AWS IoT SiteWise di akun Anda yang peran portal tersebut menyediakan akses.
- Akses ke proyek di portal tempat pengguna memiliki akses administrator (pemilik proyek) atau hanya-baca (penampil proyek).

Untuk informasi selengkapnya tentang izin pengguna portal federasi, lihat [Menggunakan peran layanan untuk AWS IoT SiteWise Monitor](#)

Sesi akses teruskan (FAS) untuk AWS IoT SiteWise

Mendukung sesi akses maju (FAS)

Ya

Saat Anda menggunakan pengguna atau peran IAM untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Saat Anda menggunakan beberapa layanan, Anda mungkin melakukan tindakan yang kemudian memulai tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan saat membuat permintaan FAS, lihat [Teruskan sesi akses](#).

Peran terkait layanan

[Peran terkait AWS layanan](#) memungkinkan layanan mengakses sumber daya di layanan lain untuk menyelesaikan tindakan atas nama Anda. Peran terkait layanan akan muncul di Akun AWS dan

dimiliki oleh layanan tersebut. Administrator IAM dapat melihat tetapi tidak dapat mengedit izin untuk peran terkait layanan.

AWS IoT SiteWise mendukung peran terkait layanan. Untuk informasi selengkapnya tentang cara membuat atau mengelola peran terkait layanan AWS IoT SiteWise, lihat [Menggunakan peran tertaut layanan untuk AWS IoT SiteWise](#).

Peran layanan

Fitur ini memungkinkan layanan untuk menerima [peran layanan](#) atas nama Anda. Peran ini mengizinkan layanan untuk mengakses sumber daya di layanan lain untuk menyelesaikan tindakan atas nama Anda. Peran layanan muncul di Anda Akun AWS dan dimiliki oleh akun. Ini berarti administrator IAM dapat mengubah izin untuk peran ini. Namun, melakukan hal itu dapat merusak fungsionalitas layanan.

AWS IoT SiteWise menggunakan peran layanan untuk memungkinkan pengguna portal SiteWise Monitor mengakses beberapa sumber AWS IoT SiteWise daya Anda atas nama Anda. Untuk informasi selengkapnya, lihat [Menggunakan peran layanan untuk AWS IoT SiteWise Monitor](#).

Anda harus memiliki izin yang diperlukan sebelum Anda dapat membuat model AWS IoT Events alarm di AWS IoT SiteWise. Untuk informasi selengkapnya, lihat [Menyiapkan izin untuk alarm AWS IoT Events](#).

Memilih IAM role dalam AWS IoT SiteWise

Saat Anda membuat portal sumber daya AWS IoT SiteWise, Anda harus memilih peran untuk memungkinkan pengguna federasi portal SiteWise Monitor Anda mengakses AWS IoT SiteWise atas nama Anda. Jika sebelumnya Anda telah membuat peran layanan, AWS IoT SiteWise berikan daftar peran untuk dipilih. Jika tidak, Anda dapat membuat peran dengan izin yang diperlukan saat membuat portal. Penting untuk memilih peran yang memungkinkan akses ke aset dan data aset Anda. Untuk informasi selengkapnya, lihat [Menggunakan peran layanan untuk AWS IoT SiteWise Monitor](#).

Otorisasi berdasarkan tanda AWS IoT SiteWise

Anda dapat melampirkan tag ke AWS IoT SiteWise sumber daya atau meneruskan tag dalam permintaan AWS IoT SiteWise. Untuk mengendalikan akses berdasarkan tag, berikan informasi tentang tag di [elemen kondisi](#) dari kebijakan dengan menggunakan kunci kondisi `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`. Untuk

informasi selengkapnya tentang penandaan sumber daya AWS IoT SiteWise, lihat [Menandai sumber daya Anda AWS IoT SiteWise](#).

Untuk melihat contoh kebijakan berbasis identitas untuk membatasi akses ke sumber daya berdasarkan tanda pada sumber daya tersebut, lihat [Melihat AWS IoT SiteWise aset berdasarkan tag](#).

AWS IoT SiteWise kebijakan berbasis identitas

Kebijakan IAM memungkinkan Anda mengontrol siapa yang dapat melakukan apa. AWS IoT SiteWise Anda dapat memutuskan tindakan apa yang diizinkan atau tidak dan menetapkan kondisi khusus untuk tindakan ini. Misalnya, Anda dapat membuat aturan tentang siapa yang dapat melihat atau mengubah informasi AWS IoT SiteWise. AWS IoT SiteWise mendukung tindakan, sumber daya, dan kunci kondisi tertentu. Untuk mempelajari semua elemen yang Anda gunakan dalam kebijakan JSON, lihat [Referensi elemen kebijakan IAM JSON](#) dalam Panduan Pengguna IAM.

Tindakan kebijakan

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Tindakan kebijakan biasanya memiliki nama yang sama dengan operasi AWS API terkait. Ada beberapa pengecualian, misalnya tindakan hanya izin yang tidak memiliki operasi API yang cocok. Ada juga beberapa operasi yang memerlukan beberapa tindakan dalam suatu kebijakan. Tindakan tambahan ini disebut tindakan dependen.

Menyertakan tindakan dalam suatu kebijakan untuk memberikan izin melakukan operasi terkait.

Tindakan kebijakan AWS IoT SiteWise menggunakan awalan berikut sebelum tindakan: `iotsitewise:`. Misalnya, untuk memberikan izin kepada seseorang untuk mengunggah data properti aset AWS IoT SiteWise dengan operasi `BatchPutAssetPropertyValue` API, Anda menyertakan `iotsitewise:BatchPutAssetPropertyValue` tindakan tersebut dalam kebijakan mereka. Pernyataan kebijakan harus mencakup salah satu `Action` atau `NotAction` elemen. AWS IoT SiteWise mendefinisikan serangkaian tindakannya sendiri yang menggambarkan tugas yang dapat Anda lakukan dengan layanan ini.

Untuk menentukan beberapa tindakan dalam satu pernyataan, pisahkan tindakan dengan koma seperti berikut:

```
"Action": [  
  "iotsitewise:action1",  
  "iotsitewise:action2"  
]
```

Anda juga dapat menentukan beberapa tindakan menggunakan wildcard (*). Misalnya, untuk menentukan semua tindakan yang dimulai dengan kata Describe, sertakan tindakan berikut.

```
"Action": "iotsitewise:Describe*"
```

Untuk melihat daftar tindakan, lihat AWS IoT SiteWise [Tindakan yang Ditentukan oleh AWS IoT SiteWise](#) dalam Panduan Pengguna IAM.

BatchPutAssetPropertyValue otorisasi

AWS IoT SiteWise mengizinkan akses ke [BatchPutAssetPropertyValuetindakan](#) dengan cara yang tidak biasa. Untuk sebagian besar tindakan, saat Anda mengizinkan atau menolak akses, tindakan tersebut mengembalikan kesalahan jika izin tidak diberikan.

Dengan `BatchPutAssetPropertyValu`, Anda dapat mengirim beberapa entri data ke aset dan properti aset yang berbeda dalam satu permintaan API. AWS IoT SiteWise mengotorisasi setiap entri data secara independen. Untuk setiap entri individu yang gagal otorisasi dalam permintaan, AWS IoT SiteWise termasuk `AccessDeniedException` dalam daftar kesalahan yang dikembalikan. AWS IoT SiteWise menerima data untuk setiap entri yang mengotorisasi dan berhasil, bahkan jika entri lain dalam permintaan yang sama gagal.

Important

Sebelum Anda menelan data ke aliran data, lakukan hal berikut:

- Otorisasi `time-series` sumber daya jika Anda menggunakan alias properti untuk mengidentifikasi aliran data.
- Otorisasi `asset` sumber daya jika Anda menggunakan ID aset untuk mengidentifikasi aset yang berisi properti aset terkait.

Sumber daya kebijakan

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek atau beberapa objek yang menjadi target penerapan tindakan. Pernyataan harus menyertakan elemen `Resource` atau `NotResource`. Praktik terbaiknya, tentukan sumber daya menggunakan [Amazon Resource Name \(ARN\)](#). Anda dapat melakukan ini untuk tindakan yang mendukung jenis sumber daya tertentu, yang dikenal sebagai izin tingkat sumber daya.

Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, misalnya operasi pencantuman, gunakan wildcard (*) untuk mengindikasikan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": "*"
```

Setiap pernyataan kebijakan IAM berlaku untuk sumber daya yang Anda tentukan menggunakan ARN. ARN memiliki sintaks umum berikut.

```
arn:${Partition}:${Service}:${Region}:${Account}:${ResourceType}/${ResourcePath}
```

Untuk informasi selengkapnya tentang format ARN, lihat [Amazon Resource Names \(ARN\) dan ruang nama AWS layanan](#).

Misalnya, untuk menentukan aset dengan ID `a1b2c3d4-5678-90ab-cdef-22222EXAMPLE` dalam pernyataan Anda, gunakan ARN berikut. ;

```
"Resource": "arn:aws:iotsitewise:region:123456789012:asset/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"
```

Untuk menentukan semua aliran data milik akun tertentu, gunakan wildcard (*):

```
"Resource": "arn:aws:iotsitewise:region:123456789012:time-series/*"
```

Untuk menentukan semua aset milik akun tertentu, gunakan wildcard (*):

```
"Resource": "arn:aws:iotsitewise:region:123456789012:asset/*"
```

Beberapa AWS IoT SiteWise tindakan, seperti untuk membuat sumber daya, tidak dapat dilakukan pada sumber daya tertentu. Dalam kasus tersebut, Anda harus menggunakan wildcard (*).

```
"Resource": "*"
```

Untuk menentukan beberapa sumber daya dalam satu pernyataan, pisahkan ARN dengan koma.

```
"Resource": [  
  "resource1",  
  "resource2"  
]
```

Untuk melihat daftar jenis AWS IoT SiteWise sumber daya dan ARNnya, lihat [Sumber Daya yang Ditentukan oleh AWS IoT SiteWise](#) dalam Panduan Pengguna IAM. Untuk mempelajari tindakan mana yang dapat menentukan ARN setiap sumber daya, lihat [Tindakan yang Ditentukan oleh AWS IoT SiteWise](#).

Kunci persyaratan kebijakan

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Elemen `Condition` (atau blok `Condition`) memungkinkan Anda menentukan kondisi di mana suatu pernyataan akan diterapkan. Elemen `Condition` bersifat opsional. Anda dapat membuat ekspresi kondisional yang menggunakan [operator kondisi](#), misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta.

Jika Anda menentukan beberapa elemen `Condition` dalam satu pernyataan, atau beberapa kunci dalam satu elemen `Condition`, AWS akan mengevaluasinya dengan menggunakan operasi AND logis. Jika Anda menentukan beberapa nilai untuk satu kunci kondisi, AWS mengevaluasi kondisi menggunakan OR operasi logis. Semua kondisi harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan kondisi. Misalnya, Anda dapat memberikan izin kepada pengguna IAM untuk mengakses sumber daya hanya jika izin tersebut mempunyai tag yang sesuai dengan nama pengguna IAM mereka. Untuk informasi selengkapnya, lihat [Elemen kebijakan IAM: variabel dan tag](#) di Panduan Pengguna IAM.

AWS mendukung kunci kondisi global dan kunci kondisi khusus layanan. Untuk melihat semua kunci kondisi AWS global, lihat [kunci konteks kondisi AWS global](#) di Panduan Pengguna IAM.

⚠ Important

Beberapa kunci syarat bersifat khusus untuk sumber daya, dan beberapa tindakan API menggunakan beberapa sumber daya. Jika Anda menulis pernyataan kebijakan dengan kunci kondisi, gunakan `Resource` elemen pernyataan untuk menentukan sumber daya yang diterapkan kunci kondisi. Jika Anda tidak melakukannya, kebijakan tersebut dapat mencegah pengguna melakukan tindakan sama sekali, karena pemeriksaan kondisi gagal untuk sumber daya yang tidak diterapkan oleh kunci kondisi. Jika Anda tidak ingin menentukan sumber daya, atau jika Anda telah menulis `Action` elemen kebijakan untuk menyertakan beberapa tindakan API, Anda harus menggunakan tipe `...IfExists` kondisi untuk memastikan bahwa kunci kondisi diabaikan untuk sumber daya yang tidak menggunakannya. Untuk informasi lebih lanjut, lihat... [IfExists](#) kondisi dalam Panduan Pengguna IAM.

AWS IoT SiteWise mendefinisikan kumpulan kunci kondisinya sendiri dan juga mendukung penggunaan beberapa kunci kondisi global. Untuk melihat semua kunci kondisi AWS global, lihat [kunci konteks kondisi AWS global](#) di Panduan Pengguna IAM.

AWS IoT SiteWise kunci kondisi

Kunci syarat	Deskripsi	Tipe
<code>iotsitewise:isAssociatedWithAssetProperty</code>	Apakah aliran data dikaitkan dengan properti aset. Gunakan kunci kondisi ini untuk menentukan izin berdasarkan keberadaan properti aset terkait untuk aliran data. Nilai contoh: <code>true</code>	String
<code>iotsitewise:assetHierarchyPath</code>	Jalur hierarki aset, yang merupakan string ID aset yang masing-masing dipisahkan oleh garis miring ke depan. Gunakan kunci kondisi ini untuk menentukan izin	String

Kunci syarat	Deskripsi	Tipe
	<p>berdasarkan subset hierarki semua aset di akun Anda.</p> <p>Nilai contoh: /a1b2c3d4-5678-90ab-cdef-2222EXAMPLE/a1b2c3d4-5678-90ab-cdef-6666EXAMPLE</p>	
<p><code>iotsitewise:propertyId</code></p>	<p>ID properti aset. Gunakan kunci kondisi ini untuk menentukan izin berdasarkan properti tertentu dari model aset. Kunci kondisi ini berlaku untuk semua aset model itu.</p> <p>Nilai contoh: a1b2c3d4-5678-90ab-cdef-3333EXAMPLE</p>	<p>String</p>
<p><code>iotsitewise:childAssetId</code></p>	<p>ID aset yang dikaitkan sebagai anak ke aset lain. Gunakan kunci kondisi ini untuk menentukan izin berdasarkan aset anak. Untuk menentukan izin berdasarkan aset induk, gunakan bagian sumber daya dari pernyataan kebijakan.</p> <p>Nilai contoh: a1b2c3d4-5678-90ab-cdef-6666EXAMPLE</p>	<p>String</p>

Kunci syarat	Deskripsi	Tipe
<code>iotsitewise:iam</code>	<p>ARN identitas IAM saat mencantumkan kebijakan akses. Gunakan kunci kondisi ini untuk menentukan izin kebijakan akses untuk identitas IAM.</p> <p>Nilai contoh: <code>arn:aws:iam::123456789012:user/JohnDoe</code></p>	Tali, Null
<code>iotsitewise:propertyAlias</code>	<p>Alias yang mengidentifikasi properti aset atau aliran data. Gunakan tombol kondisi ini untuk menentukan izin berdasarkan alias.</p>	String
<code>iotsitewise:user</code>	<p>ID pengguna Pusat Identitas IAM saat mencantumkan kebijakan akses. Gunakan kunci kondisi ini untuk menentukan izin kebijakan akses bagi pengguna Pusat Identitas IAM.</p> <p>Nilai contoh: <code>a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE</code></p>	Tali, Null

Kunci syarat	Deskripsi	Tipe
<code>iotsitewise:group</code>	<p>ID grup Pusat Identitas IAM saat mencantumkan kebijakan akses. Gunakan kunci kondisi ini untuk menentukan izin kebijakan akses untuk grup Pusat Identitas IAM.</p> <p>Nilai contoh: a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE</p>	Tali, Null
<code>iotsitewise:portal</code>	<p>ID portal dalam kebijakan akses. Gunakan kunci kondisi ini untuk menentukan izin kebijakan akses berdasarkan portal.</p> <p>Nilai contoh: a1b2c3d4-5678-90ab-cdef-77777EXAMPLE</p>	Tali, Null
<code>iotsitewise:project</code>	<p>ID proyek dalam kebijakan akses, atau ID proyek untuk dasbor. Gunakan tombol kondisi ini untuk menentukan dasbor atau akses izin kebijakan berdasarkan proyek.</p> <p>Nilai contoh: a1b2c3d4-5678-90ab-cdef-88888EXAMPLE</p>	Tali, Null

Untuk mempelajari tindakan dan sumber daya yang dapat Anda gunakan kunci kondisi, lihat [Tindakan yang Ditentukan oleh AWS IoT SiteWise](#).

Contoh-contoh

Untuk melihat contoh kebijakan AWS IoT SiteWise berbasis identitas, lihat [AWS IoT SiteWise contoh kebijakan berbasis identitas](#)

AWS IoT SiteWise contoh kebijakan berbasis identitas

Secara default, entitas (pengguna dan peran) tidak memiliki izin untuk membuat atau memodifikasi AWS IoT SiteWise sumber daya. Mereka juga tidak dapat melakukan tugas menggunakan AWS Management Console, AWS Command Line Interface (AWS CLI), atau AWS API. Untuk menyesuaikan izin, administrator AWS Identity and Access Management (IAM) harus melakukan hal berikut:

1. Buat kebijakan IAM yang memberikan izin kepada pengguna dan peran untuk melakukan operasi API tertentu pada sumber daya yang mereka butuhkan.
2. Lampirkan kebijakan tersebut ke pengguna atau grup yang memerlukan izin tersebut.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM menggunakan contoh dokumen kebijakan JSON ini, lihat [Membuat kebijakan pada tab JSON](#) dalam Panduan Pengguna IAM.

Topik

- [Praktik terbaik kebijakan](#)
- [Menggunakan konsol AWS IoT SiteWise](#)
- [Memungkinkan pengguna untuk melihat izin mereka sendiri](#)
- [Memungkinkan pengguna untuk menyerap data ke aset dalam satu hierarki](#)
- [Melihat AWS IoT SiteWise aset berdasarkan tag](#)

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus AWS IoT SiteWise sumber daya di akun Anda. Tindakan ini dikenai biaya untuk Akun AWS Anda. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan AWS terkelola dan beralih ke izin hak istimewa paling sedikit — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan AWS terkelola

yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat [kebijakan yang dikelola AWS](#) atau [kebijakan yang dikelola AWS untuk fungsi pekerjaan](#) di Panduan Pengguna IAM.

- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukan ini dengan menentukan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, juga dikenal sebagai izin hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk menerapkan izin, lihat [Kebijakan dan izin di IAM](#) di Panduan Pengguna IAM.
- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui yang spesifik Layanan AWS, seperti AWS CloudFormation. Untuk informasi selengkapnya, lihat [Elemen kebijakan JSON IAM: Syarat](#) di Panduan Pengguna IAM.
- Menggunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda guna memastikan izin yang aman dan berfungsi – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat [validasi kebijakan Analizer Akses IAM](#) di Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Anda, Akun AWS aktifkan MFA untuk keamanan tambahan. Untuk mewajibkan MFA saat operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat [Mengonfigurasi akses API yang dilindungi MFA](#) di Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat [Praktik terbaik keamanan di IAM](#) di Panduan Pengguna IAM.

Menggunakan konsol AWS IoT SiteWise

Untuk mengakses AWS IoT SiteWise konsol, Anda memerlukan satu set izin dasar. Izin ini memungkinkan Anda melihat dan mengelola detail tentang AWS IoT SiteWise sumber daya di Akun AWS.

Jika Anda membuat kebijakan yang terlalu ketat, konsol mungkin tidak berfungsi seperti yang diharapkan untuk pengguna atau peran (entitas) dengan kebijakan tersebut. Untuk memastikan bahwa entitas tersebut masih dapat menggunakan AWS IoT SiteWise konsol, lampirkan kebijakan [AWSIoTSiteWiseConsoleFullAccess](#) terkelola ke mereka atau tentukan izin yang setara untuk entitas tersebut. Untuk informasi selengkapnya, lihat [Menambahkan izin ke pengguna](#) di Panduan Pengguna IAM.

Jika entitas hanya menggunakan AWS Command Line Interface (CLI) atau AWS IoT SiteWise API, dan bukan konsol, mereka tidak memerlukan izin minimum ini. Dalam hal ini, cukup beri mereka akses ke tindakan spesifik yang mereka butuhkan untuk tugas API mereka.

Memungkinkan pengguna untuk melihat izin mereka sendiri

Contoh ini menunjukkan cara membuat kebijakan yang mengizinkan para pengguna IAM melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau menggunakan API atau secara terprogram. AWS CLI AWS

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
```

```

    "Effect": "Allow",
    "Action": [
      "iam:GetGroupPolicy",
      "iam:GetPolicyVersion",
      "iam:GetPolicy",
      "iam:ListAttachedGroupPolicies",
      "iam:ListGroupPolicies",
      "iam:ListPolicyVersions",
      "iam:ListPolicies",
      "iam:ListUsers"
    ],
    "Resource": "*"
  }
]
}

```

Memungkinkan pengguna untuk menyerap data ke aset dalam satu hierarki

Dalam contoh ini, Anda ingin memberikan pengguna dalam Akun AWS akses Anda untuk menulis data ke semua properti aset dalam hierarki aset tertentu, mulai dari aset `a1b2c3d4-5678-90ab-cdef-2222EXAMPLE` root. Kebijakan memberikan `iotsitewise:BatchPutAssetPropertyValue` izin kepada pengguna. Kebijakan ini menggunakan kunci `iotsitewise:assetHierarchyPath` kondisi untuk membatasi akses ke aset yang jalur hierarkinya cocok dengan aset atau turunannya.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PutAssetPropertyValuesForHierarchy",
      "Effect": "Allow",
      "Action": "iotsitewise:BatchPutAssetPropertyValue",
      "Resource": "arn:aws:iotsitewise:*:*:asset/*",
      "Condition": {
        "StringLike": {
          "iotsitewise:assetHierarchyPath": [
            "/a1b2c3d4-5678-90ab-cdef-2222EXAMPLE",
            "/a1b2c3d4-5678-90ab-cdef-2222EXAMPLE/*"
          ]
        }
      }
    }
  ]
}

```

```
}
```

Melihat AWS IoT SiteWise aset berdasarkan tag

Gunakan kondisi dalam kebijakan berbasis identitas Anda untuk mengontrol akses ke AWS IoT SiteWise sumber daya berdasarkan tag. Contoh ini menunjukkan cara membuat kebijakan yang memungkinkan penayangan aset. Namun, izin diberikan hanya jika tag aset `Owner` memiliki nilai nama pengguna pengguna tersebut. Kebijakan ini juga memberikan izin untuk menyelesaikan tindakan ini di konsol.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListAllAssets",
      "Effect": "Allow",
      "Action": [
        "iotsitewise:ListAssets",
        "iotsitewise:ListAssociatedAssets"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DescribeAssetIfOwner",
      "Effect": "Allow",
      "Action": "iotsitewise:DescribeAsset",
      "Resource": "arn:aws:iotsitewise:*:*:asset/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Owner": "${aws:username}"
        }
      }
    }
  ]
}
```

Lampirkan kebijakan ini ke pengguna di akun Anda. Jika pengguna bernama `richard-roe` mencoba untuk melihat AWS IoT SiteWise aset, aset harus ditandai `Owner=richard-roe` atau `owner=richard-roe`. Jika tidak, Richard ditolak aksesnya. Nama kunci tag kondisi tidak peka huruf besar/kecil. Jadi, `Owner` cocok dengan keduanya `Owner` dan `owner`. Untuk informasi selengkapnya, lihat [Elemen kebijakan IAM JSON: Syarat](#) dalam Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan adalah objek AWS yang, ketika dikaitkan dengan identitas atau sumber daya, menentukan izinnya. AWS mengevaluasi kebijakan ini ketika prinsipal (pengguna, pengguna root, atau sesi peran) membuat permintaan. Izin dalam kebijakan dapat menentukan permintaan yang diizinkan atau ditolak. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang struktur dan konten dokumen kebijakan JSON, lihat [Gambaran umum kebijakan JSON](#) dalam Panduan Pengguna IAM.

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, pengguna utama manakah yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan pada sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat menjalankan peran.

Kebijakan IAM mendefinisikan izin untuk suatu tindakan terlepas dari metode yang Anda gunakan untuk operasi. Sebagai contoh, anggap saja Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan tersebut bisa mendapatkan informasi peran dari AWS Management Console, API AWS CLI, atau AWS API.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan pengguna dan peran, di sumber daya mana, dan dengan ketentuan apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat [Membuat kebijakan IAM](#) dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan terkelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran dalam Akun AWS. Kebijakan AWS terkelola mencakup kebijakan terkelola dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan terkelola atau kebijakan inline, lihat [Memilih antara kebijakan terkelola dan kebijakan inline](#) dalam Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya yang dilampiri kebijakan tersebut, kebijakan ini menentukan jenis tindakan yang dapat dilakukan oleh pengguna utama tertentu di sumber daya tersebut dan apa ketentuannya. Anda harus [menentukan pengguna utama](#) dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

Daftar kontrol akses (ACL)

Daftar kontrol akses (ACL) mengendalikan pengguna utama mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACL sama dengan kebijakan berbasis sumber daya, meskipun tidak menggunakan format dokumen kebijakan JSON.

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung ACL. Untuk mempelajari ACL selengkapnya, silakan lihat [Gambaran umum daftar kontrol akses \(ACL\)](#) di Panduan Developer Layanan Penyimpanan Ringkas Amazon.

Tipe kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Tipe-tipe kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda berdasarkan tipe kebijakan yang lebih umum.

- Batasan izin – Batasan izin adalah fitur lanjutan di mana Anda menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM (pengguna atau peran IAM). Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan secara eksplisit terhadap salah satu kebijakan ini akan mengesampingkan izin tersebut. Untuk informasi selengkapnya tentang batasan izin, lihat [Batasan izin untuk entitas IAM](#) dalam Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCP) — SCP adalah kebijakan JSON yang menentukan izin maksimum untuk organisasi atau unit organisasi (OU) di. AWS Organizations AWS Organizations adalah

layanan untuk mengelompokkan dan mengelola secara terpusat beberapa Akun AWS yang dimiliki bisnis Anda. Jika Anda mengaktifkan semua fitur di sebuah organisasi, maka Anda dapat menerapkan kebijakan kontrol layanan (SCP) ke salah satu atau semua akun Anda. SCP membatasi izin untuk entitas di akun anggota, termasuk masing-masing. Pengguna root akun AWS Untuk informasi selengkapnya tentang Organisasi dan SCP, lihat [Cara kerja SCP](#) dalam Panduan Pengguna AWS Organizations .

- Kebijakan sesi – Kebijakan sesi adalah kebijakan lanjutan yang Anda teruskan sebagai parameter saat Anda membuat sesi sementara secara terprogram untuk peran atau pengguna gabungan. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan eksplisit dalam salah satu kebijakan ini membatalkan izin. Untuk informasi selengkapnya, lihat [Kebijakan sesi](#) dalam Panduan Pengguna IAM.

Beberapa jenis kebijakan

Ketika beberapa jenis kebijakan berlaku untuk sebuah permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat [Logika evaluasi kebijakan](#) di Panduan Pengguna IAM.

AWS kebijakan terkelola untuk AWS IoT SiteWise

Sederhanakan menambahkan izin ke pengguna, grup, dan peran menggunakan kebijakan AWS terkelola daripada menulis kebijakan sendiri. Dibutuhkan waktu dan keahlian untuk [membuat kebijakan terkelola pelanggan IAM](#) yang memberikan izin yang tepat kepada tim Anda. Untuk persiapan yang lebih cepat, pertimbangkan untuk menggunakan kebijakan AWS terkelola kami untuk kasus penggunaan umum. Temukan kebijakan AWS terkelola di Anda Akun AWS. Untuk informasi lebih lanjut tentang kebijakan yang dikelola AWS , lihat [kebijakan yang dikelola AWS](#) di Panduan Pengguna IAM.

AWS Layanan menangani pembaruan dan pemeliharaan kebijakan AWS terkelola, artinya Anda tidak dapat mengubah izin kebijakan ini. Terkadang, AWS IoT SiteWise dapat menambahkan izin untuk mengakomodasi fitur baru, yang memengaruhi semua identitas dengan kebijakan yang dilampirkan. Pembaruan semacam itu biasa terjadi pada pengenalan layanan atau fitur baru. Namun, izin tidak pernah dihapus, memastikan pengaturan Anda tetap utuh.

Selain itu, AWS mendukung kebijakan terkelola untuk fungsi pekerjaan yang mencakup beberapa layanan. Misalnya, kebijakan ReadOnlyAccess AWS terkelola menyediakan akses hanya-baca ke semua AWS layanan dan sumber daya. Saat layanan meluncurkan fitur baru, AWS tambahkan izin

hanya-baca untuk operasi dan sumber daya baru. Untuk daftar dengan deskripsi kebijakan fungsi pekerjaan, lihat [kebijakan AWS terkelola untuk fungsi pekerjaan](#) di Panduan Pengguna IAM.

AWS kebijakan terkelola: AWSIoTSiteWiseReadOnlyAccess

Gunakan kebijakan AWSIoTSiteWiseReadOnlyAccess AWS terkelola untuk mengizinkan akses hanya-baca. AWS IoT SiteWise

Anda dapat melampirkan kebijakan AWSIoTSiteWiseReadOnlyAccess ke identitas IAM Anda.

Izin tingkat layanan

Kebijakan ini menyediakan akses hanya-baca ke. AWS IoT SiteWise Tidak ada izin layanan lain yang disertakan dalam kebijakan ini.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:Describe*",
        "iotsitewise:List*",
        "iotsitewise:BatchGet*",
        "iotsitewise:Get*"
      ],
      "Resource": "*"
    }
  ]
}
```

AWS kebijakan terkelola: AWSServiceRoleForIoTSiteWise

AWSServiceRoleForIoTSiteWisePeran menggunakan AWSServiceRoleForIoTSiteWise kebijakan dengan izin berikut. Kebijakan ini:

- Memungkinkan AWS IoT SiteWise untuk menyebarkan gateway SiteWise Edge (yang berjalan).
AWS IoT Greengrass
- Memungkinkan AWS IoT SiteWise untuk melakukan logging.
- Memungkinkan AWS IoT SiteWise untuk menjalankan kueri pencarian metadata, terhadap database. AWS IoT TwinMaker

Jika Anda menggunakan AWS IoT SiteWise akun pengguna singe, `AWSServiceRoleForIoTSiteWise` peran akan membuat `AWSServiceRoleForIoTSiteWise` kebijakan di akun IAM Anda, dan melampirkannya ke peran yang ditautkan `AWSServiceRoleForIoTSiteWise` [layanan](#) untuk. AWS IoT SiteWise

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSiteWiseReadGreenGrass",
      "Effect": "Allow",
      "Action": [
        "greengrass:GetAssociatedRole",
        "greengrass:GetCoreDefinition",
        "greengrass:GetCoreDefinitionVersion",
        "greengrass:GetGroup",
        "greengrass:GetGroupVersion"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowSiteWiseAccessLogGroup",
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:DescribeLogGroups"
      ],
      "Resource": "arn:aws:logs:*:*:log-group:/aws/iotsitewise*"
    },
    {
      "Sid": "AllowSiteWiseAccessLog",
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogStream",
        "logs:DescribeLogStreams",
        "logs:PutLogEvents"
      ],
      "Resource": "arn:aws:logs:*:*:log-group:/aws/iotsitewise*:log-stream:*"
    },
    {
      "Sid": "AllowSiteWiseAccessSiteWiseManagedWorkspaceInTwinMaker",
      "Effect": "Allow",
```



```

"Action": [
  "iottwinmaker:GetWorkspace",
  "iottwinmaker:ExecuteQuery"
],
"Resource": "arn:aws:iottwinmaker:*:*:workspace/*",
"Condition": {
  "ForAnyValue:StringEquals": {
    "iottwinmaker:linkedServices": [
      "IOTSITWISE"
    ]
  }
}
}
]
}

```

AWS IoT SiteWise pembaruan kebijakan AWS terkelola

Anda dapat melihat detail tentang pembaruan kebijakan AWS terkelola untuk AWS IoT SiteWise, mulai dari saat layanan ini mulai melacak perubahan. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan umpan RSS di halaman Riwayat AWS IoT SiteWise dokumen.

Perubahan	Deskripsi	Tanggal
AWSServiceRoleForIoTSiteWise — Pembaruan ke kebijakan yang sudah ada	AWS IoT SiteWise sekarang dapat menjalankan kueri pencarian metadata, terhadap database. AWS IoT TwinMaker	6 November 2023
AWSIoTSiteWiseReadOnlyAccess – Pembaruan ke kebijakan yang ada	AWS IoT SiteWise menambahkan awalan kebijakan baru, <code>BatchGet*</code> , yang memungkinkan Anda melakukan operasi pembacaan batch.	September 16, 2022
AWSIoTSiteWiseReadOnlyAccess – Kebijakan baru	AWS IoT SiteWise menambahkan kebijakan baru untuk memberikan akses	24 November 2021

Perubahan	Deskripsi	Tanggal
	hanya-baca ke. AWS IoT SiteWise	
AWS IoT SiteWise mulai melacak perubahan	AWS IoT SiteWise mulai melacak perubahan untuk kebijakan yang AWS dikelola.	24 November 2021

Menggunakan peran tertaut layanan untuk AWS IoT SiteWise

AWS IoT SiteWise menggunakan AWS Identity and Access Management peran [terkait layanan](#) (IAM). Peran terkait layanan adalah jenis unik peran IAM yang ditautkan langsung ke. AWS IoT SiteWise Peran terkait layanan telah ditentukan sebelumnya oleh AWS IoT SiteWise dan mencakup semua izin yang diperlukan layanan untuk memanggil AWS layanan lain atas nama Anda.

Peran terkait layanan menyederhanakan konfigurasi AWS IoT SiteWise dengan secara otomatis menyertakan semua izin yang diperlukan. AWS IoT SiteWise mendefinisikan izin peran terkait layanan, dan kecuali ditentukan lain, hanya AWS IoT SiteWise dapat mengambil perannya. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin. Dan kebijakan izin tersebut tidak dapat dilampirkan ke entitas IAM lainnya.

Anda dapat menghapus peran tertaut layanan hanya setelah menghapus sumber daya terkait terlebih dahulu. Ini melindungi AWS IoT SiteWise sumber daya Anda karena Anda tidak dapat secara tidak sengaja menghapus izin untuk mengakses sumber daya.

Untuk informasi tentang layanan lain yang mendukung peran terkait layanan, lihat [Layanan AWS yang berfungsi dengan IAM](#) lalu cari layanan yang menampilkan Ya pada kolom Peran Terkait Layanan. Pilih Ya dengan tautan untuk melihat dokumentasi peran tertaut layanan untuk layanan tersebut.

Topik

- [Izin peran terkait layanan untuk AWS IoT SiteWise](#)
- [Membuat peran yang terkait dengan layanan untuk AWS IoT SiteWise](#)
- [Mengedit peran terkait layanan untuk AWS IoT SiteWise](#)
- [Menghapus peran terkait layanan untuk AWS IoT SiteWise](#)
- [Wilayah yang Didukung untuk AWS IoT SiteWise peran terkait layanan](#)

- [Menggunakan peran layanan untuk AWS IoT SiteWise Monitor](#)

Izin peran terkait layanan untuk AWS IoT SiteWise

AWS IoT SiteWise menggunakan peran terkait layanan bernama. `AWSServiceRoleForIoTSiteWise` menggunakan peran terkait layanan ini untuk menerapkan gateway SiteWise Edge (yang berjalan) dan melakukan logging. AWS IoT Greengrass

Peran `AWSServiceRoleForIoTSiteWise` terkait layanan menggunakan `AWSServiceRoleForIoTSiteWise` kebijakan dengan izin berikut. Kebijakan ini:

- Memungkinkan AWS IoT SiteWise untuk menyebarkan gateway SiteWise Edge (yang berjalan). AWS IoT Greengrass
- Memungkinkan AWS IoT SiteWise untuk melakukan logging.
- Memungkinkan AWS IoT SiteWise untuk menjalankan kueri pencarian metadata, terhadap database. AWS IoT TwinMaker

Untuk informasi selengkapnya tentang tindakan yang diizinkan `AWSServiceRoleForIoTSiteWise`, lihat [kebijakan AWS terkelola untuk AWS IoT SiteWise](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSiteWiseReadGreenGrass",
      "Effect": "Allow",
      "Action": [
        "greengrass:GetAssociatedRole",
        "greengrass:GetCoreDefinition",
        "greengrass:GetCoreDefinitionVersion",
        "greengrass:GetGroup",
        "greengrass:GetGroupVersion"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowSiteWiseAccessLogGroup",
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
```

```

    "logs:DescribeLogGroups"
  ],
  "Resource": "arn:aws:logs:*:*:log-group:/aws/iotsitewise*"
},
{
  "Sid": "AllowSiteWiseAccessLog",
  "Effect": "Allow",
  "Action": [
    "logs:CreateLogStream",
    "logs:DescribeLogStreams",
    "logs:PutLogEvents"
  ],
  "Resource": "arn:aws:logs:*:*:log-group:/aws/iotsitewise*:log-stream:*"
},
{
  "Sid": "AllowSiteWiseAccessSiteWiseManagedWorkspaceInTwinMaker",
  "Effect": "Allow",
  "Action": [
    "iottwinmaker:GetWorkspace",
    "iottwinmaker:ExecuteQuery"
  ],
  "Resource": "arn:aws:iottwinmaker:*:*:workspace/*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "iottwinmaker:linkedServices": [
        "IOTSITewise"
      ]
    }
  }
}
]
}

```

Anda dapat menggunakan log untuk memantau dan memecahkan masalah gateway SiteWise Edge Anda. Untuk informasi selengkapnya, lihat [Memantau log gateway SiteWise Edge](#).

Untuk mengizinkan entitas IAM (seperti pengguna, grup, atau peran) membuat, mengedit, atau menghapus peran terkait layanan, pertama-tama konfigurasi izin. Untuk informasi selengkapnya, lihat [Izin peran yang terkait dengan layanan](#) dalam Panduan Pengguna IAM.

Membuat peran yang terkait dengan layanan untuk AWS IoT SiteWise

Anda tidak perlu membuat peran terkait layanan secara manual. Saat Anda melakukan operasi berikut di AWS IoT SiteWise konsol, AWS IoT SiteWise buat peran terkait layanan untuk Anda.

- Buat gateway Greengrass V1.
- Konfigurasi opsi logging.
- Memilih tombol keikutsertaan di spanduk eksekusi kueri.

Jika Anda menghapus peran terkait layanan ini, dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda. Saat Anda melakukan operasi apa pun di AWS IoT SiteWise konsol, AWS IoT SiteWise buat peran terkait layanan untuk Anda lagi.

Anda juga dapat menggunakan konsol IAM atau API untuk membuat peran terkait layanan. AWS IoT SiteWise

- Untuk melakukannya di konsol IAM, buat peran dengan `AWSServiceRoleForIoTSiteWise` kebijakan dan hubungan kepercayaan dengan `aniotsitewise.amazonaws.com`.
- Untuk melakukannya menggunakan API AWS CLI atau IAM, buat peran dengan `arn:aws:iam::aws:policy/aws-service-role/AWSServiceRoleForIoTSiteWise` kebijakan dan hubungan kepercayaan dengan `aniotsitewise.amazonaws.com`.

Untuk informasi selengkapnya, lihat [Membuat peran tertaut layanan](#) dalam Panduan Pengguna IAM.

Jika Anda menghapus peran yang terhubung dengan layanan ini, Anda dapat menggunakan proses yang sama untuk membuat ulang peran tersebut.

Mengedit peran terkait layanan untuk AWS IoT SiteWise

AWS IoT SiteWise tidak memungkinkan Anda untuk mengedit peran `AWSServiceRoleForIoTSiteWise` terkait layanan. Setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat menyunting penjelasan peran menggunakan IAM. Untuk informasi lebih lanjut, lihat [Mengedit peran terkait layanan](#) dalam Panduan Pengguna IAM.

Menghapus peran terkait layanan untuk AWS IoT SiteWise

Jika fitur atau layanan yang memerlukan peran terkait layanan tidak lagi digunakan, disarankan untuk menghapus peran terkait. Ini untuk menghindari entitas tidak aktif yang tidak dipantau atau dipelihara. Tetapi, Anda harus membersihkan sumber daya peran terkait layanan sebelum menghapusnya secara manual.

Note

Jika AWS IoT SiteWise layanan menggunakan peran saat Anda mencoba menghapus sumber daya, maka penghapusan mungkin gagal. Jika hal tersebut terjadi, tunggu beberapa menit dan coba lagi.

Untuk menghapus AWS IoT SiteWise sumber daya yang digunakan oleh `AWSServiceRoleForIoTSiteWise`

1. Nonaktifkan logging untuk AWS IoT SiteWise. Lihat informasi yang lebih lengkap di [Mengubah level logging Anda](#)
2. Hapus gateway SiteWise Edge yang aktif.

Untuk menghapus peran terkait layanan secara manual menggunakan IAM

Gunakan konsol IAM, the AWS CLI, atau AWS API untuk menghapus peran `AWSServiceRoleForIoTSiteWise` terkait layanan. Untuk informasi selengkapnya, silakan lihat [Menghapus Peran Terkait Layanan](#) di Panduan Pengguna IAM.

Wilayah yang Didukung untuk AWS IoT SiteWise peran terkait layanan

AWS IoT SiteWise mendukung penggunaan peran terkait layanan di semua Wilayah tempat layanan tersedia. Untuk informasi lebih lanjut, lihat [Kuota dan Titik Akhir AWS IoT SiteWise](#).

Menggunakan peran layanan untuk AWS IoT SiteWise Monitor

Peran layanan adalah [peran IAM](#) yang diambil oleh layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat [Membuat peran untuk mendelegasikan izin ke Layanan AWS](#) dalam Panduan pengguna IAM.

Untuk memungkinkan pengguna portal SiteWise Monitor federasi mengakses sumber daya Anda AWS IoT SiteWise dan AWS IAM Identity Center sumber daya, Anda harus melampirkan peran layanan ke setiap portal yang Anda buat. Peran layanan harus menentukan SiteWise Monitor sebagai entitas tepercaya dan menyertakan kebijakan [AWSIoTSiteWiseMonitorPortalAccess](#) terkelola atau menentukan [izin yang setara](#). Kebijakan ini dikelola oleh AWS dan menentukan kumpulan izin yang digunakan SiteWise Monitor untuk mengakses sumber daya Pusat Identitas IAM Anda AWS IoT SiteWise dan IAM.

Saat membuat portal SiteWise Monitor, Anda harus memilih peran yang memungkinkan pengguna portal tersebut mengakses sumber daya Pusat Identitas IAM AWS IoT SiteWise dan IAM Anda. AWS IoT SiteWise Konsol dapat membuat dan mengonfigurasi peran untuk Anda. Anda dapat mengedit peran di IAM nanti. Pengguna portal Anda akan mengalami masalah dalam menggunakan portal SiteWise Monitor mereka jika Anda menghapus izin yang diperlukan dari peran atau menghapus peran tersebut.

Note

Portal yang dibuat sebelum 29 April 2020 tidak memerlukan peran layanan. Jika Anda membuat portal sebelum tanggal ini, Anda harus melampirkan peran layanan untuk terus menggunakannya. Untuk melakukannya, navigasikan ke halaman Portal di [AWS IoT SiteWise konsol](#), lalu pilih Migrasi semua portal untuk menggunakan peran IAM.

Bagian berikut menjelaskan cara membuat dan mengelola peran layanan SiteWise Monitor di AWS Management Console atau AWS Command Line Interface.

Daftar Isi

- [Izin peran layanan untuk Monitor SiteWise](#)
- [Mengelola peran layanan SiteWise Monitor \(konsol\)](#)
 - [Menemukan peran layanan portal \(konsol\)](#)
 - [Membuat peran layanan SiteWise Monitor \(AWS IoT SiteWise konsol\)](#)
 - [Membuat peran layanan SiteWise Monitor \(konsol IAM\)](#)
 - [Mengubah peran layanan portal \(konsol\)](#)
- [Mengelola peran layanan SiteWise Monitor \(CLI\)](#)
 - [Menemukan peran layanan portal \(CLI\)](#)
 - [Membuat peran layanan SiteWise Monitor \(CLI\)](#)

- [SiteWise Pantau pembaruan ke AWSIoTSiteWiseMonitorServiceRole](#)

Izin peran layanan untuk Monitor SiteWise

Saat membuat portal, AWS IoT SiteWise Anda dapat membuat peran yang namanya dimulai AWSIoTSiteWiseMonitorServiceRole. Peran ini memungkinkan pengguna SiteWise Monitor federasi untuk mengakses konfigurasi portal, aset, data aset, dan konfigurasi Pusat Identitas IAM Anda.

Peran tersebut mempercayai layanan berikut untuk mengambil peran:

- `monitor.iotsitewise.amazonaws.com`

Peran tersebut menggunakan kebijakan izin berikut, yang namanya dimulai dengan AWSIoTSiteWiseMonitorServicePortalPolicy, untuk memungkinkan pengguna SiteWise Monitor menyelesaikan tindakan pada sumber daya di akun Anda. Kebijakan [AWSIoTSiteWiseMonitorPortalAccess](#)terkelola mendefinisikan izin yang setara.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:DescribePortal",
        "iotsitewise:CreateProject",
        "iotsitewise:DescribeProject",
        "iotsitewise:UpdateProject",
        "iotsitewise>DeleteProject",
        "iotsitewise:ListProjects",
        "iotsitewise:BatchAssociateProjectAssets",
        "iotsitewise:BatchDisassociateProjectAssets",
        "iotsitewise:ListProjectAssets",
        "iotsitewise:CreateDashboard",
        "iotsitewise:DescribeDashboard",
        "iotsitewise:UpdateDashboard",
        "iotsitewise>DeleteDashboard",
        "iotsitewise:ListDashboards",
        "iotsitewise:CreateAccessPolicy",
        "iotsitewise:DescribeAccessPolicy",
        "iotsitewise:UpdateAccessPolicy",
        "iotsitewise>DeleteAccessPolicy",

```



```

        "iotsitewise:ListAccessPolicies",
        "iotsitewise:DescribeAsset",
        "iotsitewise:ListAssets",
        "iotsitewise:ListAssociatedAssets",
        "iotsitewise:DescribeAssetProperty",
        "iotsitewise:GetAssetPropertyValue",
        "iotsitewise:GetAssetPropertyValueHistory",
        "iotsitewise:GetAssetPropertyAggregates",
        "iotsitewise:BatchPutAssetPropertyValue",
        "iotsitewise:ListAssetRelationships",
        "iotsitewise:DescribeAssetModel",
        "iotsitewise:ListAssetModels",
        "iotsitewise:UpdateAssetModel",
        "iotsitewise:UpdateAssetModelPropertyRouting",
        "sso-directory:DescribeUsers",
        "sso-directory:DescribeUser",
        "iotevents:DescribeAlarmModel",
        "iotevents:ListTagsForResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "iotevents:BatchAcknowledgeAlarm",
        "iotevents:BatchSnoozeAlarm",
        "iotevents:BatchEnableAlarm",
        "iotevents:BatchDisableAlarm"
    ],
    "Resource": "*",
    "Condition": {
        "Null": {
            "iotevents:keyValue": "false"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "iotevents:CreateAlarmModel",
        "iotevents:TagResource"
    ],
    "Resource": "*",
    "Condition": {

```

```

        "Null": {
            "aws:RequestTag/iotsitewisemonitor": "false"
        }
    },
    {
        "Effect": "Allow",
        "Action": [
            "iotevents:UpdateAlarmModel",
            "iotevents:DeleteAlarmModel"
        ],
        "Resource": "*",
        "Condition": {
            "Null": {
                "aws:ResourceTag/iotsitewisemonitor": "false"
            }
        }
    },
    {
        "Effect": "Allow",
        "Action": [
            "iam:PassRole"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "iam:PassedToService": [
                    "iotevents.amazonaws.com"
                ]
            }
        }
    }
]
}

```

Untuk informasi selengkapnya tentang izin yang diperlukan untuk alarm, lihat [Menyiapkan izin untuk alarm AWS IoT Events](#)

Saat pengguna portal masuk, SiteWise Monitor akan membuat [kebijakan sesi](#) berdasarkan persimpangan peran layanan dan kebijakan akses pengguna tersebut. Kebijakan akses menentukan tingkat akses identitas ke portal dan proyek Anda. Untuk informasi selengkapnya tentang izin portal dan kebijakan akses, lihat [Mengelola portal SiteWise Monitor Anda](#) dan [CreateAccessPolicy](#).

Mengelola peran layanan SiteWise Monitor (konsol)

Konsol AWS IoT SiteWise Memfasilitasi pengelolaan peran layanan SiteWise Monitor untuk portal. Setelah membuat portal, konsol memeriksa peran yang ada yang cocok untuk lampiran. Jika tidak ada yang tersedia, konsol dapat membuat dan mengonfigurasi peran layanan untuk Anda. Untuk informasi selengkapnya, lihat [Membuat portal](#).

Topik

- [Menemukan peran layanan portal \(konsol\)](#)
- [Membuat peran layanan SiteWise Monitor \(AWS IoT SiteWise konsol\)](#)
- [Membuat peran layanan SiteWise Monitor \(konsol IAM\)](#)
- [Mengubah peran layanan portal \(konsol\)](#)

Menemukan peran layanan portal (konsol)

Gunakan langkah-langkah berikut untuk menemukan peran layanan yang dilampirkan ke portal SiteWise Monitor.

Untuk menemukan peran layanan portal

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Portal.
3. Pilih portal yang ingin Anda temukan peran layanannya.

Peran yang dilampirkan ke portal muncul di bawah Izin, peran Layanan.

Membuat peran layanan SiteWise Monitor (AWS IoT SiteWise konsol)

Saat membuat portal SiteWise Monitor, Anda dapat membuat peran layanan untuk portal Anda. Untuk informasi selengkapnya, lihat [Membuat portal](#).

Anda juga dapat membuat peran layanan untuk portal yang ada di AWS IoT SiteWise konsol. Ini menggantikan peran layanan portal yang ada.

Untuk membuat peran layanan untuk portal yang ada

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Portal.

3. Pilih portal yang ingin Anda buat peran layanan baru.
4. Di bawah Detail portal, pilih Edit.
5. Di bawah Izin, pilih Buat dan gunakan peran layanan baru dari daftar.
6. Masukkan nama untuk peran baru Anda.
7. Pilih Simpan.

Membuat peran layanan SiteWise Monitor (konsol IAM)

Anda dapat membuat peran layanan dari templat peran layanan di konsol IAM. Templat peran ini mencakup kebijakan [AWSIoTSiteWiseMonitorPortalAccess](#)sterkelola dan menetapkan SiteWise Monitor sebagai entitas tepercaya.

Untuk membuat peran layanan dari template peran layanan portal

1. Arahkan ke [konsol IAM](#).
2. Di panel navigasi, pilih Peran.
3. Pilih Buat peran.
4. Dalam Pilih kasus penggunaan, pilih IoT SiteWise.
5. Di Pilih kasus penggunaan Anda, pilih IoT SiteWise Monitor - Portal.
6. Pilih Selanjutnya: Izin.
7. Pilih Berikutnya: Tanda.
8. Pilih Berikutnya: Tinjauan.
9. Masukkan nama Peran untuk peran layanan baru.
10. Pilih Buat peran.

Mengubah peran layanan portal (konsol)

Gunakan prosedur berikut untuk memilih peran layanan SiteWise Monitor yang berbeda untuk portal.

Untuk mengubah peran layanan portal

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi, pilih Portal.
3. Pilih portal yang ingin Anda ubah peran layanannya.

4. Di bawah Detail portal, pilih Edit.
5. Di bawah Izin, pilih Gunakan peran yang ada.
6. Pilih peran yang ada untuk dilampirkan ke portal ini.
7. Pilih Simpan.

Mengelola peran layanan SiteWise Monitor (CLI)

Anda dapat menggunakan AWS CLI untuk tugas manajemen peran layanan portal berikut:

Topik

- [Menemukan peran layanan portal \(CLI\)](#)
- [Membuat peran layanan SiteWise Monitor \(CLI\)](#)

Menemukan peran layanan portal (CLI)

Untuk menemukan peran layanan yang dilampirkan ke portal SiteWise Monitor, jalankan perintah berikut untuk mencantumkan semua portal Anda di Wilayah saat ini.

```
aws iotsitewise list-portals
```

Operasi mengembalikan respons yang berisi ringkasan portal Anda dalam format berikut.

```
{
  "portalSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
      "name": "WindFarmPortal",
      "description": "A portal that contains wind farm projects for Example Corp.",
      "roleArn": "arn:aws:iam::123456789012:role/service-role/role-name",
      "startUrl": "https://a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE.app.iotsitewise.aws",
      "creationDate": "2020-02-04T23:01:52.90248068Z",
      "lastUpdateDate": "2020-02-04T23:01:52.90248078Z"
    }
  ]
}
```

Anda juga dapat menggunakan [DescribePortal](#) operasi untuk menemukan peran portal Anda jika Anda mengetahui ID portal Anda.

Membuat peran layanan SiteWise Monitor (CLI)

Gunakan langkah-langkah berikut untuk membuat peran layanan SiteWise Monitor baru.

Untuk membuat peran layanan SiteWise Monitor

1. Buat peran dengan kebijakan kepercayaan yang memungkinkan SiteWise Monitor untuk mengambil peran. Contoh ini membuat peran bernama **MySiteWiseMonitorPortalRole** dari kebijakan kepercayaan yang disimpan dalam string JSON.

Linux, macOS, or Unix

```
aws iam create-role --role-name MySiteWiseMonitorPortalRole --assume-role-policy-document '{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "monitor.iotsitewise.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}'
```

Windows command prompt

```
aws iam create-role --role-name MySiteWiseMonitorPortalRole --assume-role-policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"Service\":\"monitor.iotsitewise.amazonaws.com\"},\"Action\":\"sts:AssumeRole\"}]}"
```

2. Salin peran ARN dari metadata peran dalam output. Saat Anda membuat portal, Anda menggunakan ARN ini untuk mengaitkan peran dengan portal Anda. Untuk informasi selengkapnya tentang membuat portal, lihat [CreatePortal](#) di Referensi AWS IoT SiteWise API.
3. Lampirkan `AWSIoTSiteWiseMonitorPortalAccess` kebijakan ke peran, atau lampirkan kebijakan yang menentukan izin yang setara.

```
aws iam attach-role-policy --role-name MySiteWiseMonitorPortalRole --policy-arn arn:aws:iam::aws:policy/service-role/AWSIoTSiteWiseMonitorPortalAccess
```

Untuk melampirkan peran layanan ke portal yang ada

1. Untuk mengambil detail portal yang ada, jalankan perintah berikut. Ganti *portal-id* dengan ID portal.

```
aws iotsitewise describe-portal --portal-id portal-id
```

Operasi mengembalikan respon yang berisi rincian portal dalam format berikut.

```
{
  "portalId": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
  "portalArn": "arn:aws:iotsitewise:region:account-id:portal/a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
  "portalName": "WindFarmPortal",
  "portalDescription": "A portal that contains wind farm projects for Example Corp.",
  "portalClientId": "E-1a2b3c4d5e6f_sn6tbqHVzLWVEXAMPLE",
  "portalStartUrl": "https://a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE.app.iotsitewise.aws",
  "portalContactEmail": "support@example.com",
  "portalStatus": {
    "state": "ACTIVE"
  },
  "portalCreationDate": "2020-04-29T23:01:52.90248068Z",
  "portalLastUpdateDate": "2020-04-29T00:28:26.103548287Z",
  "roleArn": "arn:aws:iam::123456789012:role/service-role/AWSIoTSiteWiseMonitorServiceRole_1aEXAMPLE"
}
```

2. Untuk melampirkan peran layanan ke portal, jalankan perintah berikut. Ganti *role-arn* dengan ARN peran layanan, dan ganti parameter yang tersisa dengan nilai portal yang ada.

```
aws iotsitewise update-portal \
  --portal-id portal-id \
  --role-arn role-arn \
  --portal-name portal-name \
  --portal-description portal-description \
  --portal-contact-email portal-contact-email
```

SiteWise Pantau pembaruan ke AWSIoTSiteWiseMonitorServiceRole

Anda dapat melihat detail tentang pembaruan AWSIoTSiteWiseMonitorServiceRole untuk SiteWise Monitor, mulai dari saat layanan ini mulai melacak perubahan. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan umpan RSS di halaman Riwayat AWS IoT SiteWise dokumen.

Perubahan	Deskripsi	Tanggal
AWSIoTSiteWiseMonitorPortal Access — Kebijakan yang diperbarui	AWS IoT SiteWise memperbarui kebijakan AWSIoTSiteWiseMonitorPortal Access terkelola untuk fitur alarm.	27 Mei 2021
AWS IoT SiteWise mulai melacak perubahan	AWS IoT SiteWise mulai melacak perubahan untuk peran layanannya.	15 Desember 2020

Menyiapkan izin untuk alarm AWS IoT Events

Bila Anda menggunakan model AWS IoT Events alarm untuk memantau properti AWS IoT SiteWise aset, Anda harus memiliki izin IAM berikut:

- Peran AWS IoT Events layanan yang memungkinkan AWS IoT Events untuk mengirim data ke AWS IoT SiteWise. Untuk informasi selengkapnya, lihat [Identitas dan manajemen akses AWS IoT Events](#) di Panduan AWS IoT Events Pengembang.
- Anda harus memiliki izin AWS IoT SiteWise tindakan berikut:
`ioticsitewise:DescribeAssetModel`
`danioticsitewise:UpdateAssetModelPropertyRouting`. Izin ini memungkinkan AWS IoT SiteWise untuk mengirim nilai properti aset ke model AWS IoT Events alarm.

Untuk informasi selengkapnya, lihat [Kebijakan berbasis sumber daya](#) di Panduan Pengguna IAM.

Izin tindakan yang diperlukan

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa. Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan.

Sebelum Anda menentukan model AWS IoT Events alarm, Anda harus memberikan izin berikut yang memungkinkan AWS IoT SiteWise untuk mengirim nilai properti aset ke model alarm.

- `iotsitewise:DescribeAssetModel`— Memungkinkan AWS IoT Events untuk memeriksa apakah properti aset ada.
- `iotsitewise:UpdateAssetModelPropertyRouting`— Memungkinkan AWS IoT SiteWise untuk secara otomatis membuat langganan yang memungkinkan AWS IoT SiteWise untuk mengirim data ke AWS IoT Events.

Untuk informasi selengkapnya tentang tindakan yang AWS IoT SiteWise didukung, lihat [Tindakan yang ditentukan oleh AWS IoT SiteWise](#) dalam Referensi Otorisasi Layanan.

Example Contoh kebijakan izin 1

Kebijakan berikut memungkinkan AWS IoT SiteWise untuk mengirim nilai properti aset ke model AWS IoT Events alarm apa pun.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotevents:CreateAlarmModel",
        "iotevents:UpdateAlarmModel"
      ],
      "Resource": "arn:aws:iotevents:us-east-1:123456789012:alarmModel/*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:DescribeAssetModel",
        "iotsitewise:UpdateAssetModelPropertyRouting"
      ]
    }
  ]
}
```

```

    ],
    "Resource": "arn:aws:iotsitewise:us-east-1:123456789012:asset-model/*"
  }
]
}

```

Example Contoh kebijakan izin 2

Kebijakan berikut memungkinkan AWS IoT SiteWise untuk mengirim nilai properti aset tertentu ke model AWS IoT Events alarm tertentu.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotevents:CreateAlarmModel",
        "iotevents:UpdateAlarmModel"
      ],
      "Resource": "arn:aws:iotevents:us-east-1:123456789012:alarmModel/*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:DescribeAssetModel"
      ],
      "Resource": "arn:aws:iotsitewise:us-east-1:123456789012:asset-model/*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:UpdateAssetModelPropertyRouting"
      ],
      "Resource": [
        "arn:aws:iotsitewise:us-east-1:123456789012:asset-model/12345678-90ab-cdef-1234-567890abcdef"
      ],
      "Condition": {
        "StringLike": {
          "iotsitewise:propertyId": "abcdef12-3456-7890-abcd-ef1234567890",
          "iotevents:alarmModelArn": "arn:aws:iotevents:us-east-1:123456789012:alarmModel/MyAlarmModel"
        }
      }
    }
  ]
}

```

```

    }
  }
]
}

```

ListInputRoutings Izin (Opsional)

Saat Anda memperbarui atau menghapus model aset, AWS IoT SiteWise dapat memeriksa apakah model alarm di AWS IoT Events memantau properti aset yang terkait dengan model aset ini. Ini mencegah Anda menghapus properti aset yang saat ini digunakan AWS IoT Events alarm. Untuk mengaktifkan fitur ini AWS IoT SiteWise, Anda harus memiliki `iotevents:ListInputRoutings` izin. Izin ini memungkinkan AWS IoT SiteWise untuk melakukan panggilan ke operasi [ListInputRoutings](#) API yang didukung oleh AWS IoT Events.

Note

Kami sangat menyarankan Anda menambahkan `ListInputRoutings` izin.

Example Contoh kebijakan izin

Kebijakan berikut memungkinkan Anda memperbarui dan menghapus model aset, dan menggunakan `ListInputRoutings` API di dalamnya AWS IoT SiteWise.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:UpdateAssetModel",
        "iotsitewise>DeleteAssetModel",
        "iotevents:ListInputRoutings"
      ],
      "Resource": "arn:aws:iotsitewise:us-east-1:123456789012:asset-model/*"
    }
  ]
}

```

Izin yang diperlukan untuk Monitor SiteWise

Jika ingin menggunakan fitur alarm di portal SiteWise Monitor, Anda harus memperbarui [peran layanan SiteWise Monitor](#) dengan kebijakan berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:DescribePortal",
        "iotsitewise:CreateProject",
        "iotsitewise:DescribeProject",
        "iotsitewise:UpdateProject",
        "iotsitewise>DeleteProject",
        "iotsitewise:ListProjects",
        "iotsitewise:BatchAssociateProjectAssets",
        "iotsitewise:BatchDisassociateProjectAssets",
        "iotsitewise:ListProjectAssets",
        "iotsitewise:CreateDashboard",
        "iotsitewise:DescribeDashboard",
        "iotsitewise:UpdateDashboard",
        "iotsitewise>DeleteDashboard",
        "iotsitewise:ListDashboards",
        "iotsitewise:CreateAccessPolicy",
        "iotsitewise:DescribeAccessPolicy",
        "iotsitewise:UpdateAccessPolicy",
        "iotsitewise>DeleteAccessPolicy",
        "iotsitewise:ListAccessPolicies",
        "iotsitewise:DescribeAsset",
        "iotsitewise:ListAssets",
        "iotsitewise:ListAssociatedAssets",
        "iotsitewise:DescribeAssetProperty",
        "iotsitewise:GetAssetPropertyValue",
        "iotsitewise:GetAssetPropertyValueHistory",
        "iotsitewise:GetAssetPropertyAggregates",
        "iotsitewise:BatchPutAssetPropertyValue",
        "iotsitewise:ListAssetRelationships",
        "iotsitewise:DescribeAssetModel",
        "iotsitewise:ListAssetModels",
        "iotsitewise:UpdateAssetModel",
        "iotsitewise:UpdateAssetModelPropertyRouting",

```

```

        "sso-directory:DescribeUsers",
        "sso-directory:DescribeUser",
        "iotevents:DescribeAlarmModel",
        "iotevents:ListTagsForResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "iotevents:BatchAcknowledgeAlarm",
        "iotevents:BatchSnoozeAlarm",
        "iotevents:BatchEnableAlarm",
        "iotevents:BatchDisableAlarm"
    ],
    "Resource": "*",
    "Condition": {
        "Null": {
            "iotevents:keyValue": "false"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "iotevents>CreateAlarmModel",
        "iotevents:TagResource"
    ],
    "Resource": "*",
    "Condition": {
        "Null": {
            "aws:RequestTag/iotsitewisemonitor": "false"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "iotevents:UpdateAlarmModel",
        "iotevents>DeleteAlarmModel"
    ],
    "Resource": "*",
    "Condition": {
        "Null": {

```

```

        "aws:ResourceTag/iotsitewisemonitor": "false"
    }
}
},
{
    "Effect": "Allow",
    "Action": [
        "iam:PassRole"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": [
                "iotevents.amazonaws.com"
            ]
        }
    }
}
]
}

```

Pencegahan confused deputy lintas layanan

Masalah deputy yang bingung adalah masalah keamanan di mana entitas yang tidak memiliki izin untuk melakukan tindakan dapat memaksa entitas yang lebih istimewa untuk melakukan tindakan. Pada tahun AWS, peniruan lintas layanan dapat mengakibatkan masalah wakil yang membingungkan. Peniruan identitas lintas layanan dapat terjadi ketika satu layanan (layanan yang dipanggil) memanggil layanan lain (layanan yang dipanggil). Layanan panggilan dapat dimanipulasi untuk menggunakan izinnya untuk bertindak atas sumber daya pelanggan lain dengan cara yang seharusnya tidak memiliki izin untuk mengakses. Untuk mencegah hal ini, AWS menyediakan alat yang membantu Anda melindungi data untuk semua layanan dengan pengguna utama layanan yang telah diberi akses ke sumber daya di akun Anda.

Sebaiknya gunakan kunci konteks kondisi [aws:SourceAccount](#) global [aws:SourceArn](#) dan dalam kebijakan sumber daya untuk membatasi izin yang AWS IoT SiteWise memberikan layanan lain ke sumber daya. Jika `aws:SourceArn` nilai tidak berisi ID akun, seperti bucket Amazon S3 Amazon Resource Name (ARN), Anda harus menggunakan kedua kunci konteks kondisi global untuk membatasi izin. Jika Anda menggunakan kunci konteks kondisi global dan nilai `aws:SourceArn` berisi ID akun, nilai `aws:SourceAccount` dan akun dalam nilai `aws:SourceArn` harus menggunakan ID akun yang sama saat digunakan dalam pernyataan kebijakan yang sama.

- Gunakan `aws:SourceArn` jika Anda hanya ingin satu sumber daya dikaitkan dengan akses lintas layanan.
- Gunakan `aws:SourceAccount` jika Anda ingin mengizinkan sumber daya apa pun di akun tersebut dikaitkan dengan penggunaan lintas layanan.

Nilai `aws:SourceArn` harus menjadi sumber daya AWS IoT SiteWise pelanggan yang terkait dengan `sts:AssumeRole` permintaan.

Cara paling efektif untuk melindungi dari masalah confused deputy adalah dengan menggunakan kunci konteks kondisi global `aws:SourceArn` dengan ARN lengkap sumber daya. Jika Anda tidak mengetahui ARN lengkap sumber daya atau jika Anda menentukan beberapa sumber daya, gunakan kunci kondisi konteks `aws:SourceArn` global dengan wildcard (*) untuk bagian ARN yang tidak diketahui. Misalnya, `arn:aws:servicename:*:123456789012:*`.

Example — Pencegahan Deputi Bingung

Contoh berikut menunjukkan bagaimana Anda dapat menggunakan kunci konteks kondisi `aws:SourceAccount` global `aws:SourceArn` dan AWS IoT SiteWise untuk mencegah masalah wakil yang membingungkan.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "iotsitewise.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Resource": [
      "arn:aws:iotsitewise::ResourceName/*"
    ],
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:iotsitewise:*:123456789012:*"
      },
      "StringEquals": {
        "aws:SourceAccount": "123456789012"
      }
    }
  }
}
```

```
}
```

Memecahkan masalah AWS IoT SiteWise identitas dan akses

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan AWS IoT SiteWise dan AWS Identity and Access Management (IAM).

Topik

- [Saya tidak berwenang untuk melakukan tindakan di AWS IoT SiteWise](#)
- [Saya tidak berwenang untuk melakukan iam:PassRole](#)
- [Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS IoT SiteWise sumber daya saya](#)

Saya tidak berwenang untuk melakukan tindakan di AWS IoT SiteWise

Jika AWS Management Console memberitahu Anda bahwa Anda tidak berwenang untuk melakukan tindakan, maka Anda harus menghubungi administrator Anda untuk bantuan. Administrator adalah orang yang memberikan nama pengguna dan kata sandi kepada Anda.

Contoh kesalahan berikut terjadi ketika pengguna mateojackson IAM mencoba menggunakan konsol untuk melihat detail tentang aset tetapi tidak memiliki `iotsitewise:DescribeAsset` izin.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
iotsitewise:DescribeAsset on resource: a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Dalam hal ini, Mateo meminta administratornya untuk memperbarui kebijakannya untuk memungkinkannya mengakses sumber daya aset dengan ID `a1b2c3d4-5678-90ab-cdef-22222EXAMPLE` menggunakan `iotsitewise:DescribeAsset` tindakan tersebut.

Saya tidak berwenang untuk melakukan **iam:PassRole**

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran AWS IoT SiteWise.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol tersebut untuk melakukan tindakan di AWS IoT SiteWise. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS IoT SiteWise sumber daya saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau pengguna di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACL), Anda dapat menggunakan kebijakan tersebut untuk memberi pengguna akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa hal berikut:

- Untuk mempelajari apakah AWS IoT SiteWise mendukung fitur ini, lihat [Bagaimana AWS IoT SiteWise bekerja dengan IAM](#).
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat [Menyediakan akses ke pengguna IAM di pengguna lain Akun AWS yang Anda miliki](#) di Panduan Pengguna IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat [Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga](#) dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat [Memberikan akses kepada pengguna eksternal yang sah \(federasi identitas\)](#) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara penggunaan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat [Perbedaan antara peran IAM dan kebijakan berbasis sumber daya](#) di Panduan Pengguna IAM.

Validasi kepatuhan untuk AWS IoT SiteWise

AWS IoT SiteWise tidak dalam lingkup program AWS kepatuhan apa pun.

Untuk daftar AWS layanan dalam lingkup program kepatuhan tertentu, lihat [AWS Layanan dalam Lingkup oleh AWS Layanan Program Kepatuhan](#) . Untuk informasi umum, lihat [Program AWS Kepatuhan Program AWS](#) .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat [Mengunduh laporan di AWS Artifact](#) .

Tanggung jawab kepatuhan Anda saat menggunakan AWS IoT SiteWise ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- [Panduan Memulai Cepat Keamanan dan Kepatuhan Panduan](#) Keamanan dan Kepatuhan — Panduan penerapan ini membahas pertimbangan arsitektur dan memberikan langkah-langkah untuk menerapkan lingkungan dasar yang berfokus pada keamanan dan kepatuhan. AWS
- [Arsitektur untuk Whitepaper Keamanan dan Kepatuhan HIPAA — Whitepaper](#) ini menjelaskan bagaimana perusahaan dapat menggunakan untuk membuat aplikasi yang sesuai dengan HIPAA. AWS
- [AWS Sumber Daya AWS](#) — Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- [Mengevaluasi sumber daya dengan aturan](#) dalam Panduan AWS Config Pengembang — AWS Config Layanan menilai seberapa baik konfigurasi sumber daya Anda mematuhi praktik internal, pedoman industri, dan peraturan.
- [AWS Security Hub](#)— AWS Layanan ini memberikan pandangan komprehensif tentang keadaan keamanan Anda di dalamnya AWS yang membantu Anda memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik.
- [Sepuluh aturan emas keamanan untuk solusi IoT Industri](#) - Posting blog ini memperkenalkan sepuluh aturan emas yang membantu mengamankan sistem kontrol industri (ICS) Anda, Internet of Things industri (IIoT), dan lingkungan cloud.
- [Praktik Terbaik Keamanan untuk PL Manufaktur](#) — Whitepaper ini menjelaskan praktik terbaik keamanan untuk merancang, menerapkan, dan merancang beban kerja manufaktur hybrid lokal ini untuk Cloud. AWS

Ketahanan di AWS IoT SiteWise

Infrastruktur AWS global dibangun di sekitar AWS Wilayah dan Zona Ketersediaan. AWS Wilayah menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi, yang terhubung dengan latensi rendah, throughput tinggi, dan jaringan yang sangat redundan. Dengan Availability Zone, Anda dapat mendesain dan mengoperasikan aplikasi dan basis data yang secara otomatis mengalami kegagalan di antara zona tanpa gangguan. Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

AWS IoT SiteWise sepenuhnya dikelola dan menggunakan AWS layanan yang sangat tersedia dan tahan lama, seperti Amazon S3 dan Amazon EC2. Untuk memastikan ketersediaan jika terjadi gangguan zona ketersediaan, AWS IoT SiteWise beroperasi di beberapa zona ketersediaan.

Untuk informasi selengkapnya tentang AWS Wilayah dan Availability Zone, lihat [Infrastruktur AWS Global](#).

Selain infrastruktur AWS global, AWS IoT SiteWise menawarkan beberapa fitur untuk membantu mendukung ketahanan data dan kebutuhan cadangan Anda:

- Anda dapat mempublikasikan pembaruan nilai properti AWS IoT Core melalui pesan MQTT, lalu mengonfigurasi aturan untuk menindaklanjuti data tersebut. Dengan fitur ini, Anda dapat mencadangkan data di AWS layanan lain seperti Amazon S3 dan Amazon DynamoDB. Lihat informasi yang lebih lengkap di [Berinteraksi dengan layanan lain AWS](#) dan [Ekspor data ke Amazon S3 dengan pemberitahuan properti aset](#).
- Anda dapat menggunakan AWS IoT SiteWise Get * API untuk mengambil dan membuat cadangan data properti aset historis. Untuk informasi selengkapnya, lihat [Menanyakan nilai properti aset historis](#).
- Anda dapat menggunakan AWS IoT SiteWise Describe * API untuk mengambil definisi sumber daya Anda, seperti aset dan model. Anda dapat membuat cadangan definisi ini dan kemudian menggunakannya untuk membuat ulang sumber daya Anda. Untuk informasi lebih lanjut, lihat [Referensi API AWS IoT SiteWise](#).

Keamanan infrastruktur di AWS IoT SiteWise

Sebagai layanan terkelola, AWS IoT SiteWise dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan AWS keamanan dan cara AWS melindungi infrastruktur, lihat [Keamanan](#)

[AWS Cloud](#). Untuk mendesain AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat [Perlindungan Infrastruktur dalam Kerangka Kerja](#) yang AWS Diarsiteksikan dengan Baik Pilar Keamanan.

Anda menggunakan panggilan API yang AWS dipublikasikan untuk mengakses AWS IoT SiteWise melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS). Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Sandi cocok dengan sistem kerahasiaan maju sempurna (perfect forward secrecy, PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Selain itu, permintaan harus ditandatangani dengan menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan pengguna utama IAM. Atau Anda dapat menggunakan [AWS Security Token Service](#) (AWS STS) untuk menghasilkan kredensial keamanan sementara untuk menandatangani permintaan.

SiteWise Edge gateway, yang berjalan AWS IoT Greengrass, menggunakan sertifikat X.509 dan kunci kriptografi untuk menghubungkan dan mengautentikasi ke Cloud. AWS Untuk informasi selengkapnya, lihat [Autentikasi dan otorisasi perangkat AWS IoT Greengrass](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

Konfigurasi dan analisis kerentanan

Armada IoT dapat terdiri dari sejumlah besar perangkat yang memiliki kemampuan beragam, berumur panjang, dan didistribusikan secara geografis. Karakteristik ini membuat pengaturan armada kompleks dan rawan kesalahan. Karena perangkat biasanya memiliki daya pemrosesan, memori, dan penyimpanan yang terbatas, mereka tidak selalu dapat mendukung enkripsi dan langkah-langkah keamanan lainnya. Selain itu, perangkat sering menggunakan perangkat lunak dengan kerentanan yang diketahui. Faktor-faktor ini membuat armada IoT menjadi target yang menarik bagi peretas dan menyulitkan untuk mengamankan armada perangkat Anda secara berkelanjutan.

AWS IoT Device Defender mengatasi tantangan ini dengan menyediakan alat untuk mengidentifikasi masalah keamanan dan penyimpangan dari praktik terbaik. Gunakan AWS IoT Device Defender untuk menganalisis, mengaudit, dan memantau perangkat yang terhubung untuk mendeteksi perilaku abnormal, dan mengurangi risiko keamanan. AWS IoT Device Defender dapat mengaudit armada perangkat untuk memastikan mereka mematuhi praktik terbaik keamanan dan mendeteksi

perilaku abnormal pada perangkat. Hal ini memungkinkan untuk menerapkan kebijakan keamanan yang konsisten di seluruh armada AWS IoT perangkat Anda dan merespons dengan cepat ketika perangkat dikompromikan. Untuk informasi lebih lanjut, lihat [AWS IoT Device Defender](#) dalam Panduan Pengembang AWS IoT .

Jika Anda menggunakan gateway SiteWise Edge untuk menyerap data ke layanan, Anda bertanggung jawab untuk mengonfigurasi dan memelihara lingkungan gateway SiteWise Edge Anda. Tanggung jawab ini mencakup peningkatan ke versi terbaru dari perangkat lunak sistem, perangkat AWS IoT Greengrass lunak, dan AWS IoT SiteWise konektor SiteWise Edge gateway. Untuk informasi selengkapnya, lihat [Mengkonfigurasi AWS IoT Greengrass inti](#) di Panduan AWS IoT Greengrass Version 1 Pengembang dan [Memutakhirkan konektor](#).

Titik akhir VPC

Endpoint VPC antarmuka membuat koneksi pribadi antara virtual private cloud (VPC) Anda dan AWS IoT SiteWise [AWS PrivateLink](#) memberdayakan titik akhir antarmuka, memungkinkan akses pribadi ke operasi AWS IoT SiteWise API. Anda dapat melewati kebutuhan akan gateway internet, perangkat NAT, koneksi VPN, atau AWS Direct Connect Instans di VPC Anda tidak memerlukan alamat IP publik untuk berkomunikasi AWS IoT SiteWise dengan operasi API. Lalu lintas antara VPC Anda dan AWS IoT SiteWise tidak meninggalkan jaringan. AWS

Setiap titik akhir antarmuka diwakili oleh satu atau lebih [antarmuka jaringan elastis](#) dalam subnet Anda.

Sebelum menyiapkan titik akhir VPC antarmuka AWS IoT SiteWise, tinjau [properti dan batasan titik akhir Antarmuka di](#) Panduan Pengguna Amazon VPC.

Untuk informasi selengkapnya, lihat [Antarmuka VPC endpoint \(AWS PrivateLink\)](#) dalam Panduan Pengguna Amazon VPC.

Operasi API yang didukung untuk titik akhir VPC

AWS IoT SiteWise mendukung panggilan ke operasi AWS IoT SiteWise API berikut dari VPC Anda:

- Untuk semua operasi API bidang data, gunakan titik akhir berikut: Ganti *region* dengan Wilayah AWS

```
data.iotsitewise.region.amazonaws.com
```

Operasi API bidang data meliputi:


- [BatchGetAssetPropertyValue](#)
 - [BatchGetAssetPropertyValueHistory](#)
 - [BatchPutAssetPropertyValue](#)
 - [GetAssetPropertyAggregates](#)
 - [GetAssetPropertyValue](#)
 - [GetAssetPropertyValueHistory](#)
 - [GetInterpolatedAssetPropertyValues](#)
- Untuk operasi API bidang kontrol yang Anda gunakan untuk mengelola model aset, aset, gateway SiteWise Edge, tag, dan konfigurasi akun, gunakan titik akhir berikut. Ganti *region* dengan Anda Wilayah AWS.

```
api.iotsitewise.region.amazonaws.com
```

Operasi API bidang kontrol yang didukung mencakup hal-hal berikut:

- [AssociateAssets](#)
- [CreateAsset](#)
- [CreateAssetModel](#)
- [DeleteAsset](#)
- [DeleteAssetModel](#)
- [DeleteDashboard](#)
- [DescribeAsset](#)
- [DescribeAssetModel](#)
- [DescribeAssetProperty](#)
- [DescribeDashboard](#)
- [DescribeLoggingOptions](#)
- [DisassociateAssets](#)
- [ListAssetModels](#)
- [ListAssetRelationships](#)
- [ListAssets](#)

- [PutLoggingOptions](#)
- [UpdateAsset](#)
- [UpdateAssetModel](#)
- [UpdateAssetProperty](#)
- [CreateGateway](#)
- [DeleteGateway](#)
- [DescribeDefaultEncryptionConfiguration](#)
- [DescribeGateway](#)
- [DescribeGatewayCapabilityConfiguration](#)
- [DescribeStorageConfiguration](#)
- [ListGateways](#)
- [ListTagsForResource](#)
- [UpdateGateway](#)
- [UpdateGatewayCapabilityConfiguration](#)
- [PutDefaultEncryptionConfiguration](#)
- [PutStorageConfiguration](#)
- [TagResource](#)
- [UntagResource](#)

 Note

Titik akhir VPC antarmuka untuk operasi API bidang kontrol saat ini tidak mendukung panggilan ke operasi SiteWise Monitor API berikut:

- [BatchAssociateProjectAssets](#)
- [BatchDisassociateProjectAssets](#)
- [CreateAccessPolicy](#)
- [CreateDashboard](#)
- [CreatePortal](#)
- [CreateProject](#)
- [DeleteAccessPolicy](#)

- [DeleteProject](#)
- [DescribeAccessPolicy](#)
- [DescribePortal](#)
- [DescribeProject](#)
- [ListAccessPolicies](#)
- [ListDashboards](#)
- [ListPortals](#)
- [ListProjects](#)
- [ListProjectAssets](#)
- [UpdateAccessPolicy](#)
- [UpdateDashboard](#)
- [UpdatePortal](#)
- [UpdateProject](#)

Buat VPC endpoint antarmuka untuk AWS IoT SiteWise

Untuk membuat titik akhir VPC untuk AWS IoT SiteWise layanan, gunakan konsol VPC Amazon atau (). AWS Command Line Interface AWS CLI Untuk informasi selengkapnya, lihat [Membuat titik akhir antarmuka](#) dalam Panduan Pengguna Amazon VPC.

Buat titik akhir VPC AWS IoT SiteWise dengan menggunakan salah satu nama layanan berikut:

- Untuk operasi API bidang data, gunakan nama layanan berikut:

```
com.amazonaws.region.iotsitewise.data
```

- Untuk operasi API bidang kontrol, gunakan nama layanan berikut:

```
com.amazonaws.region.iotsitewise.api
```

Mengakses AWS IoT SiteWise melalui antarmuka VPC endpoint

Saat Anda membuat titik akhir antarmuka, kami menghasilkan nama host DNS khusus titik akhir yang dapat Anda gunakan untuk berkomunikasi. AWS IoT SiteWise Opsi DNS pribadi diaktifkan secara

default. Untuk informasi selengkapnya, lihat [Menggunakan zona yang dihosting pribadi](#) di Panduan Pengguna Amazon VPC.

Jika Anda mengaktifkan DNS pribadi untuk titik akhir, Anda dapat membuat permintaan API AWS IoT SiteWise melalui salah satu titik akhir VPC berikut.

- Untuk operasi API bidang data, gunakan titik akhir berikut: Ganti *wilayah* dengan *wilayah* Anda Wilayah AWS.

```
data.iotsitewise.region.amazonaws.com
```

- Untuk operasi API bidang kontrol, gunakan titik akhir berikut: Ganti *wilayah* dengan *wilayah* Anda Wilayah AWS.

```
api.iotsitewise.region.amazonaws.com
```

Jika Anda menonaktifkan DNS pribadi untuk titik akhir, Anda harus melakukan hal berikut untuk mengakses AWS IoT SiteWise melalui titik akhir:

1. Tentukan url titik akhir VPC dalam permintaan API.

- Untuk operasi API bidang data, gunakan url titik akhir berikut. Ganti *vpc-endpoint-id* dan *wilayah* dengan ID titik akhir VPC dan Wilayah Anda.

```
vpc-endpoint-id.data.iotsitewise.region.vpce.amazonaws.com
```

- Untuk operasi API bidang kontrol, gunakan url titik akhir berikut. Ganti *vpc-endpoint-id* dan *wilayah* dengan ID titik akhir VPC dan Wilayah Anda.

```
vpc-endpoint-id.api.iotsitewise.region.vpce.amazonaws.com
```

2. Nonaktifkan injeksi awalan host. AWS SDK AWS CLI dan menambahkan titik akhir layanan dengan berbagai awalan host saat Anda memanggil setiap operasi API. Fitur ini menyebabkan AWS CLI dan AWS SDK menghasilkan URL yang tidak valid AWS IoT SiteWise saat Anda menentukan titik akhir VPC.

Important

Anda tidak dapat menonaktifkan injeksi awalan host di AWS CLI atau file. AWS Tools for PowerShell Ini berarti bahwa jika Anda menonaktifkan DNS pribadi, maka Anda tidak

dapat menggunakan alat ini untuk mengakses AWS IoT SiteWise melalui titik akhir VPC. Aktifkan DNS pribadi untuk menggunakan AWS CLI atau AWS Tools for PowerShell untuk mengakses AWS IoT SiteWise melalui titik akhir.

Untuk informasi selengkapnya tentang cara menonaktifkan injeksi awalan host di AWS SDK, lihat bagian dokumentasi berikut untuk setiap SDK:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java](#)
- [AWS SDK for Java 2.x](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for .NET](#)
- [AWS SDK for PHP](#)
- [AWS SDK for Python \(Boto3\)](#)
- [AWS SDK for Ruby](#)

Untuk informasi selengkapnya, lihat [Mengakses layanan melalui titik akhir antarmuka](#) dalam Panduan Pengguna Amazon VPC.

Membuat kebijakan titik akhir VPC untuk AWS IoT SiteWise

Anda dapat melampirkan kebijakan titik akhir ke VPC endpoint yang mengendalikan akses ke AWS IoT SiteWise. Kebijakan menentukan informasi berikut ini:

- Kepala sekolah yang dapat melakukan operasi.
- Operasi yang dapat dilakukan.
- Sumber daya di mana operasi dapat dilakukan.

Untuk informasi selengkapnya, lihat [Mengontrol akses ke layanan dengan titik akhir VPC](#) dalam Panduan Pengguna Amazon VPC.

Contoh: Kebijakan titik akhir VPC untuk tindakan AWS IoT SiteWise

Membuat kebijakan titik akhir VPC

Berikut ini adalah contoh kebijakan endpoint untuk AWS IoT SiteWise. Saat dilampirkan ke titik akhir, kebijakan ini memberikan akses ke AWS IoT SiteWise tindakan yang tercantum untuk pengguna di Akun AWS **123456789012** *iotsitewiseadmin* pada aset yang ditentukan.

```
{
  "Statement": [
    {
      "Action": [
        "iotsitewise:CreateAsset",
        "iotsitewise:ListGateways",
        "iotsitewise:ListTagsForResource"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:iotsitewise:us-west-2:123456789012:asset/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "Principal": {
        "AWS": [
          "123456789012:user/iotsitewiseadmin"
        ]
      }
    }
  ]
}
```

Praktik terbaik keamanan untuk AWS IoT SiteWise

Topik ini berisi praktik terbaik keamanan untuk AWS IoT SiteWise.

Menggunakan kredensi otentikasi di server OPC-UA Anda

Memerlukan kredensi otentikasi untuk terhubung ke server OPC-UA Anda. Konsultasikan dokumentasi untuk server Anda untuk melakukannya. Kemudian, untuk memungkinkan gateway SiteWise Edge Anda terhubung ke server OPC-UA Anda, tambahkan rahasia otentikasi server ke gateway Edge Anda SiteWise . Untuk informasi selengkapnya, lihat [Mengkonfigurasi otentikasi sumber](#).

Gunakan mode komunikasi terenkripsi untuk server OPC-UA Anda

Pilih mode keamanan pesan terenkripsi yang tidak digunakan lagi saat Anda mengonfigurasi sumber OPC-UA untuk gateway Edge Anda. SiteWise Ini membantu mengamankan data industri Anda saat

bergerak dari server OPC-UA Anda ke gateway SiteWise Edge. Lihat informasi yang lebih lengkap di [Data dalam transit melalui jaringan lokal](#) dan [Mengkonfigurasi sumber data](#).

Perbarui komponen Anda

Jika Anda menggunakan gateway SiteWise Edge untuk menyerap data ke layanan, Anda bertanggung jawab untuk mengonfigurasi dan memelihara lingkungan gateway Edge Anda SiteWise. Tanggung jawab ini mencakup peningkatan ke versi terbaru dari perangkat lunak, perangkat AWS IoT Greengrass lunak, dan konektor sistem gateway.

Note

Konektor AWS IoT SiteWise Edge menyimpan rahasia di sistem file Anda. Rahasia ini mengontrol siapa yang dapat melihat data yang di-cache dalam gateway SiteWise Edge Anda. Sangat disarankan agar Anda mengaktifkan enkripsi disk atau file-system untuk sistem yang menjalankan gateway SiteWise Edge Anda.

Enkripsi sistem file gateway SiteWise Edge Anda

Enkripsi dan amankan gateway SiteWise Edge Anda, sehingga data industri Anda aman saat bergerak melalui gateway SiteWise Edge. Jika gateway SiteWise Edge Anda memiliki modul keamanan perangkat keras, Anda dapat mengonfigurasi AWS IoT Greengrass untuk mengamankan gateway SiteWise Edge Anda. Untuk informasi selengkapnya, lihat [Integrasi keamanan perangkat keras](#) di Panduan AWS IoT Greengrass Version 1 Pengembang. Jika tidak, lihat dokumentasi untuk sistem operasi Anda untuk mempelajari cara mengenkripsi dan mengamankan sistem file Anda.

Akses aman ke konfigurasi tepi Anda

Jangan bagikan kata sandi aplikasi konsol tepi atau kata sandi aplikasi SiteWise Monitor Anda. Jangan letakkan kata sandi ini di tempat di mana siapa pun dapat melihatnya. Terapkan kebijakan rotasi kata sandi yang sehat dengan mengonfigurasi kedaluwarsa yang sesuai untuk kata sandi Anda.

Grant SiteWise Monitor pengguna izin minimum yang mungkin

Ikuti prinsip hak istimewa terkecil dengan menggunakan set minimum izin kebijakan akses untuk pengguna portal Anda.

- Saat Anda membuat portal, tentukan peran yang memungkinkan kumpulan aset minimum yang diperlukan untuk portal tersebut. Untuk informasi selengkapnya, lihat [Menggunakan peran layanan untuk AWS IoT SiteWise Monitor](#).
- Saat Anda dan administrator portal membuat dan berbagi proyek, gunakan kumpulan aset minimum yang diperlukan untuk proyek tersebut.
- Ketika identitas tidak lagi membutuhkan akses ke portal atau proyek, hapus mereka dari sumber daya itu. Jika identitas itu tidak lagi berlaku untuk organisasi Anda, hapus identitas itu dari toko identitas Anda.

Praktik terbaik prinsip terkecil juga berlaku untuk peran IAM. Untuk informasi selengkapnya, lihat [Praktik terbaik kebijakan](#).

Jangan mengekspos informasi sensitif

Anda harus mencegah pencatatan kredensial dan informasi sensitif lainnya, seperti informasi identitas pribadi (PII). Kami menyarankan Anda menerapkan perlindungan berikut meskipun akses ke log lokal di gateway SiteWise Edge memerlukan hak akses root dan akses ke CloudWatch Log memerlukan izin IAM.

- Jangan gunakan informasi sensitif dalam nama, deskripsi, atau properti aset atau model Anda.
- Jangan gunakan informasi sensitif di gateway SiteWise Edge atau nama sumber.
- Jangan gunakan informasi sensitif dalam nama atau deskripsi portal, proyek, atau dasbor Anda.

Ikuti praktik terbaik AWS IoT Greengrass keamanan

Ikuti praktik terbaik AWS IoT Greengrass keamanan untuk gateway SiteWise Edge Anda. Untuk informasi selengkapnya, lihat [Praktik terbaik keamanan](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

Lihat juga

- [Praktik terbaik keamanan](#) di Panduan AWS IoT Pengembang
- [Sepuluh aturan emas keamanan untuk solusi IoT Industri](#)

Penebangan dan pemantauan di AWS IoT SiteWise

Pemantauan adalah bagian penting dari menjaga keandalan, ketersediaan, dan kinerja AWS IoT SiteWise dan AWS solusi Anda yang lain. AWS IoT SiteWise mendukung alat pemantauan berikut untuk menonton layanan, melaporkan ketika ada sesuatu yang salah, dan mengambil tindakan otomatis bila perlu:

- Amazon CloudWatch memantau AWS sumber daya Anda dan aplikasi yang Anda jalankan AWS secara real time. Kumpulkan dan lacak metrik, buat dasbor khusus, dan setel alarm yang memberi tahu Anda atau mengambil tindakan saat metrik tertentu mencapai ambang batas tertentu. Misalnya, Anda dapat CloudWatch melacak penggunaan CPU atau metrik lain dari instans Amazon EC2 Anda dan secara otomatis meluncurkan instans baru bila diperlukan. Untuk informasi selengkapnya, lihat [Panduan CloudWatch Pengguna Amazon](#).
- Amazon CloudWatch Logs memantau, menyimpan, dan mengakses file log Anda dari gateway SiteWise Edge CloudTrail, dan sumber lainnya. CloudWatch Log dapat memantau informasi dalam file log dan memberi tahu Anda ketika ambang batas tertentu terpenuhi. Anda juga dapat mengarsipkan data log dalam penyimpanan yang sangat durabel. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon CloudWatch Logs](#).
- AWS CloudTrail menangkap panggilan API dan peristiwa terkait yang dibuat oleh atau atas nama AWS akun Anda. Kemudian CloudTrail mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Anda dapat mengidentifikasi pengguna dan akun mana yang dipanggil AWS, alamat IP sumber dari mana panggilan dilakukan, dan kapan panggilan terjadi. Untuk informasi selengkapnya, lihat [Panduan Pengguna AWS CloudTrail](#).

Topik

- [Pemantauan dengan Amazon CloudWatch Logs](#)
- [Memantau log gateway SiteWise Edge](#)
- [Pemantauan AWS IoT SiteWise dengan CloudWatch metrik Amazon](#)
- [Logging panggilan AWS IoT SiteWise API dengan AWS CloudTrail](#)

Pemantauan dengan Amazon CloudWatch Logs

Konfigurasi AWS IoT SiteWise untuk mencatat informasi ke CloudWatch Log untuk memantau dan memecahkan masalah layanan.

Saat Anda menggunakan AWS IoT SiteWise konsol, AWS IoT SiteWise buat peran terkait layanan yang memungkinkan layanan mencatat informasi atas nama Anda. Jika Anda tidak menggunakan AWS IoT SiteWise konsol, Anda harus membuat peran terkait layanan secara manual untuk menerima log. Untuk informasi selengkapnya, lihat [Membuat peran yang terkait dengan layanan untuk AWS IoT SiteWise](#).

Anda harus memiliki kebijakan sumber daya yang memungkinkan AWS IoT SiteWise untuk menempatkan peristiwa log ke CloudWatch aliran. Untuk membuat dan memperbarui kebijakan sumber daya untuk CloudWatch Log, jalankan perintah berikut. Ganti *logging-policy-name* dengan nama kebijakan yang akan dibuat.

```
aws logs put-resource-policy --policy-name logging-policy-name --policy-document "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Sid\": \"IoTSiteWiseToCloudWatchLogs\", \"Effect\": \"Allow\", \"Principal\": { \"Service\": [ \"iotsitewise.amazonaws.com\" ] }, \"Action\": \"logs:PutLogEvents\", \"Resource\": \"*\" } ] }"
```

CloudWatch Log juga mendukung kunci konteks SourceAccount kondisi [aws: SourceArn](#) dan [aws:](#). Kunci konteks kondisi ini bersifat opsional.

Untuk membuat atau memperbarui kebijakan sumber daya yang memungkinkan AWS IoT SiteWise untuk hanya menempatkan log yang terkait dengan AWS IoT SiteWise sumber daya tertentu ke dalam CloudWatch aliran, jalankan perintah dan lakukan hal berikut:

- Ganti *logging-policy-name* dengan nama kebijakan yang akan dibuat.
- Ganti *Source-ARN* dengan ARN AWS IoT SiteWise sumber daya Anda, seperti model aset atau aset. Untuk menemukan ARN untuk setiap jenis sumber daya, lihat Jenis AWS IoT SiteWise sumber daya yang [ditentukan oleh AWS IoT SiteWise](#) dalam Referensi Otorisasi Layanan.
- Ganti *Account-ID* dengan ID AWS akun yang terkait dengan sumber daya yang ditentukan AWS IoT SiteWise .

```
aws logs put-resource-policy --policy-name logging-policy-name --policy-document "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Sid\": \"IoTSiteWiseToCloudWatchLogs\", \"Effect\": \"Allow\", \"Principal\": { \"Service\": [ \"iotsitewise.amazonaws.com\" ] }, \"Action\": \"logs:PutLogEvents\", \"Resource\": \"*\", \"Condition\": { \"StringLike\": { \"aws:SourceArn\": [ \"source-ARN\" ], \"aws:SourceAccount\": [ \"account-ID\" ] } } } ] }"
```

Secara default, AWS IoT SiteWise tidak mencatat informasi ke CloudWatch Log. Untuk mengaktifkan logging, pilih level logging selain Disabled (OFF). AWS IoT SiteWise mendukung tingkat logging berikut:

- OFF— Logging dimatikan.
- ERROR— Kesalahan dicatat.
- INFO— Kesalahan dan pesan informasi dicatat.

Anda dapat mengonfigurasi gateway SiteWise Edge untuk mencatat informasi ke CloudWatch Log melalui AWS IoT Greengrass. Untuk informasi selengkapnya, lihat [Memantau log gateway SiteWise Edge](#).

Anda juga dapat mengonfigurasi AWS IoT Core untuk mencatat informasi ke CloudWatch Log jika Anda memecahkan masalah tindakan aturan AWS IoT SiteWise. Untuk informasi selengkapnya, lihat [Memecahkan masalah tindakan aturan AWS IoT SiteWise](#).

Daftar Isi

- [Mengelola masuk AWS IoT SiteWise](#)
 - [Menemukan tingkat logging Anda](#)
 - [Mengubah level logging Anda](#)
- [Contoh: entri file AWS IoT SiteWise log](#)

Mengelola masuk AWS IoT SiteWise

Gunakan AWS IoT SiteWise konsol atau AWS CLI untuk tugas konfigurasi logging berikut.

Menemukan tingkat logging Anda

Console

Gunakan prosedur berikut untuk menemukan level logging Anda saat ini di AWS IoT SiteWise konsol.

Untuk menemukan level AWS IoT SiteWise logging Anda saat ini

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Opsi Logging.

Status logging saat ini muncul di bawah status Logging. Jika logging diaktifkan, level logging saat ini muncul di bawah Level verbositas.

AWS CLI

Jalankan perintah berikut untuk menemukan level AWS IoT SiteWise logging Anda saat ini dengan file AWS CLI.

```
aws iotsitewise describe-logging-options
```

Operasi mengembalikan respons yang berisi tingkat logging Anda dalam format berikut.

```
{
  "loggingOptions": {
    "level": "String"
  }
}
```

Mengubah level logging Anda

Gunakan prosedur berikut untuk mengubah level logging Anda di AWS IoT SiteWise konsol atau menggunakan AWS CLI.

Console

Untuk mengubah level AWS IoT SiteWise logging Anda

1. Navigasikan ke [konsol AWS IoT SiteWise](#) tersebut.
2. Di panel navigasi kiri, pilih Opsi Logging.
3. Pilih Edit.
4. Pilih Level verbositas untuk diaktifkan.
5. Pilih Simpan.

AWS CLI

Jalankan AWS CLI perintah berikut untuk mengubah level AWS IoT SiteWise logging Anda. Ganti *logging-level* dengan *level* logging yang Anda inginkan.

```
aws iotsitewise put-logging-options --logging-options level=logging-level
```

Contoh: entri file AWS IoT SiteWise log

Setiap entri AWS IoT SiteWise log mencakup informasi peristiwa dan sumber daya yang relevan untuk acara tersebut, sehingga Anda dapat memahami dan menganalisis data log.

Contoh berikut menunjukkan entri CloudWatch AWS IoT SiteWise Log yang mencatat saat Anda berhasil membuat model aset.

```
{
  "eventTime": "2020-05-05T00:10:22.902Z",
  "logLevel": "INFO",
  "eventType": "AssetModelCreationSuccess",
  "message": "Successfully created asset model.",
  "resources": {
    "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
  }
}
```

Memantau log gateway SiteWise Edge

Anda dapat mengonfigurasi gateway AWS IoT SiteWise Edge Anda untuk mencatat informasi ke Amazon CloudWatch Logs atau sistem file lokal.

Topik

- [Menggunakan CloudWatch Log Amazon](#)
- [Menggunakan log layanan](#)
- [Menggunakan log peristiwa](#)

Menggunakan CloudWatch Log Amazon

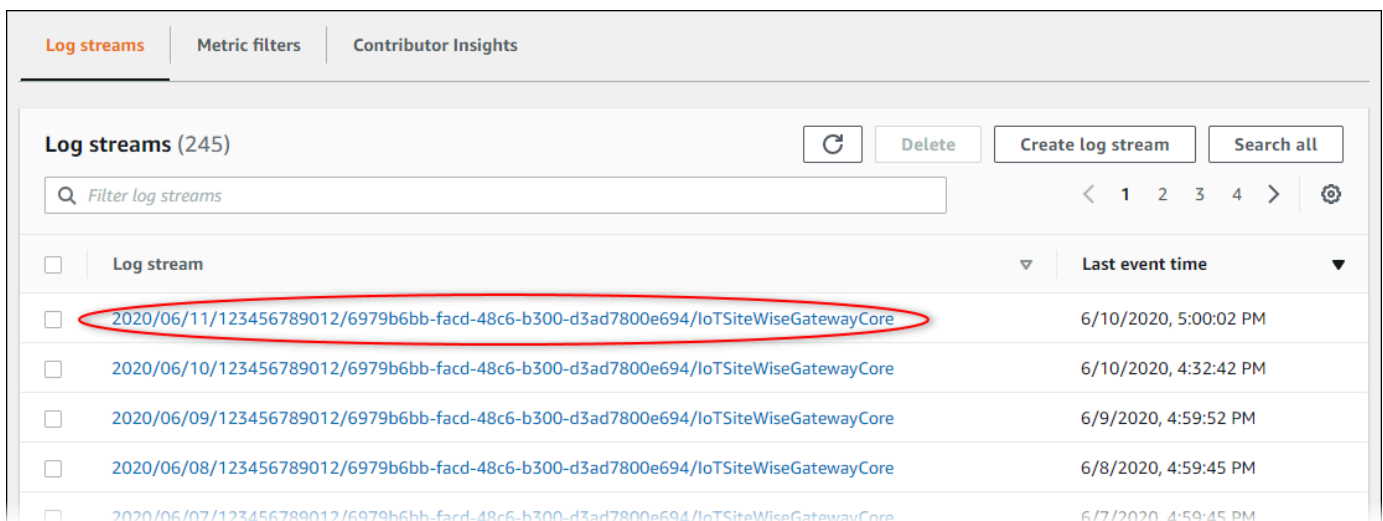
Anda dapat mengonfigurasi gateway SiteWise Edge Anda untuk mengirim log ke CloudWatch Log. Untuk informasi selengkapnya, lihat [Mengaktifkan pencatatan untuk CloudWatch Log](#) di Panduan AWS IoT Greengrass Version 2 Pengembang.

Untuk mengkonfigurasi dan mengakses CloudWatch Log (Konsol)

1. Navigasikan ke [konsol CloudWatch](#) tersebut.
2. Pada panel navigasi, pilih Grup log.
3. Anda dapat menemukan log AWS IoT SiteWise komponen di grup log berikut:
 - `/aws/greengrass/UserComponent/region/aws.iot.SiteWiseEdgeCollectorOpcua`— Log untuk komponen gateway SiteWise Edge yang mengumpulkan data dari sumber OPC-UA gateway SiteWise Edge.
 - `/aws/greengrass/UserComponent/region/aws.iot.SiteWiseEdgePublisher`— Log untuk komponen gateway SiteWise Edge yang menerbitkan aliran data OPC-UA ke. AWS IoT SiteWise

Pilih grup log untuk fungsi yang akan di-debug.

4. Pilih aliran log yang memiliki nama yang diakhiri dengan nama AWS IoT Greengrass grup Anda. Secara default, CloudWatch menampilkan aliran log terbaru terlebih dahulu.



5. Untuk menampilkan log dari 5 menit terakhir, lakukan hal berikut:
 - a. Pilih kustom di sudut kanan atas.
 - b. Pilih Relatif.
 - c. Pilih 5 menit.
 - d. Pilih Terapkan.

The screenshot shows the 'Log events' interface with a modal window for filtering. The modal window has tabs for 'Absolute' and 'Relative' (the latter is selected and circled in red). Under the 'Relative' tab, there are sections for 'Minutes', 'Hours', 'Days', and 'Weeks'. In the 'Minutes' section, the '5' option is selected and circled in red. Below these sections, there is a text input field containing '5' and a dropdown menu set to 'Minutes'. At the bottom of the modal, there are 'Clear', 'Cancel', and 'Apply' buttons, with the 'Apply' button circled in red.

6. (Opsional) Untuk melihat lebih sedikit log, Anda dapat memilih 1m dari sudut kanan atas.
7. Gulir ke bagian bawah entri log untuk menampilkan log terbaru.

Menggunakan log layanan

SiteWise Perangkat gateway tepi menyertakan file log layanan untuk membantu masalah debug. Bagian berikut akan membantu Anda menemukan dan memanfaatkan file log layanan untuk komponen AWS IoT SiteWise OPC-UA Collector dan AWS IoT SiteWise Publisher.

AWS IoT SiteWise File log layanan Kolektor OPC-UA

Komponen AWS IoT SiteWise OPC-UA Collector menggunakan file log berikut.

Linux

```
/greengrass/v2/logs/aws.iot.SiteWiseEdgeCollectorOpcua.log
```

Windows

```
C:\greengrass\v2\logs\aws.iot.SiteWiseEdgeCollectorOpcua.log
```

Untuk melihat log komponen ini

- Jalankan perintah berikut pada perangkat inti untuk melihat file log komponen ini secara real time. Ganti */greengrass/v2* atau *C:\greengrass\v2* dengan jalur ke folder AWS IoT Greengrass root.

Linux

```
sudo tail -f /greengrass/v2/logs/aws.iot.SiteWiseEdgeCollector0pcua.log
```

Windows (PowerShell)

```
Get-Content C:\greengrass\v2\logs\aws.iot.SiteWiseEdgeCollector0pcua.log -Tail 10 -Wait
```

AWS IoT SiteWise File log layanan penerbit

Komponen AWS IoT SiteWise Publisher menggunakan file log berikut.

Linux

```
/greengrass/v2/logs/aws.iot.SiteWiseEdgePublisher.log
```

Windows

```
C:\greengrass\v2\logs\aws.iot.SiteWiseEdgePublisher.log
```

Untuk melihat log komponen ini

- Jalankan perintah berikut pada perangkat inti untuk melihat file log komponen ini secara real time. Ganti */greengrass/v2* atau *C:\greengrass\v2* dengan jalur ke folder AWS IoT Greengrass root.

Linux

```
sudo tail -f /greengrass/v2/logs/aws.iot.SiteWiseEdgePublisher.log
```

Windows (PowerShell)

```
Get-Content C:\greengrass\v2\logs\aws.iot.SiteWiseEdgePublisher.log -Tail 10 -  
Wait
```

Menggunakan log peristiwa

SiteWise Perangkat gateway tepi menyertakan file log peristiwa untuk membantu masalah debug. Bagian berikut akan membantu Anda menemukan dan memanfaatkan file log peristiwa untuk komponen AWS IoT SiteWise OPC-UA Collector dan AWS IoT SiteWise Publisher.

AWS IoT SiteWise Log peristiwa Kolektor OPC-UA

Komponen AWS IoT SiteWise OPC-UA Collector mencakup log peristiwa untuk membantu pelanggan mengidentifikasi dan memperbaiki masalah. File log terpisah dari file log lokal, dan ditemukan di lokasi berikut. Ganti `/greengrass/v2` atau `C:\greengrass\v2` dengan jalur ke folder AWS IoT Greengrass root.

Linux

```
/greengrass/v2/work/aws.iot.SiteWiseEdgeCollectorOpcua/logs/  
IotSiteWiseOpcUaCollectorEvents.log
```

Windows

```
C:\greengrass\v2\work\aws.iot.SiteWiseEdgeCollectorOpcua\logs  
\IotSiteWiseOpcUaCollectorEvents.log
```

Log ini mencakup informasi terperinci dan instruksi pemecahan masalah. Informasi pemecahan masalah disediakan bersama diagnostik, dengan deskripsi tentang cara memperbaiki masalah, dan terkadang dengan tautan ke informasi lebih lanjut. Informasi diagnostik meliputi:

- Tingkat kepelikan
- Stempel Waktu
- Informasi khusus acara tambahan

Example Contoh log

```
dataSourceConnectionSuccess:
  Summary: Successfully connected to OpcUa server
  Level: INFO
  Timestamp: '2023-06-15T21:04:16.303Z'
  Description: Successfully connected to the data source.
  AssociatedMetrics:
  - Name: FetchedDataStreams
    Description: The number of fetched data streams for this data source
    Value: 1.0
    Namespace: IoTSiteWise
    Dimensions:
    - Name: SourceName
      Value: SourceName{value=OPC-UA Server}
    - Name: ThingName
      Value: test-core
  AssociatedData:
  - Name: DataSourceTrace
    Description: Name of the data source
    Data:
    - OPC-UA Server
  - Name: EndpointUri
    Description: The endpoint to which the connection was attempted.
    Data:
    - '"opc.tcp://10.0.0.1:1234"'
```

AWS IoT SiteWise Log acara penerbit

Komponen AWS IoT SiteWise Publisher menyertakan log peristiwa untuk membantu pelanggan mengidentifikasi dan memperbaiki masalah. File log terpisah dari file log lokal, dan ditemukan di lokasi berikut. Ganti */greengrass/v2* atau *C:\greengrass\v2* dengan jalur ke folder AWS IoT Greengrass root.

Linux

```
/greengrass/v2/work/aws.iot.SiteWiseEdgePublisher/logs/
IotSiteWisePublisherEvents.log
```

Windows

```
C:\greengrass\v2\work\aws.iot.SiteWiseEdgePublisher\logs  
\IotSiteWisePublisherEvents.log
```

Log ini mencakup informasi terperinci dan instruksi pemecahan masalah. Informasi pemecahan masalah disediakan bersama diagnostik, dengan deskripsi tentang cara memperbaiki masalah, dan terkadang dengan tautan ke informasi lebih lanjut. Informasi diagnostik meliputi:

- Tingkat kepelikan
- Stempel Waktu
- Informasi khusus acara tambahan

Example Contoh log

```
accountBeingThrottled:  
  Summary: Data upload speed slowed due to quota limits  
  Level: WARN  
  Timestamp: '2023-06-09T21:30:24.654Z'  
  Description: The IoT SiteWise Publisher is limited to the "Rate of data points  
  ingested"  
  quota for a customers account. See the associated documentation and associated  
  metric for the number of requests that were limited for more information. Note  
  that this may be temporary and not require any change, although if the issue  
  continues  
  you may need to request an increase for the mentioned quota.  
  FurtherInformation:  
  - https://docs.aws.amazon.com/iot-sitewise/latest/userguide/quotas.html  
  - https://docs.aws.amazon.com/iot-sitewise/latest/userguide/troubleshooting-gateway.html#gateway-issue-data-streams  
  AssociatedMetrics:  
  - Name: TotalErrorCount  
    Description: The total number of errors of this type that occurred.  
    Value: 327724.0  
  AssociatedData:  
  - Name: AggregatePropertyAliases  
    Description: The aggregated property aliases of the throttled data.  
    FileLocation: /greengrass/v2/work/aws.iot.SiteWiseEdgePublisher/./logs/data/  
  AggregatePropertyAliases_1686346224654.log
```


Pemantauan AWS IoT SiteWise dengan CloudWatch metrik Amazon

Anda dapat memantau AWS IoT SiteWise penggunaan CloudWatch, yang mengumpulkan data mentah dan memprosesnya menjadi metrik yang dapat dibaca, mendekati waktu nyata. Statistik ini disimpan untuk jangka waktu 15 bulan, sehingga Anda dapat mengakses informasi historis dan mendapatkan perspektif yang lebih baik tentang performa aplikasi atau layanan web Anda. Anda juga dapat mengatur alarm yang memperhatikan ambang batas tertentu dan mengirim notifikasi atau mengambil tindakan saat ambang batas tersebut terpenuhi. Untuk informasi selengkapnya, lihat [Panduan CloudWatch Pengguna Amazon](#).

AWS IoT SiteWise menerbitkan metrik dan dimensi yang tercantum di bagian di bawah ini ke namespace. `AWS/IoTSiteWise`

Tip

AWS IoT SiteWise menerbitkan metrik pada interval satu menit. Saat Anda melihat metrik ini dalam grafik di CloudWatch konsol, kami sarankan Anda memilih Periode 1 menit. Ini memungkinkan Anda melihat resolusi tertinggi yang tersedia dari data metrik Anda.

Topik

- [AWS IoT Greengrass Version 2 metrik gateway](#)
- [AWS IoT Greengrass Version 1 metrik gateway](#)

AWS IoT Greengrass Version 2 metrik gateway

AWS IoT SiteWise menerbitkan metrik gateway SiteWise Edge berikut. Semua metrik gateway SiteWise Edge dipublikasikan dalam interval satu menit.

SiteWise Metrik gerbang tepi

Metrik	Deskripsi
Gateway.CpuUsage	Penggunaan CPU dari gateway SiteWise Edge. Unit: Persentase

Metrik	Deskripsi
	Dimensi: Tidak ada
<code>Gateway.TotalDiskSpace</code>	Total ruang disk dari gateway SiteWise Edge. Unit: Bit Dimensi: Tidak ada
<code>Gateway.UsedDiskSpace</code>	Ruang disk yang digunakan dari gateway SiteWise Edge. Unit: Bit Dimensi: Tidak ada
<code>Gateway.AvailableDiskSpace</code>	Ruang disk yang tersedia dari gateway SiteWise Edge. Unit: Bit Dimensi: Tidak ada
<code>Gateway.UsedPercentageDiskSpace</code>	Persentase ruang disk yang digunakan dari gateway SiteWise Edge. Unit: Bit Dimensi: Tidak ada
<code>Gateway.TotalMemory</code>	Memori total gateway SiteWise Edge. Unit: Bit Dimensi: Tidak ada

Metrik	Deskripsi
<code>Gateway.UsedMemory</code>	<p>Memori yang digunakan dari gateway SiteWise Edge.</p> <p>Unit: Bit</p> <p>Dimensi: Tidak ada</p>
<code>Gateway.AvailableMemory</code>	<p>Memori yang tersedia dari gateway SiteWise Edge.</p> <p>Unit: Bit</p> <p>Dimensi: Tidak ada</p>
<code>Gateway.UsedPercentageMemory</code>	<p>Memori persentase yang digunakan dari gateway SiteWise Edge.</p> <p>Unit: Bit</p> <p>Dimensi: Tidak ada</p>
<code>Gateway.CloudConnectivity</code>	<p>Status konektivitas cloud dari gateway SiteWise Edge.</p> <p>Satuan: Tidak ada</p> <p>Dimensi: GatewayId</p>
<code>Gateway.SWE.Component.RunningStatus</code>	<p>Status berjalan komponen pada gateway SiteWise Edge.</p> <p>Satuan: Tidak ada</p> <p>Dimensi: GatewayId</p>

Metrik kolektor OPC-UA

Metrik	Deskripsi
<code>OpcUaCollector.Heartbeat</code>	<p>Dihasilkan setiap menit untuk setiap sumber OPC-UA (<code>sourceName</code>) yang terhubung ke gateway SiteWise Edge (<code>gatewayId</code>).</p> <p>Unit: Hitung (1 mewakili sumber terhubung dan 0 mewakili sumber terputus.)</p> <p>Dimensi: <code>GatewayId</code>, <code>SourceName</code></p>
<code>OpcUaCollector.ActiveDataStreamCount</code>	<p>Jumlah aliran data yang dilanggan gateway SiteWise Edge (<code>gatewayId</code>) untuk sumber OPC-UA (<code>sourceName</code>).</p> <p>Unit: Jumlah</p> <p>Dimensi: <code>GatewayId</code>, <code>SourceName</code>, <code>PropertyGroup</code></p>
<code>OpcUaCollector.IncomingValuesCount</code>	<p>Jumlah titik data yang diterima gateway SiteWise Edge (<code>gatewayId</code>) untuk sumber OPC-UA (<code>sourceName</code>), dihasilkan setiap menit.</p> <p>Unit: Jumlah</p> <p>Dimensi: <code>GatewayId</code>, <code>SourceName</code>, <code>PropertyGroup</code></p>
<code>OpcUaCollector.IncomingValuesError</code>	<p>Jumlah titik data yang diterima oleh gateway SiteWise Edge (<code>gatewayId</code>) dari sumber OPC-UA (<code>sourceName</code>) yang bukan nilai yang valid. Titik data ini tidak dicerna oleh OpcUa Kolektor, dihasilkan setiap menit.</p> <p>Unit: Jumlah</p>

Metrik	Deskripsi
	Dimensi: GatewayId, SourceName, PropertyGroup
<code>OpcUaCollector.ConversionErrors</code>	<p>Jumlah titik data yang diterima gateway SiteWise Edge (gatewayId) untuk sumber OPC-UA (sourceName) yang mengakibatkan kesalahan konversi saat mengirim data ke. AWS IoT SiteWise Titik data ini tidak akan dicerna oleh OpcUa Kolektor.</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId, SourceName</p>

AWS IoT SiteWise metrik penerbit

Metrik	Deskripsi
<code>IoTSiteWisePublisher.Heartbeat</code>	<p>Dihasilkan setiap menit oleh Penerbit di gateway SiteWise Edge.</p> <p>Unit: 1 (1 mewakili Penerbit sedang berjalan dan kehilangan titik data yang mewakili Penerbit tidak berjalan.)</p> <p>Dimensi: GatewayId</p>
<code>IoTSiteWisePublisher.PublishSuccessCount</code>	<p>Jumlah titik data yang berhasil dipublikasikan oleh gateway SiteWise Edge (GatewayId) ke cloud, dihasilkan setiap menit.</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId</p>
<code>IoTSiteWisePublisher.PublishFailureCount</code>	<p>Jumlah titik data yang gagal dipublikasikan oleh gateway SiteWise Edge (GatewayId), dihasilkan setiap menit.</p>

Metrik	Deskripsi
	Unit: Jumlah Dimensi: GatewayId
<code>IoTSiteWisePublisher.PublisherRejectedCount</code>	Jumlah titik data yang ditolak oleh gateway SiteWise Edge (GatewayId) dari sisi cloud, dihasilkan setiap menit. Unit: Jumlah Dimensi: GatewayId
<code>IoTSiteWisePublisher.DroppedCount</code>	Jumlah titik data yang dijatuhkan oleh gateway SiteWise Edge (GatewayId) dan tidak dipublikasikan ke cloud, dihasilkan setiap menit. Unit: Jumlah Dimensi: GatewayId

AWS IoT Greengrass Version 1 metrik gateway

AWS IoT SiteWise menerbitkan metrik gateway SiteWise Edge berikut. Semua metrik gateway SiteWise Edge dipublikasikan dalam interval satu menit.

Important

Untuk menerima metrik gateway SiteWise Edge, Anda harus menggunakan setidaknya versi 6 AWS IoT SiteWise konektor di gateway SiteWise Edge Anda. Untuk informasi selengkapnya, lihat [kolektor AWS IoT SiteWise OPC-UA](#) di Panduan AWS IoT Greengrass Version 1 Pengembang.

SiteWise Metrik gerbang tepi

Metrik	Deskripsi
Gateway.Heartbeat	<p>Dihasilkan setiap menit untuk setiap gateway SiteWise Edge (gatewayId) yang terhubung.</p> <p>Unit: 1 (1 yang mewakili gateway SiteWise Edge naik dan kehilangan titik data yang mewakili gateway SiteWise Edge terputus dari cloud.)</p> <p>Dimensi: GatewayId</p>
Gateway.PublishSuccessCount	<p>Jumlah titik data yang berhasil dipublikasikan oleh gateway SiteWise Edge (gatewayId).</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId</p>
Gateway.PublishFailureCount	<p>Jumlah titik data yang gagal dipublikasikan oleh gateway SiteWise Edge (gatewayId).</p> <p>Metrik ini menghitung kesalahan yang dihasilkan dari panggilan gateway SiteWise Edge ke BatchPutAssetPropertyValue operasi. Untuk informasi selengkapnya tentang pemecahan masalah gateway SiteWise Edge, lihat. Memecahkan masalah gateway Edge SiteWise</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId</p>
Gateway.ProcessFailureCount	<p>Jumlah titik data yang gagal diproses oleh gateway SiteWise Edge (gatewayId).</p> <p>Kesalahan jumlah metrik ini terjadi antara gateway SiteWise Edge dan sumber gateway SiteWise Edge, termasuk kesalahan yang</p>

Metrik	Deskripsi
	<p>dilaporkan oleh sumber. Untuk informasi selengkapnya tentang pemecahan masalah gateway SiteWise Edge, lihat. Memecahkan masalah gateway Edge SiteWise</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId</p>
<code>Gateway.PublishRejectedCount</code>	<p>Jumlah titik data dari gateway SiteWise Edge (gatewayId) yang ditolak.</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId</p>

Metrik terkait OPC-UA

Metrik	Deskripsi
<code>OPCUACollector.Heartbeat</code>	<p>Dihasilkan setiap menit untuk setiap sumber OPC-UA (sourceName) yang terhubung ke gateway SiteWise Edge (gatewayId).</p> <p>Unit: Hitung (1 mewakili sumber terhubung dan 0 mewakili sumber terputus.)</p> <p>Dimensi: GatewayId, SourceName</p>
<code>OPCUACollector.ActiveDataStreamCount</code>	<p>Jumlah aliran data yang dilanggan gateway SiteWise Edge (gatewayId) untuk sumber OPC-UA (). sourceName</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId, SourceName, PropertyGroup</p>

Metrik	Deskripsi
<code>OpcUaCollector.IncomingValuesCount</code>	<p>Jumlah titik data yang diterima gateway SiteWise Edge (<code>gatewayId</code>) untuk sumber OPC-UA (<code>sourceName</code>), dihasilkan setiap menit.</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId, SourceName, PropertyGroup</p>
<code>OpcUaCollector.IncomingValuesError</code>	<p>Jumlah titik data yang diterima gateway SiteWise Edge (<code>gatewayId</code>) dari sumber OPC-UA (<code>sourceName</code>) yang bukan nilai yang valid. Titik data ini tidak akan dicerna oleh OpcUa Kolektor, yang dihasilkan setiap menit.</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId, SourceName, PropertyGroup</p>
<code>OpcUaCollector.ConversionErrors</code>	<p>Jumlah titik data yang diterima gateway SiteWise Edge (<code>gatewayId</code>) untuk sumber OPC-UA (<code>sourceName</code>) yang mengakibatkan kesalahan konversi saat mengirim data ke. AWS IoT SiteWise Titik data ini tidak akan dicerna oleh OpcUa Kolektor.</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId, SourceName</p>

Metrik terkait EIP

Metrik	Deskripsi
<code>EIPCollector.Heartbeat</code>	<p>Dihasilkan setiap menit untuk setiap EIP Source (<code>sourceName</code>) yang terhubung ke gateway SiteWise Edge (<code>gatewayId</code>).</p> <p>Unit: 1 (1 mewakili sumber terhubung dan kehilangan titik data yang mewakili sumber terputus.)</p> <p>Dimensi: <code>GatewayId</code>, <code>SourceName</code></p>
<code>EIPCollector.IncomingValuesCount</code>	<p>Jumlah aliran data yang gateway SiteWise Edge (<code>gatewayId</code>) berlangganan untuk sumber EIP (<code>sourceName</code>).</p> <p>Unit: Jumlah</p> <p>Dimensi: <code>GatewayId</code>, <code>SourceName</code></p>
<code>EIPCollector.ActiveDataStreamCount</code>	<p>Jumlah titik data yang diterima gateway SiteWise Edge (<code>gatewayId</code>) untuk sumber EIP (<code>sourceName</code>).</p> <p>Unit: Jumlah</p> <p>Dimensi: <code>GatewayId</code>, <code>SourceName</code></p>

Metrik terkait Modbus

Metrik	Deskripsi
<code>ModbusTCPCollector.Heartbeat</code>	<p>Dihasilkan setiap menit untuk setiap Modbus Source (<code>sourceName</code>) yang terhubung ke gateway SiteWise Edge (<code>gatewayId</code>).</p>

Metrik	Deskripsi
	<p>Unit: 1 (1 mewakili sumber Modbus terhubung dan kehilangan titik data yang mewakili sumber terputus.)</p> <p>Dimensi: GatewayId, SourceName</p>
<code>ModbusTCPCollector.IncomingValuesCount</code>	<p>Jumlah aliran data yang gateway SiteWise Edge (gatewayId) berlangganan untuk sumber Modbus (). sourceName</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId, SourceName</p>
<code>ModbusTCPCollector.ActiveDataStreamCount</code>	<p>Jumlah titik data yang diterima gateway SiteWise Edge (gatewayId) untuk sumber Modbus (sourceName).</p> <p>Unit: Jumlah</p> <p>Dimensi: GatewayId, SourceName</p>

Logging panggilan AWS IoT SiteWise API dengan AWS CloudTrail

AWS IoT SiteWise terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di AWS IoT SiteWise. CloudTrail menangkap panggilan API untuk AWS IoT SiteWise sebagai acara. Panggilan yang diambil termasuk panggilan dari AWS IoT SiteWise konsol dan panggilan kode ke operasi AWS IoT SiteWise API. Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara terus menerus ke bucket Amazon S3, termasuk acara untuk. AWS IoT SiteWise Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat AWS IoT SiteWise, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk informasi selengkapnya CloudTrail, lihat [Panduan AWS CloudTrail Pengguna](#).

AWS IoT SiteWise informasi di CloudTrail

CloudTrail diaktifkan di AWS akun Anda saat Anda membuat akun. Ketika aktivitas acara yang didukung terjadi di AWS IoT SiteWise, aktivitas tersebut direkam dalam suatu CloudTrail peristiwa bersama dengan peristiwa AWS layanan lainnya dalam riwayat Acara. Anda dapat melihat, mencari, dan mengunduh acara terbaru di AWS akun Anda. Untuk informasi selengkapnya, lihat [Melihat peristiwa dengan riwayat CloudTrail acara](#).

Untuk catatan peristiwa yang sedang berlangsung di AWS akun Anda, termasuk acara untuk AWS IoT SiteWise, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Secara default, saat Anda membuat jejak di konsol, jejak tersebut berlaku untuk semua AWS Wilayah. Jejak mencatat peristiwa dari semua Wilayah di AWS partisi dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat berikut:

- [Gambaran umum untuk membuat jejak](#)
- [CloudTrail layanan dan integrasi yang didukung](#)
- [Mengonfigurasi notifikasi Amazon SNS untuk CloudTrail](#)
- [Menerima file CloudTrail log dari beberapa Wilayah](#) dan [Menerima file CloudTrail log dari beberapa akun](#)

Setiap peristiwa atau entri log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan berikut:

- Apakah permintaan itu dibuat dengan kredensial pengguna root atau AWS Identity and Access Management (IAM).
- Apakah permintaan dibuat dengan kredensial keamanan sementara untuk suatu peran atau pengguna gabungan.
- Apakah permintaan itu dibuat oleh AWS layanan lain.

Untuk informasi selengkapnya, lihat [Elemen userIdentity CloudTrail](#).

AWS IoT SiteWise peristiwa data di CloudTrail

[Peristiwa data](#) memberikan informasi tentang operasi sumber daya yang dilakukan pada atau di sumber daya (misalnya, membaca atau menulis ke objek Amazon S3). Ini juga dikenal sebagai

operasi bidang data. Peristiwa data seringkali merupakan aktivitas volume tinggi. Secara default, CloudTrail tidak mencatat peristiwa data. Riwayat CloudTrail peristiwa tidak merekam peristiwa data.

Biaya tambahan berlaku untuk peristiwa data. Untuk informasi selengkapnya tentang CloudTrail harga, lihat [AWS CloudTrail Harga](#).

Anda dapat mencatat peristiwa data untuk jenis AWS IoT SiteWise sumber daya menggunakan CloudTrail konsol AWS CLI, atau operasi CloudTrail API. [Tabel](#) di bagian ini menunjukkan jenis sumber daya yang tersedia untuk AWS IoT SiteWise.

- Untuk mencatat peristiwa data menggunakan CloudTrail konsol, buat [penyimpanan data jejak atau peristiwa](#) untuk mencatat peristiwa data, atau [perbarui penyimpanan data jejak atau peristiwa yang ada](#) untuk mencatat peristiwa data.
 1. Pilih Peristiwa data untuk mencatat peristiwa data.
 2. Dari daftar tipe peristiwa Data, pilih jenis sumber daya yang ingin Anda log peristiwa data.
 3. Pilih template pemilih log yang ingin Anda gunakan. Anda dapat mencatat semua peristiwa data untuk jenis sumber daya, mencatat semua `readOnly` peristiwa, mencatat semua `writeOnly` peristiwa, atau membuat templat pemilih log khusus untuk memfilter pada `readOnlyeventName`, dan `resources.ARN` bidang.
- Untuk mencatat peristiwa data menggunakan AWS CLI, konfigurasi `--advanced-event-selectors` parameter untuk mengatur `eventCategory` bidang sama dengan Data dan `resources.type` bidang sama dengan nilai tipe sumber daya (lihat [tabel](#)). Anda dapat menambahkan kondisi untuk memfilter nilai `readOnly`, `eventName`, dan `resources.ARN` bidang.
 - Untuk mengonfigurasi jejak untuk mencatat peristiwa data, jalankan [AWS CloudTrail put-event-selectors](#) perintah. Untuk informasi selengkapnya, lihat [Mencatat peristiwa data untuk jejak dengan AWS CLI](#)
 - Untuk mengonfigurasi penyimpanan data peristiwa untuk mencatat peristiwa data, jalankan [AWS CloudTrail create-event-data-store](#) perintah untuk membuat penyimpanan data peristiwa baru untuk mencatat peristiwa data, atau jalankan [AWS CloudTrail update-event-data-store](#) perintah untuk memperbarui penyimpanan data peristiwa yang ada. Untuk informasi selengkapnya, lihat [Mencatat peristiwa data untuk menyimpan data peristiwa dengan AWS CLI](#).

Tabel berikut mencantumkan jenis AWS IoT SiteWise sumber daya. Kolom tipe peristiwa data (konsol) menunjukkan nilai yang akan dipilih dari daftar tipe peristiwa Data di CloudTrail konsol. Kolom nilai `resources.type` menunjukkan **resources.type** nilai, yang akan Anda tentukan saat mengonfigurasi penyeleksi acara lanjutan menggunakan API atau AWS CLI CloudTrail

CloudTrailKolom API Data yang dicatat ke menampilkan panggilan API yang dicatat CloudTrail untuk jenis sumber daya.

Jenis peristiwa data (konsol)	nilai resources.type	API data masuk CloudTrail ke*
AWS IoT SiteWise aset	AWS::IoTSiteWise::Asset	<ul style="list-style-type: none"> • BatchPutAssetProperty • GetAssetPropertyValue • GetAssetPropertyValueHistory • GetAssetPropertyAggregates • GetInterpolatedAssetPropertyValues • BatchGetAssetPropertyValue • BatchGetAssetPropertyValueHistory • BatchGetAssetPropertyAggregates
AWS IoT SiteWise deret waktu	AWS::IoTSiteWise::TimeSeries	<ul style="list-style-type: none"> • BatchPutAssetProperty • GetAssetPropertyValue • GetAssetPropertyValueHistory • GetAssetPropertyAggregates • GetInterpolatedAssetPropertyValues • BatchGetAssetPropertyValue • BatchGetAssetPropertyValueHistory

Jenis peristiwa data (konsol)	nilai resources.type	API data masuk CloudTrail ke*
		<ul style="list-style-type: none"> BatchGetAssetPropertyAggregates

Note

Resources.type yang dicatat dalam peristiwa Cloudtrail bergantung pada pengenal yang digunakan dalam permintaan API. Jika id aset ditentukan dalam permintaan maka Asset resources.type dicatat, jika tidak resources.type dicatat. TimeSeries

* Anda dapat mengonfigurasi pemilih acara lanjutan untuk memfilter pada `eventName`, `readOnly`, dan `resources.ARN` bidang untuk mencatat hanya peristiwa yang penting bagi Anda. Untuk informasi lebih lanjut tentang bidang ini, lihat [AdvancedFieldSelector](#).

AWS IoT SiteWise acara manajemen di CloudTrail

[Acara manajemen](#) memberikan informasi tentang operasi manajemen yang dilakukan pada sumber daya di AWS akun Anda. Ini juga dikenal sebagai operasi pesawat kontrol. Secara default, CloudTrail mencatat peristiwa manajemen.

AWS IoT SiteWise mencatat semua operasi pesawat AWS IoT SiteWise kontrol sebagai peristiwa manajemen. Untuk daftar operasi bidang AWS IoT SiteWise kontrol yang AWS IoT SiteWise masuk ke log CloudTrail, lihat [Referensi AWS IoT SiteWise API](#).

Contoh: entri file AWS IoT SiteWise log

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Peristiwa mewakili permintaan tunggal dari sumber mana pun dan mencakup informasi tentang operasi yang diminta, tanggal dan waktu operasi, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi file tersebut tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan CreateAsset operasi.

```
{
```

```
"eventVersion": "1.05",
"userIdentity": {
  "type": "IAMUser",
  "principalId": "AIDACKCEVSQ6C2EXAMPLE",
  "arn": "arn:aws:iam::123456789012:user/Administrator",
  "accountId": "123456789012",
  "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
  "userName": "Administrator",
  "sessionContext": {
    "sessionIssuer": {},
    "webIdFederationData": {},
    "attributes": {
      "mfaAuthenticated": "false",
      "creationDate": "2020-03-11T17:26:40Z"
    }
  }
},
"invokedBy": "signin.amazonaws.com",
},
"eventTime": "2020-03-11T18:01:22Z",
"eventSource": "iotsitewise.amazonaws.com",
"eventName": "CreateAsset",
"awsRegion": "us-east-1",
"sourceIPAddress": "203.0.113.0",
"userAgent": "signin.amazonaws.com",
"requestParameters": {
  "assetName": "Wind Turbine 1",
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-00000EXAMPLE"
},
"responseElements": {
  "assetId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
  "assetArn": "arn:aws:iotsitewise:us-east-1:123456789012:asset/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
  "assetStatus": {
    "state": "CREATING"
  }
},
"requestID": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
"eventID": "a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}
```


Menandai sumber daya Anda AWS IoT SiteWise

Menandai AWS IoT SiteWise sumber daya Anda menyediakan cara yang ampuh untuk mengkategorikan, mengelola, dan mengambil aset organisasi secara efisien. Dengan menetapkan tag, yang terdiri dari pasangan nilai kunci, Anda dapat melampirkan metadata deskriptif ke sumber daya Anda. Metadata dari tag dapat digunakan untuk merampingkan operasi. Misalnya, dalam skenario ladang angin, tag memungkinkan Anda memberi label turbin dengan atribut tertentu seperti lokasi, kapasitas, dan status operasional, memungkinkan identifikasi dan manajemen cepat di dalamnya. AWS IoT SiteWise

Mengintegrasikan tag dengan kebijakan AWS Identity and Access Management (IAM) meningkatkan keamanan dan kontrol operasional dengan mendefinisikan aturan akses bersyarat. Ini berarti Anda dapat menentukan bahwa hanya pengguna dengan tag tertentu. Misalnya, hanya mereka yang ditandai dengan peran atau departemen tertentu, yang dapat mengakses atau memodifikasi sumber daya tertentu.

Menggunakan tag di AWS IoT SiteWise

Gunakan tag untuk mengkategorikan AWS IoT SiteWise sumber daya Anda berdasarkan tujuan, pemilik, lingkungan, atau klasifikasi lainnya untuk kasus penggunaan Anda. Ketika Anda memiliki banyak sumber daya dari jenis yang sama, Anda dapat dengan cepat mengidentifikasi sumber daya tertentu berdasarkan tag-nya.

Setiap tag terdiri dari kunci dan nilai opsional yang Anda tentukan. Misalnya, Anda dapat membuat serangkaian tag untuk model aset Anda untuk melacaknya sesuai dengan proses industri yang mereka dukung. Disarankan untuk mengembangkan satu set kunci tag yang disesuaikan untuk setiap jenis sumber daya yang Anda kelola. Menggunakan satu set kunci tag yang konsisten dapat membuatnya lebih mudah mengelola sumber daya.

Menandai dengan AWS Management Console

Editor Tag di AWS Management Console menyediakan cara terpusat dan terpadu bagi Anda untuk membuat dan mengelola tag Anda untuk sumber daya dari semua AWS layanan. Untuk informasi selengkapnya, lihat [Editor tanda](#) di Panduan Pengguna AWS Resource Groups .

Menandai dengan API AWS IoT SiteWise

AWS IoT SiteWise API juga menggunakan tag. Sebelum Anda membuat tag, perhatikan pembatasan penandaan. Untuk informasi selengkapnya, lihat [Konvensi penamaan dan penggunaan tag](#) di.

Referensi Umum AWS

- Untuk menambahkan tanda selama pembuatan sumber daya, tentukan tag dalam properti tags sumber daya.
- Untuk menambahkan tag ke sumber daya yang ada, atau untuk memperbarui nilai tag, gunakan [TagResource](#) operasi.
- Untuk menghapus tag dari sumber daya, gunakan [UntagResource](#) operasi.
- Untuk mengambil tag yang terkait dengan sumber daya, gunakan [ListTagsForResource](#) operasi, atau jelaskan sumber daya dan periksa propertinya tags.

Tabel berikut mencantumkan sumber daya yang dapat Anda beri tag menggunakan AWS IoT SiteWise API `Create` dan terkait serta `Describe` operasinya.

Sumber daya yang dapat ditandai AWS IoT SiteWise

Sumber daya	Buat operasi	Jelaskan operasi
Model aset atau model komponen	CreateAssetModel	DescribeAssetModel
Aset	CreateAsset	DescribeAsset
SiteWise Gerbang tepi	CreateGateway	DescribeGateway
Portal	CreatePortal	DescribePortal
Proyek	CreateProject	DescribeProject
Dasbor	CreateDashboard	DescribeDashboard
Kebijakan akses	CreateAccessPolicy	DescribeAccessPolicy
Seri waktu	BatchPutAssetPropertyValue	DescribeTimeSeries

Untuk [BatchPutAssetPropertyValue](#), Anda dapat mengonfigurasi sumber data untuk mengirim data industri AWS IoT SiteWise sebelum Anda membuat model aset dan aset. AWS IoT SiteWise secara otomatis membuat aliran data untuk menerima aliran data mentah dari peralatan Anda. Untuk informasi selengkapnya, lihat [Mengelola konsumsi data](#).

Gunakan operasi berikut untuk melihat dan mengelola tag untuk sumber daya yang mendukung penandaan:

- [TagResource](#)— Menambahkan tag ke sumber daya, atau memperbarui nilai tag yang ada.
- [ListTagsForResource](#)— Daftar tag untuk sumber daya.
- [UntagResource](#)— Menghapus tag dari sumber daya.

Tambahkan atau hapus tag dari sumber daya kapan saja. Untuk memperbarui nilai kunci tag yang ada, tambahkan tag baru dengan kunci yang sama dan nilai baru yang Anda inginkan ke sumber daya. Tindakan ini menggantikan nilai lama dengan yang baru. Meskipun dimungkinkan untuk menetapkan string kosong sebagai nilai tag, Anda tidak dapat menetapkan nilai null.

Menghapus sumber daya juga menghapus tag apa pun yang terkait dengannya.

Menggunakan tanda dengan kebijakan IAM

Gunakan tag sumber daya dalam kebijakan IAM Anda untuk mengontrol akses dan izin pengguna. Misalnya, kebijakan dapat memungkinkan pengguna untuk hanya membuat sumber daya yang memiliki tag tertentu yang dilampirkan. Kebijakan juga dapat membatasi pengguna untuk membuat atau memodifikasi sumber daya yang memiliki tanda tertentu.

Note

Jika Anda menggunakan tanda untuk mengizinkan atau menolak akses pengguna ke sumber daya, Anda harus menolak memberi pengguna kemampuan untuk menambahkan atau menghapus tanda tersebut untuk sumber daya yang sama. Jika tidak, pengguna dapat melewati batasan Anda dan mendapatkan akses ke sumber daya dengan memodifikasi tag-nya.

Anda dapat menggunakan kunci konteks kondisi berikut dan nilai dalam `Condition` elemen (juga disebut `Condition` blok) dari pernyataan kebijakan.

```
aws:ResourceTag/tag-key: tag-value
```

Izinkan atau tolak tindakan pada sumber daya dengan tanda tertentu.

```
aws:RequestTag/tag-key: tag-value
```

Mengharuskan tag tertentu digunakan (atau tidak digunakan) saat membuat atau memodifikasi sumber daya yang dapat diberi tag.

```
aws:TagKeys: [tag-key, ...]
```

Mengharuskan sekumpulan kunci tag tertentu digunakan (atau tidak digunakan) saat membuat atau memodifikasi sumber daya yang dapat diberi tag.

Note

Kunci dan nilai konteks syarat dalam kebijakan IAM hanya berlaku pada tindakan yang memiliki sumber daya yang dapat ditandai sebagai parameter wajib. Misalnya, Anda dapat mengatur akses bersyarat berbasis tag untuk [ListAssets](#). Anda tidak dapat menyetel akses bersyarat berbasis tag [PutLoggingOptions](#) karena tidak ada sumber daya yang dapat diberi tag yang direferensikan dalam permintaan.

Untuk informasi selengkapnya, lihat [Mengontrol akses ke AWS sumber daya menggunakan tag sumber daya](#) dan [referensi kebijakan IAM JSON](#) di Panduan Pengguna IAM.

Contoh kebijakan IAM menggunakan tag

- [Melihat AWS IoT SiteWise aset berdasarkan tag](#)

Pemecahan masalah AWS IoT SiteWise

Gunakan informasi di bagian ini untuk memecahkan masalah dan menyelesaikan masalah. AWS IoT SiteWise

Topik

- [Memecahkan masalah operasi impor dan ekspor massal](#)
- [Memecahkan masalah portal AWS IoT SiteWise](#)
- [Memecahkan masalah gateway Edge SiteWise](#)
- [Memecahkan masalah tindakan aturan AWS IoT SiteWise](#)

Memecahkan masalah operasi impor dan ekspor massal

Untuk menangani dan mendiagnosis kesalahan yang dihasilkan selama tugas transfer, lihat AWS IoT TwinMaker GetMetadataTransferJobAPI:

1. Setelah membuat dan menjalankan tugas transfer, panggil GetMetadataTransferJobAPI:

```
aws iottwinmaker get-metadata-transfer-job \  
--metadata-transfer-job-id your_metadata_transfer_job_id \  
--region us-east-1
```

2. Keadaan pekerjaan berubah menjadi salah satu negara di bawah ini:
 - DISELESAIKAN
 - DIBATALKAN
 - ERROR
3. GetMetadataTransferJobAPI mengembalikan [MetadataTransferJobProgress](#) objek.
4. MetadataTransferJobProgressObjek berisi parameter berikut:
 - FailedCount: Menunjukkan jumlah aset yang gagal selama proses transfer.
 - SkippedCount: Menunjukkan jumlah aset yang dilewati selama proses transfer.
 - SucceededCount: Menunjukkan jumlah aset yang berhasil selama proses transfer.
 - TotalCount: Menunjukkan jumlah total aset yang terlibat dalam proses transfer.

- Selain itu, elemen `reportURL` dikembalikan oleh panggilan API, yang berisi URL yang telah ditandatangani sebelumnya. Jika pekerjaan transfer Anda memiliki kesalahan yang perlu diselidiki, Anda dapat mengunduh laporan kesalahan lengkap di URL ini.

Memecahkan masalah portal AWS IoT SiteWise

Memecahkan masalah umum dengan portal Anda AWS IoT SiteWise .

Pengguna dan administrator tidak dapat mengakses portal AWS IoT SiteWise

Jika pengguna atau administrator tidak dapat mengakses AWS IoT SiteWise portal Anda, Anda mungkin telah membatasi izin dalam kebijakan terlampir AWS Identity and Access Management (IAM) yang mencegah login berhasil.

Lihat contoh kebijakan IAM berikut yang akan mengakibatkan kegagalan login:

Note

Setiap kebijakan IAM terlampir yang menyertakan "Condition" elemen akan menyebabkan kegagalan login.

Contoh 1: Kondisi di sini adalah IP terbatas, dan ini akan menyebabkan kegagalan login.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:DescribePortal"
      ],
      "Resource": "*",
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "REPLACESAMPLEIP"
          ]
        }
      }
    }
  ]
}
```

```

    }
  }
]
}

```

Contoh 2: Kondisi di sini adalah tag yang disertakan, dan ini akan menyebabkan kegagalan login.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotsitewise:DescribePortal"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "aws:ResourceTag/project": "*"
        }
      }
    }
  ]
}

```

Saat menambahkan pengguna atau administrator ke portal, hindari membuat kebijakan IAM yang membatasi izin pengguna, seperti IP terbatas. Kebijakan terlampir dengan izin terbatas tidak akan dapat terhubung ke AWS IoT SiteWise portal.

Memecahkan masalah gateway Edge SiteWise

AWS IoT SiteWise Edge gateway menjalankan satu set komponen. AWS IoT Greengrass Anda dapat mengonfigurasi gateway SiteWise Edge Anda untuk mencatat peristiwa konektor ke Amazon CloudWatch dan ke sistem file lokal gateway SiteWise Edge Anda. Kemudian, Anda dapat melihat file log yang terkait dengan konektor untuk memecahkan masalah gateway SiteWise Edge Anda.

Anda juga dapat melihat CloudWatch metrik yang dilaporkan oleh gateway SiteWise Edge untuk memecahkan masalah dengan konektivitas atau aliran data. Untuk informasi selengkapnya, lihat [AWS IoT Greengrass Version 1 metrik gateway](#).

Topik

- [Mengkonfigurasi dan mengakses log gateway SiteWise Edge](#)
- [Memecahkan masalah gateway SiteWise Edge](#)
- [Memecahkan masalah AWS IoT Greengrass](#)

Mengkonfigurasi dan mengakses log gateway SiteWise Edge

Sebelum Anda dapat melihat log gateway SiteWise Edge, Anda harus mengonfigurasi gateway SiteWise Edge Anda untuk mengirim log ke Amazon CloudWatch Logs atau menyimpan log di sistem file lokal.

- Gunakan CloudWatch Log jika Anda ingin menggunakannya AWS Management Console untuk melihat file log gateway SiteWise Edge Anda. Untuk informasi selengkapnya, lihat [Menggunakan CloudWatch Log Amazon](#).
- Gunakan log sistem file lokal jika Anda ingin menggunakan baris perintah atau perangkat lunak lokal untuk melihat file log gateway SiteWise Edge Anda. Untuk informasi selengkapnya, lihat [Menggunakan log layanan](#).

Memecahkan masalah gateway SiteWise Edge

Gunakan informasi berikut untuk memecahkan masalah gateway SiteWise Edge.

Masalah

- [Tidak dapat menerapkan paket ke gateway SiteWise Edge](#)
- [Sumber Modbus TCP tidak sinkron](#)
- [Tidak dapat terhubung ke pengelola aliran](#)
- [Tidak dapat terhubung ke sumber OPC-UA](#)
- [AWS IoT SiteWise tidak menerima data dari server OPC-UA](#)
- [Tidak ada data yang ditampilkan di dasbor](#)
- [“Tidak dapat menemukan atau memuat kelas utama” muncul di aws.iot. SiteWiseEdgePublisher log pada kesalahan /greengrass/v2/logs](#)

Tidak dapat menerapkan paket ke gateway SiteWise Edge

Jika AWS IoT Greengrass nucleus component (`aws.greengrass.Nucleus`) sudah kedaluwarsa, Anda mungkin tidak dapat menerapkan paket ke gateway SiteWise Edge Anda. Anda dapat

menggunakan AWS IoT Greengrass V2 konsol untuk memutakhirkan komponen AWS IoT Greengrass nukleus.

Tingkatkan komponen AWS IoT Greengrass nukleus (konsol)

1. Navigasikan ke [konsol AWS IoT Greengrass](#) tersebut.
2. Di panel navigasi, di bawah AWS IoT Greengrass, pilih Deployment.
3. Dalam daftar Deployment, pilih deployment yang ingin Anda revisi.
4. Pilih Revise.
5. Pada halaman Tentukan target, pilih Selanjutnya.
6. Pada halaman Pilih komponen, di bawah Komponen publik, di kotak pencarian, masukkan **aws.greengrass.Nucleus**, lalu pilih AWS.GreenGrass.Nucleus.
7. Pilih Berikutnya.
8. Pada halaman Konfigurasi komponen, pilih Berikutnya.
9. Pada halaman Konfigurasi pengaturan lanjutan, pilih Berikutnya.
10. Di halaman Tinjau, pilih Deploy.

Sumber Modbus TCP tidak sinkron

Sumber Modbus TCP Anda mungkin tidak sinkron jika tipe data sumber Anda adalah ASCII, UTF8, atau ISO8859 dan Anda menjalankan versi lama konektor Adaptor Protokol Modbus-TCP. Untuk meningkatkan konektor ke versi terbaru, lakukan hal berikut:

1. Masuk ke [konsol AWS IoT Greengrass V1](#).
2. Di panel navigasi, pilih Grup.
3. Di bawah AWS IoT Greengrass grup, pilih grup sasaran.
4. Di panel navigasi, pilih Konektor.
5. Di kolom Upgrade, pilih Tersedia.
6. Pada halaman konektor Upgrade, pilih versi terbaru, lalu pilih Upgrade.

Untuk informasi selengkapnya, lihat [Konektor Adaptor Protokol Modbus-TCP](#) di Panduan Pengembang. AWS IoT Greengrass Version 1

Tidak dapat terhubung ke pengelola aliran

Anda mungkin melihat pesan log `swPublisher` kesalahan berikut jika pengelola aliran tidak diaktifkan di AWS IoT Greengrass grup gateway SiteWise Edge Anda.

```
com.amazonaws.greengrass.streammanager.client.StreamManagerClientImpl: Connect failed
```

Pada versi 6, AWS IoT SiteWise konektor membutuhkan manajer aliran. Untuk informasi selengkapnya tentang cara mengaktifkan pengelola aliran, lihat langkah 5 dari [Mengkonfigurasi grup AWS IoT Greengrass](#).

Tidak dapat terhubung ke sumber OPC-UA

Anda mungkin melihat pesan log `OPCUACollector` kesalahan berikut jika versi OpenJDK yang diinstal tidak didukung.

```
java.security.KeyStoreException: Key protection algorithm not found:
java.security.UnrecoverableKeyException: Encrypt Private Key failed: unrecognized
algorithm name: PBESWithSHA1AndDESede
Failed to start OPC-UA Connection for Source 'Server 1': Failed to add key to
store
```

Untuk menurunkan versi ke versi OpenJDK yang didukung, ikuti langkah-langkah di bagian ini. Langkah-langkah ini mengasumsikan bahwa Anda menggunakan perangkat dengan Ubuntu. Jika Anda menggunakan distribusi Linux yang berbeda, lihat dokumentasi yang relevan untuk perangkat Anda.

Untuk menurunkan versi ke dukungan Amazon Corretto 8

1. Untuk menghapus instalasi OpenJDK saat ini, jalankan salah satu perintah berikut.

- ```
sudo apt purge -y openjdk-8-jre-headless
```
- ```
sudo apt-get purge -y java-1.8.0-amazon-corretto-jdk
```

2. Untuk mengunduh dan menginstal [Amazon Corretto 8](#) yang didukung, jalankan perintah berikut.

```
curl -s https://corretto.aws/downloads/resources/8.282.08.1/java-1.8.0-amazon-
corretto-jdk_8.282.08-1_amd64.deb --output /tmp/java-1.8.0-amazon-corretto-
jdk_8.282.08-1_amd64.deb
```

```
sudo apt-get update && sudo apt-get install java-common
sudo dpkg --install /tmp/java-1.8.0-amazon-corretto-jdk_8.282.08-1_amd64.deb
```

3. Untuk me-restart perangkat lunak AWS IoT Greengrass V1 Core, jalankan perintah berikut.

```
sudo /greengrass/ggc/core/greengrassd restart
```

AWS IoT SiteWise tidak menerima data dari server OPC-UA

Jika AWS IoT SiteWise aset Anda tidak menerima data yang dikirim oleh server OPC-UA, Anda dapat mencari log gateway SiteWise Edge untuk memecahkan masalah. Cari `swPublisher` log tingkat info yang berisi pesan berikut.

```
Emitting diagnostic name=PublishError.SomeException
```

Berdasarkan jenis *SomeException* di log, gunakan jenis pengecualian berikut dan masalah terkait untuk memecahkan masalah gateway SiteWise Edge Anda:

- `ResourceNotFoundException`— Server OPC-UA Anda mengirim data yang tidak cocok dengan alias properti untuk aset apa pun. Pengecualian ini dapat terjadi dalam dua kasus:
 - Alias properti Anda tidak sama persis dengan variabel OPC-UA Anda, termasuk awalan sumber apa pun yang Anda tentukan. Periksa apakah alias properti dan awalan sumber Anda sudah benar.
 - Anda belum memetakan variabel OPC-UA Anda ke properti aset. Untuk informasi selengkapnya, lihat [Memetakan aliran data industri ke properti aset](#).

Jika Anda sudah memetakan semua variabel OPC-UA yang Anda inginkan AWS IoT SiteWise, Anda dapat memfilter variabel OPC-UA mana yang dikirim gateway Edge. SiteWise Untuk informasi selengkapnya, lihat [Menggunakan filter node OPC-UA](#).

- `AccessDeniedException`— AWS IoT Greengrass Grup gateway SiteWise Edge Anda tidak memiliki izin yang cukup untuk menggunakan [BatchPutAssetPropertyValue](#) operasi untuk mengirim data ke properti aset. Untuk informasi selengkapnya, lihat [persyaratan AWS IoT SiteWise konektor](#).
- `InvalidRequestException`— Jenis data variabel OPC-UA Anda tidak cocok dengan tipe data properti aset Anda. Misalnya, jika variabel OPC-UA memiliki tipe data integer, properti aset terkait Anda harus tipe data integer. Properti aset tipe ganda tidak dapat menerima nilai integer OPC-UA. Untuk memperbaiki masalah ini, tentukan properti baru dengan tipe data yang benar.

- **TimestampOutOfRangeException**— Gateway SiteWise Edge Anda mengirim data yang berada di luar jangkauan yang AWS IoT SiteWise menerima. AWS IoT SiteWise menolak setiap titik data dengan stempel waktu lebih awal dari 7 hari di masa lalu atau lebih baru dari 5 menit di masa depan. Jika gateway SiteWise Edge Anda kehilangan daya atau koneksi ke AWS Cloud, Anda mungkin perlu menghapus cache gateway SiteWise Edge Anda.
- **ThrottlingException** atau **LimitExceededException**— Permintaan Anda melebihi kuota AWS IoT SiteWise layanan, seperti tingkat titik data yang dicerna atau tingkat permintaan untuk operasi API data properti aset. Periksa apakah konfigurasi Anda tidak melebihi [AWS IoT SiteWise kuota](#).

Tidak ada data yang ditampilkan di dasbor

Jika tidak ada data yang ditampilkan di dasbor Anda, maka di [AWS IoT SiteWise konsol](#), periksa apakah konfigurasi Publisher dan Sumber Data tidak sinkron. Untuk memperbaiki masalah ini, gunakan produk berikut:

1. Masuk ke [AWS IoT SiteWise konsol](#).
2. Di bagian Edge, buka bagian Gateways.
3. Di bawah Sumber data, pilih Edit.

The screenshot shows the AWS IoT SiteWise console interface. On the left is a navigation menu with options like Edge, Gateways, Build, Monitor, and Settings. The main content area is divided into several sections:

- Gateway configuration:** Shows Gateway ID (408c0731-32fc-4a7e-bf4d-ba3daaa8cfb4) and Greengrass core device (ErrorGatewayGreengrassCoreDevice-4r2SMI_GH).
- Edge capabilities:** Shows Data collection pack (Activated), Data processing pack (Activated), and Edge LDAP/AD connection (Not activated).
- Publisher configuration:** Shows Configuration status (Out of sync), Publishing order (Oldest first), Compress uploaded data (Active), Cutoff period (Not configured), Retention period (Not configured), Rotation period (Not configured), and Export size limit (Not configured).
- Data sources (1):** A table with one entry: 'demo source' of type 'OPC-UA' with destination 'SiteWise Cloud' and configuration status 'Out of sync'. A green arrow points to the 'Edit' button for this source.

4. Pilih nama sumber baru, dan pilih Simpan untuk mengonfirmasi perubahan Anda.

5. Verifikasi perubahan Anda dengan mengonfirmasi nama sumber data telah diperbarui di tabel Sumber data.

Mengubah nama sumber data dapat mempercepat sinkronisasi dari cloud ke tepi, memperbaiki kesalahan Out of sync.

“Tidak dapat menemukan atau memuat kelas utama” muncul di aws.iot.SiteWiseEdgePublisher log pada kesalahan /greengrass/v2/logs

Jika Anda melihat kesalahan ini, Anda mungkin perlu memperbarui versi java dari gateway SiteWise Edge Anda.

- Dari terminal, jalankan perintah berikut:

```
java -version
```

Versi java gateway SiteWise Edge Anda berjalan dengan akan muncul di bawah OpenJDK Runtime Environment. Anda akan melihat respons seperti berikut:

```
openjdk version "11.0.20" 2023-07-18 LTS
OpenJDK Runtime Environment Corretto011.0.20.8.1 (build 11.0.20+8-LTS
OpenJDK 64-Bit Server VM Corretto-11.0.20.8.1 (build 11.0.20+8-LTS, mixed node)
```

Jika Anda menjalankan Java versi 11.0.20.8.1 Anda harus memperbarui paket IoT SiteWise Publisher ke versi 2.4.1 atau yang lebih baru. Hanya versi java 11.0.20.8.1 yang terpengaruh, lingkungan dengan versi java lainnya dapat terus menggunakan versi lama dari komponen IoT Publisher. SiteWise Untuk informasi selengkapnya tentang memperbarui paket komponen, lihat [Mengubah versi paket komponen gateway SiteWise Edge](#).

Memecahkan masalah AWS IoT Greengrass

Untuk menemukan solusi untuk banyak masalah saat mengonfigurasi atau menerapkan gateway SiteWise Edge AWS IoT Greengrass, lihat [Pemecahan Masalah AWS IoT Greengrass](#) di Panduan Pengembang. AWS IoT Greengrass

Memecahkan masalah tindakan aturan AWS IoT SiteWise

Untuk memecahkan masalah tindakan AWS IoT SiteWise aturan Anda AWS IoT Core, Anda dapat melakukan salah satu prosedur berikut:

- Konfigurasi CloudWatch Log Amazon
- Konfigurasi tindakan kesalahan publikasi ulang untuk aturan Anda

Kemudian, bandingkan pesan kesalahan dengan kesalahan dalam topik ini untuk memecahkan masalah Anda.

Topik

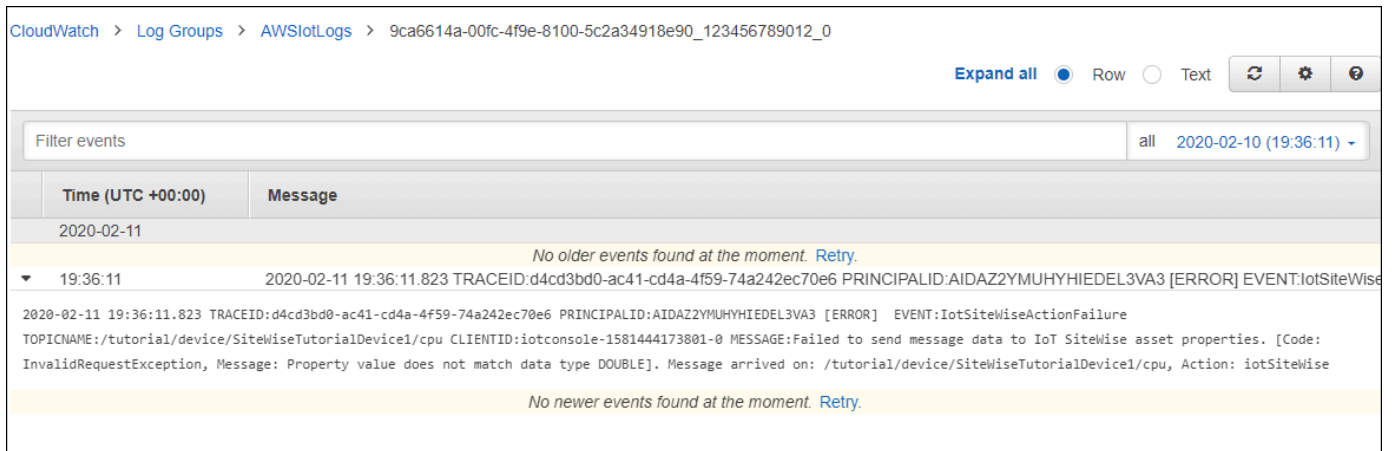
- [Mengkonfigurasi log AWS IoT Core](#)
- [Mengonfigurasi tindakan kesalahan penerbitan ulang](#)
- [Memecahkan masalah](#)
- [Memecahkan masalah aturan](#)
- [Memecahkan masalah aturan](#)

Mengkonfigurasi log AWS IoT Core

Anda dapat mengonfigurasi AWS IoT untuk mencatat berbagai tingkat informasi ke CloudWatch Log.

Untuk mengkonfigurasi dan mengakses CloudWatch Log

1. Untuk mengonfigurasi pencatatan AWS IoT Core, lihat [Pemantauan dengan CloudWatch Log](#) di Panduan AWS IoT Pengembang.
2. Navigasikan ke [konsol CloudWatch](#) tersebut.
3. Pada panel navigasi, pilih Grup log.
4. Pilih AWSIoTLogsgруп.
5. Pilih aliran log terbaru. Secara default, CloudWatch menampilkan aliran log terbaru terlebih dahulu.
6. Pilih entri log untuk memperluas pesan log. Entri log Anda mungkin terlihat seperti tangkapan layar berikut.



The screenshot shows the AWS CloudWatch console interface for a Log Group. The breadcrumb navigation is: CloudWatch > Log Groups > AWSIoTLogs > 9ca6614a-00fc-4f9e-8100-5c2a34918e90_123456789012_0. The interface includes a search bar for filter events, a date range selector set to 'all' for '2020-02-10 (19:36:11)', and a table of log events. The table has two columns: 'Time (UTC +00:00)' and 'Message'. A log entry is expanded, showing a timestamp of '2020-02-11 19:36:11' and a message: '2020-02-11 19:36:11.823 TRACEID:d4cd3bd0-ac41-cd4a-4f59-74a242ec70e6 PRINCIPALID:AIDAZ2YMUHYHIEDEL3VA3 [ERROR] EVENT:IotSiteWiseActionFailure TOPICNAME:/tutorial/device/SiteWiseTutorialDevice1/cpu CLIENTID:iotconsole-1581444173801-0 MESSAGE:Failed to send message data to IoT SiteWise asset properties. [Code: InvalidRequestException, Message: Property value does not match data type DOUBLE]. Message arrived on: /tutorial/device/SiteWiseTutorialDevice1/cpu, Action: iotSiteWise'. The message is highlighted in yellow, and there are 'Retry' links above and below the entry.

7. Bandingkan pesan kesalahan dengan kesalahan dalam topik ini untuk memecahkan masalah Anda.

Mengonfigurasi tindakan kesalahan penerbitan ulang

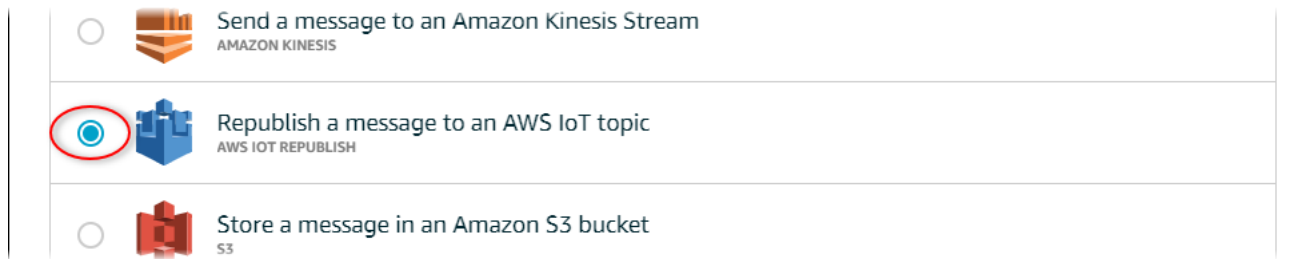
Anda dapat mengonfigurasi tindakan kesalahan pada aturan Anda untuk menangani pesan kesalahan. Dalam prosedur ini, Anda mengonfigurasi tindakan aturan penerbitan ulang sebagai tindakan kesalahan untuk melihat pesan kesalahan di klien pengujian MQTT.

Note

Tindakan kesalahan penerbitan ulang hanya menghasilkan log ERROR level yang setara. Jika Anda ingin lebih banyak log verbose, Anda harus [mengonfigurasi CloudWatch Log](#).

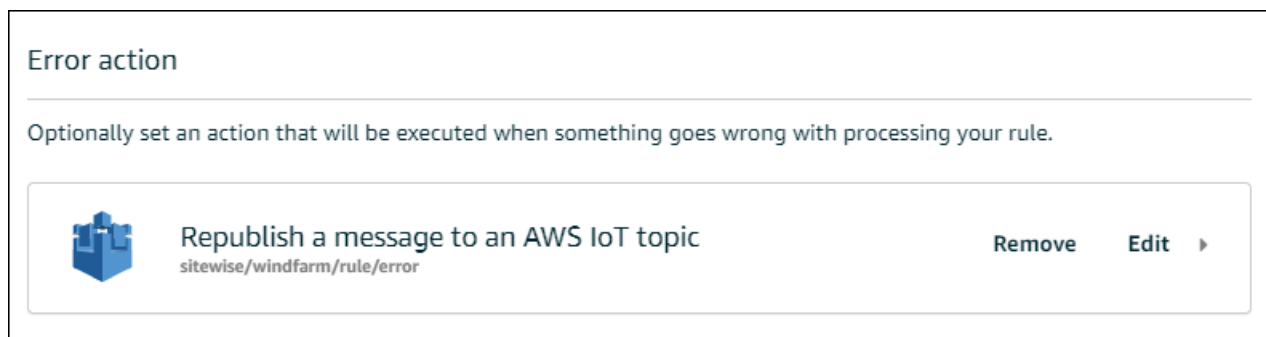
Untuk menambahkan tindakan kesalahan penerbitan ulang ke aturan

1. Navigasikan ke [konsol AWS IoT](#) tersebut.
2. Di panel navigasi kiri, pilih Bertindak lalu pilih Aturan.
3. Pilih aturan Anda.
4. Di bawah Tindakan kesalahan, pilih Tambah tindakan.
5. Pilih Menerbitkan ulang pesan ke suatu AWS IoT topik.



6. Pilih Konfigurasi tindakan di bagian bawah halaman.
7. Di Topik, masukkan topik unik (misalnya, **sitewise/windfarm/rule/error**). AWS IoT Core akan menerbitkan ulang pesan kesalahan ke topik ini.
8. Pilih Pilih untuk memberikan AWS IoT Core akses untuk melakukan tindakan kesalahan.
9. Pilih Pilih di samping peran yang Anda buat untuk aturan.
10. Pilih Perbarui Peran untuk menambahkan izin tambahan ke peran.
11. Pilih Tambahkan tindakan.

Tindakan kesalahan aturan Anda akan terlihat mirip dengan tangkapan layar berikut.



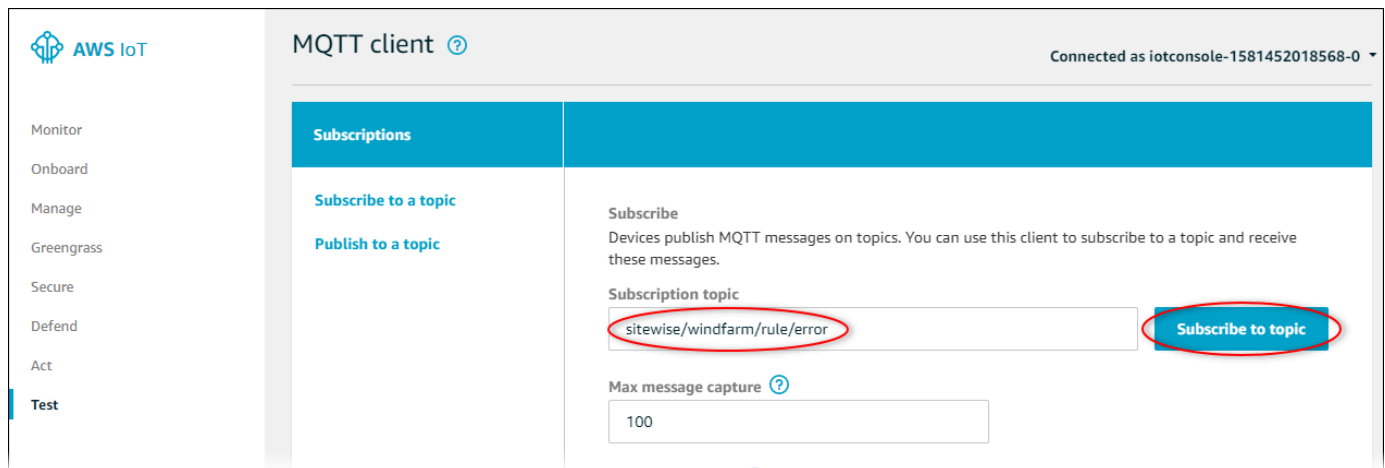
12. Pilih panah belakang di kiri atas konsol untuk kembali ke rumah AWS IoT konsol.

Setelah mengatur tindakan kesalahan penerbitan ulang, Anda dapat melihat pesan kesalahan di klien pengujian MQTT di. AWS IoT Core

Dalam prosedur berikut, Anda berlangganan topik kesalahan di klien pengujian MQTT. Di klien pengujian MQTT, Anda dapat menerima pesan galat aturan untuk memecahkan masalah.

Untuk berlangganan topik tindakan kesalahan

1. Navigasikan ke [konsol AWS IoT](#) tersebut.
2. Di halaman navigasi kiri, pilih Uji untuk membuka klien pengujian MQTT.
3. Di bidang Subscription topic, masukkan topik error yang Anda konfigurasi sebelumnya (misalnya, **sitewise/windfarm/rule/error**) dan pilih Subscribe to topic.



- Perhatikan pesan kesalahan muncul dan kemudian perluas `failures` array dalam pesan kesalahan apa pun.

Selanjutnya, bandingkan pesan kesalahan dengan kesalahan dalam topik ini untuk memecahkan masalah Anda.

Memecahkan masalah

Gunakan informasi berikut untuk memecahkan masalah aturan.

Masalah

- [Kesalahan: Anggota harus berada dalam 604800 detik sebelum dan 300 detik setelah stempel waktu saat ini](#)
- [Kesalahan: Nilai properti tidak cocok dengan tipe data <type>](#)
- [Kesalahan: Pengguna: <role-arn>tidak berwenang untuk melakukan: iotsitewise: pada sumber daya BatchPutAssetPropertyValue](#)
- [Kesalahan: iot.amazonaws.com tidak dapat melakukan: sts: on resource: AssumeRole <role-arn>](#)
- [Info: Tidak ada permintaan yang dikirim. PutAssetPropertyValueEntries kosong setelah melakukan templat substitusi.](#)

Kesalahan: Anggota harus berada dalam 604800 detik sebelum dan 300 detik setelah stempel waktu saat ini

Stempel waktu Anda lebih tua dari 7 hari atau lebih baru dari 5 menit, dibandingkan dengan waktu epoch Unix saat ini. Coba yang berikut ini:

- Periksa apakah stempel waktu Anda dalam waktu Unix epoch (UTC). Jika Anda memberikan stempel waktu dengan zona waktu yang berbeda, Anda menerima kesalahan ini.
- Periksa apakah stempel waktu Anda dalam hitungan detik. AWS IoT SiteWise mengharapkan stempel waktu dibagi menjadi waktu dalam hitungan detik (dalam waktu epoch Unix) dan diimbangi dalam nanodetik.
- Periksa apakah Anda mengunggah data yang diberi stempel waktu selambat-lambatnya 7 hari sebelumnya.

Kesalahan: Nilai properti tidak cocok dengan tipe data <type>

Entri dalam tindakan aturan Anda memiliki tipe data yang berbeda dari properti aset target. Misalnya, properti aset target Anda adalah DOUBLE dan tipe data yang Anda pilih adalah Integer atau Anda meneruskan nilainya. `integerValue` Coba yang berikut ini:

- Jika Anda mengonfigurasi aturan dari AWS IoT konsol, periksa apakah Anda memilih tipe Data yang benar untuk setiap entri.
- Jika Anda mengonfigurasi aturan dari API atau AWS Command Line Interface (AWS CLI), periksa apakah `value` objek Anda menggunakan bidang tipe yang benar (misalnya, `doubleValue` untuk DOUBLE properti).

Kesalahan: Pengguna: <role-arn>tidak berwenang untuk melakukan: `iotsitewise:BatchPutAssetPropertyValue` pada sumber daya `BatchPutAssetPropertyValue`

Aturan Anda tidak diizinkan untuk mengakses properti aset target, atau properti aset target tidak ada. Coba yang berikut ini:


- Periksa apakah alias properti Anda benar dan bahwa Anda memiliki properti aset dengan alias properti yang diberikan. Untuk informasi selengkapnya, lihat [Memetakan aliran data industri ke properti aset](#).
- Pastikan aturan Anda memiliki peran dan peran tersebut mengizinkan `iotsitewise:BatchPutAssetPropertyValue` izin ke properti aset yang ditargetkan, seperti melalui hierarki aset target. Untuk informasi selengkapnya, lihat [Memberikan akses AWS IoT yang diperlukan](#).

Kesalahan: `iot.amazonaws.com` tidak dapat melakukan: `sts: on resource: AssumeRole <role-arn>`

Pengguna Anda tidak diizinkan untuk mengambil peran pada aturan Anda di AWS Identity and Access Management (IAM).

Periksa apakah pengguna Anda diizinkan `iam:PassRole` izin untuk peran pada aturan Anda. Untuk informasi selengkapnya, lihat [Meneruskan izin peran](#) di Panduan AWS IoT Pengembang.

Info: Tidak ada permintaan yang dikirim. `PutAssetPropertyValueEntries` kosong setelah melakukan templat substitusi.

 Note

Pesan ini adalah log INFO level.

Permintaan Anda harus memiliki setidaknya satu entri dengan semua parameter yang diperlukan.

Periksa apakah parameter aturan Anda, termasuk templat substitusi, menghasilkan nilai yang tidak kosong. Templat substitusi tidak dapat mengakses nilai yang ditentukan dalam AS klausa dalam pernyataan kueri aturan Anda. Untuk informasi selengkapnya, lihat [Templat substitusi](#) di Panduan AWS IoT Pengembang.

Memecahkan masalah aturan

Ikuti langkah-langkah dalam prosedur ini untuk memecahkan masalah aturan Anda jika data penggunaan CPU dan memori tidak muncul AWS IoT SiteWise seperti yang diharapkan. Dalam prosedur ini, Anda mengonfigurasi tindakan aturan penerbitan ulang sebagai tindakan kesalahan untuk melihat pesan kesalahan di klien pengujian MQTT. Anda juga dapat mengonfigurasi logging ke CloudWatch Log untuk memecahkan masalah. Untuk informasi selengkapnya, lihat [Memecahkan masalah tindakan aturan AWS IoT SiteWise](#).

Untuk menambahkan tindakan kesalahan penerbitan ulang ke aturan

1. Navigasikan ke [konsol AWS IoT](#) tersebut.
2. Di panel navigasi kiri, pilih Perutean pesan, lalu pilih Aturan.
3. Pilih aturan yang Anda buat sebelumnya dan pilih Edit.

4. Di bawah Tindakan kesalahan - opsional, pilih Tambahkan tindakan kesalahan.
5. Pilih Menerbitkan ulang pesan ke suatu AWS IoT topik.
6. Di Topik, masukkan jalur ke kesalahan Anda (misalnya, **sitewise/rule/tutorial/error**). AWS IoT Core akan menerbitkan ulang pesan kesalahan ke topik ini.
7. Pilih peran yang Anda buat sebelumnya (misalnya, SiteWiseTutorialDeviceRuleRole).
8. Pilih Perbarui.

Setelah mengatur tindakan kesalahan penerbitan ulang, Anda dapat melihat pesan kesalahan di klien pengujian MQTT di AWS IoT Core

Dalam prosedur berikut, Anda berlangganan topik kesalahan di klien pengujian MQTT.

Untuk berlangganan topik tindakan kesalahan

1. Navigasikan ke [konsol AWS IoT](#) tersebut.
2. Di halaman navigasi kiri, pilih klien pengujian MQTT untuk membuka klien pengujian MQTT.
3. Di bidang Filter topik, masukkan **sitewise/rule/tutorial/error** dan pilih Berlangganan.

Saat pesan kesalahan muncul, lihat `failures` larik dalam pesan kesalahan apa pun untuk mendiagnosis masalah. Untuk informasi selengkapnya tentang kemungkinan masalah dan cara mengatasinya, lihat [Memecahkan masalah tindakan aturan AWS IoT SiteWise](#).

Jika kesalahan tidak muncul, periksa apakah aturan Anda diaktifkan dan Anda berlangganan topik yang sama dengan yang Anda konfigurasi dalam tindakan kesalahan penerbitan ulang. Jika kesalahan masih tidak muncul setelah Anda melakukannya, periksa apakah skrip perangkat berjalan dan memperbarui bayangan perangkat dengan sukses.

Note

Anda juga dapat berlangganan topik pembaruan bayangan perangkat Anda untuk melihat muatan yang diuraikan AWS IoT SiteWise tindakan Anda. Untuk melakukannya, berlangganan topik berikut.

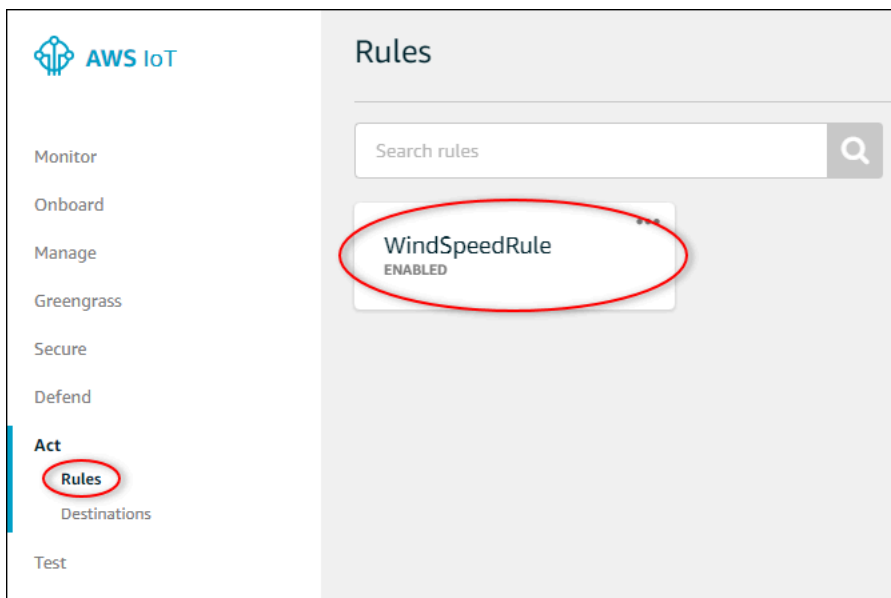
```
$aws/things/+/shadow/update/accepted
```

Memecahkan masalah aturan

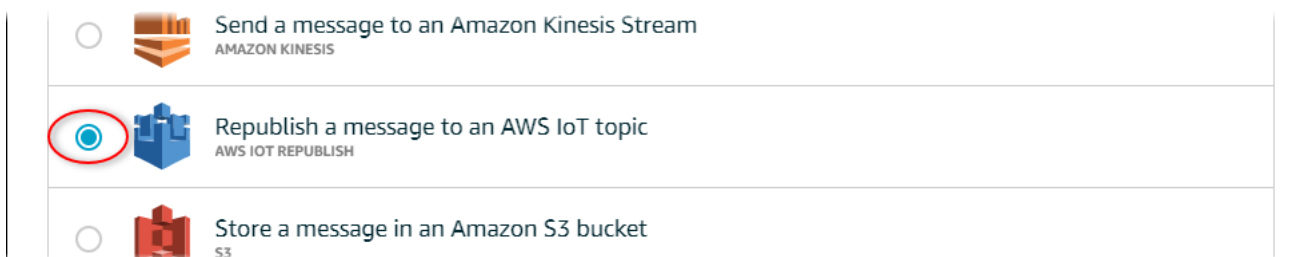
Ikuti langkah-langkah dalam prosedur ini untuk memecahkan masalah aturan Anda jika data aset demo tidak muncul di tabel DynamoDB seperti yang diharapkan. Dalam prosedur ini, Anda mengonfigurasi tindakan aturan penerbitan ulang sebagai tindakan kesalahan untuk melihat pesan kesalahan di klien pengujian MQTT. Anda juga dapat mengonfigurasi logging ke CloudWatch Log untuk memecahkan masalah. Untuk informasi selengkapnya, lihat [Memantau dengan CloudWatch Log](#) di Panduan AWS IoT Pengembang.

Untuk menambahkan tindakan kesalahan penerbitan ulang ke aturan

1. Navigasikan ke [konsol AWS IoT](#) tersebut.
2. Di panel navigasi kiri, pilih Bertindak lalu pilih Aturan.
3. Pilih aturan yang Anda buat sebelumnya.



4. Di bawah Tindakan kesalahan, pilih Tambah tindakan.
5. Pilih Menerbitkan ulang pesan ke suatu AWS IoT topik.



6. Pilih Konfigurasi tindakan di bagian bawah halaman.

- Di Topik, masukkan **windspeed/error**. AWS IoT Core akan menerbitkan ulang pesan kesalahan ke topik ini.

Configure action

Republish a message to an AWS IoT topic
AWS IOT REPUBLISH

This action will republish the message to another AWS IoT topic.

*Topic

windspeed/error

Quality of Service

0 - The message is delivered zero or more times.

1 - The message is delivered one or more times.

Choose or create a role to grant AWS IoT access to perform this action.

No role selected Create Role **Select**

Cancel Add action

- Pilih Pilih untuk memberikan akses AWS IoT Inti untuk melakukan tindakan kesalahan menggunakan peran yang Anda buat sebelumnya.
- Pilih Pilih di sebelah peran Anda.

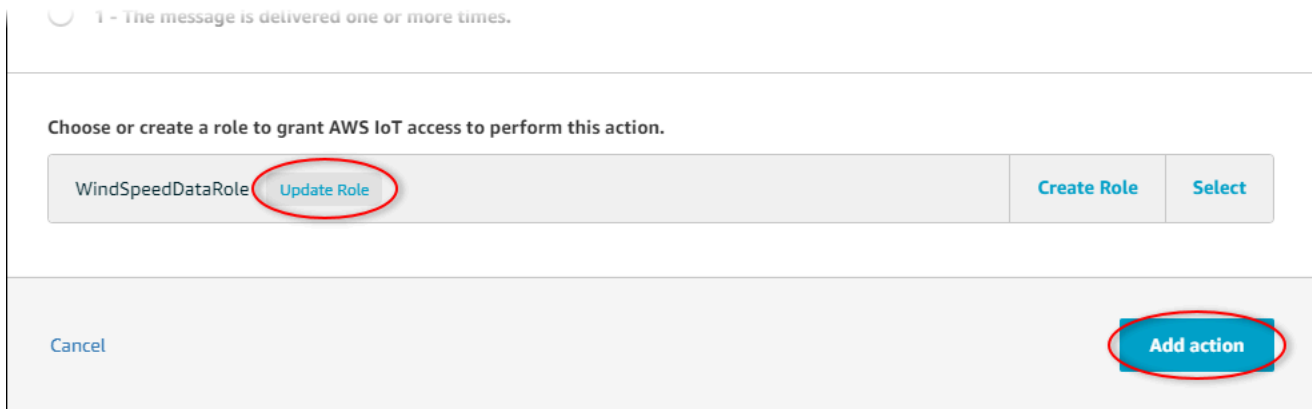
Choose or create a role to grant AWS IoT access to perform this action.

No role selected Refresh Create Role Close

Search for IAM roles

WindSpeedDataRole **Select**

- Pilih Perbarui Peran untuk menambahkan izin tambahan ke peran.



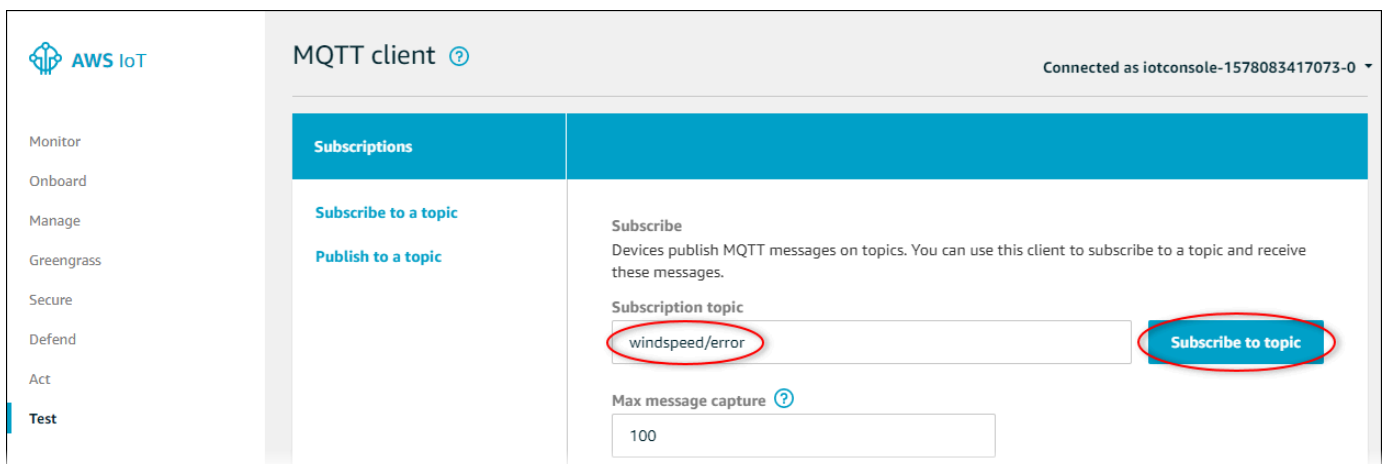
11. Pilih Tambahkan tindakan untuk menyelesaikan penambahan tindakan kesalahan.
12. Pilih panah belakang di kiri atas konsol untuk kembali ke rumah konsol AWS IoT Core.

Setelah mengatur tindakan kesalahan penerbitan ulang, Anda dapat melihat pesan kesalahan di klien pengujian MQTT di Core. AWS IoT

Dalam prosedur berikut, Anda berlangganan topik kesalahan di klien pengujian MQTT.

Untuk berlangganan topik tindakan kesalahan

1. Di halaman navigasi kiri konsol AWS IoT inti, pilih Uji.
2. Di bidang Subscription topic, masukkan **windspeed/error** dan pilih Subscribe to topic.



3. Perhatikan pesan kesalahan muncul dan jelajahi failures larik dalam pesan kesalahan untuk mendiagnosis masalah umum berikut:
 - Kesalahan ketik dalam pernyataan kueri aturan
 - Izin peran tidak memadai

Jika kesalahan tidak muncul, periksa apakah aturan Anda diaktifkan dan Anda berlangganan topik yang sama dengan yang Anda konfigurasi dalam tindakan kesalahan penerbitan ulang. Jika kesalahan masih tidak muncul, periksa apakah aset ladang angin demo Anda masih ada dan Anda mengaktifkan notifikasi pada properti kecepatan angin. Jika aset demo Anda kedaluwarsa dan hilang AWS IoT SiteWise, Anda dapat membuat demo baru dan memperbarui pernyataan kueri aturan untuk mencerminkan model aset dan ID properti yang diperbarui.

AWS IoT SiteWise titik akhir dan kuota

Bagian berikut menjelaskan titik akhir dan kuota untuk AWS IoT SiteWise

Daftar Isi

- [AWS IoT SiteWise titik akhir](#)
- [AWS IoT SiteWise kuota](#)

AWS IoT SiteWise titik akhir

Untuk terhubung secara terprogram ke AWS IoT SiteWise, Anda menggunakan titik akhir. AWS SDK dan AWS Command Line Interface (AWS CLI) secara otomatis menggunakan endpoint default di Region. Untuk informasi selengkapnya tentang Wilayah AWS IoT SiteWise yang tersedia, lihat [AWS IoT SiteWise titik akhir dan kuota](#) di Referensi Umum AWS

AWS IoT SiteWise mendukung titik akhir berikut.

Gunakan titik akhir ini untuk mengakses operasi API bidang data berikut:

[BatchPutAssetPropertyValue](#), [GetAssetPropertyAggregatesGetAssetPropertyValue](#), [GetAssetPropertyValueHistory](#) dan [GetInterpolatedAssetPropertyValues](#). Ganti *region* dengan AWS wilayah Anda.

AWS IoT SiteWise menawarkan titik akhir konsolidasi ini untuk operasi API bidang kontrol yang Anda gunakan untuk mengelola model aset, aset, gateway SiteWise Edge, tag, dan konfigurasi akun. Ganti *region* dengan AWS wilayah Anda.

Note

- Secara default, AWS IoT SiteWise gunakan titik akhir konsolidasi saat Anda melakukan panggilan ke operasi API bidang kontrol yang didukung.
- Kami menyarankan Anda menggunakan titik akhir konsolidasi untuk operasi API bidang kontrol yang didukung.
- Anda tidak dapat menggunakan titik akhir konsolidasi untuk mengakses operasi SiteWise Monitor API.

Operasi API bidang kontrol yang didukung meliputi [AssociateAssets](#), [CreateAsset](#), [CreateAssetModel](#), [DeleteAsset](#), [DeleteAssetModelDeleteDashboard](#), [DescribeAsset](#), [DescribeAssetModel](#), [DescribeAssetProperty](#), [DescribeDashboard](#), [DescribeLoggingOptions](#), [DisassociateAssets](#), [ListAssetModels](#), [ListAssetRelationships](#), [ListAssets](#), [ListAssociatedAssets](#), [PutLoggingOptions](#), [UpdateAsset](#), [UpdateAssetModel](#), [UpdateAssetProperty](#), [CreateGateway](#), [DeleteGateway](#), [DescribeGateway](#), [DescribeGatewayCapabilityConfiguration](#), [ListGateways](#), [UpdateGateway](#), [UpdateGatewayCapabilityConfiguration](#), [DescribeStorageConfiguration](#), [PutStorageConfiguration](#), [DescribeDefaultEncryptionConfiguration](#), [ListTagsForResource](#), [PutDefaultEncryptionConfiguration](#), [TagResource](#), dan [UntagResource](#).

Titik akhir VPC antarmuka untuk operasi API bidang kontrol hanya mendukung titik akhir konsolidasi. Untuk informasi selengkapnya, lihat [Titik akhir VPC](#).

Gunakan titik akhir ini untuk mengakses operasi API berikut: [DescribeStorageConfiguration](#), [PutStorageConfiguration](#), [DescribeDefaultEncryptionConfiguration](#), [ListTagsForResource](#), [PutDefaultEncryptionConfigurationTagResource](#), dan [UntagResource](#). Ganti *region* dengan AWS wilayah Anda.

Gunakan titik akhir ini untuk mengakses operasi API berikut: [AssociateAssets](#), [CreateAssetCreateAssetModel](#), [DeleteAsset](#), [DeleteAssetModel](#), [DeleteDashboard](#), [DescribeAsset](#), [DescribeAssetModel](#), [DescribeAssetProperty](#), [DescribeDashboard](#), [DescribeLoggingOptions](#), [DisassociateAssets](#), [ListAssetModels](#), [ListAssetRelationships](#), [ListAssets](#), [ListAssociatedAssets](#), [PutLoggingOptionsUpdateAsset](#) https://docs.aws.amazon.com/iot-sitewise/latest/APIReference/API_UpdateAsset.html, [UpdateAssetModel](#), dan [UpdateAssetProperty](#). Ganti *region* dengan AWS wilayah Anda.

Gunakan titik akhir ini untuk mengakses operasi API berikut: [CreateGateway](#), [DeleteGateway](#), [DescribeGateway](#), [DescribeGatewayCapabilityConfiguration](#), [ListGatewaysUpdateGateway](#), dan [UpdateGatewayCapabilityConfiguration](#). Ganti *region* dengan AWS wilayah Anda.

Gunakan titik akhir ini untuk mengakses operasi API berikut: [BatchAssociateProjectAssets](#), [BatchDisassociateProjectAssetsCreateAccessPolicy](#), [CreateDashboard](#), [CreatePortal](#), [CreateProject](#), [DeleteAccessPolicy](#), [DeletePortal](#), [DeleteProject](#), [DescribeAccessPolicy](#), [DescribePortal](#), [DescribeProject](#), [ListAccessPolicies](#), [ListDashboards](#), [ListPortals](#), [ListProjectAssets](#), [ListProjectsUpdateAccessPolicy](#) https://docs.aws.amazon.com/iot-sitewise/latest/APIReference/API_UpdateAccessPolicy.html, [UpdateDashboard](#), [UpdatePortal](#), dan [UpdateProject](#). Ganti *region* dengan AWS wilayah Anda.

AWS IoT SiteWise kuota

Tabel berikut menjelaskan kuota di AWS IoT SiteWise. Untuk informasi selengkapnya tentang kuota dan cara meminta kenaikan kuota, lihat [kuota AWS layanan](#) di. Referensi Umum AWS Untuk informasi lebih lanjut tentang AWS IoT SiteWise kuota, lihat [kuota AWS IoT SiteWise layanan](#) di. Referensi Umum AWS

Kuota untuk aset dan model aset

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Jumlah model aset per Wilayah per AWS akun	1000	Ya	
Jumlah aset per model aset	10.000	Ya	
Jumlah aset anak per aset induk	2000	Ya	
Kedalaman pohon hierarki model aset	30	Ya	
Jumlah definisi hierarki per model aset	30	Ya	
Jumlah properti di tingkat root per model aset	500	Ya	Jumlah maksimum ini <code>assetMode</code> <code>lPropert</code> <code>es</code> untuk setiap model aset. Hitungan ini tidak termasuk <code>composite</code> <code>ModelProp</code> <code>erties</code> . Kuota ini juga berlaku untuk setiap aset unik yang

Sumber daya	Kuota	Dapat disesuaikan	Catatan
			dibuat dari model aset ini.
Jumlah properti per model aset	5000	Ya	Jumlah maksimum properti dari model aset jenis ASSET_MODEL atau COMPONENT_MODEL . Angka ini ditentukan dengan menggabungkan properti model aset akar dan model komposit yang disertakan component-model-based atau sebaris. Kuota ini juga berlaku untuk setiap aset unik yang dibuat dari model aset ini.
Jumlah properti per model komposit	100	Ya	Jumlah maksimum properti yang diizinkan untuk model komposit. Juga, jumlah maksimum properti yang diizinkan untuk model aset tipe COMPONENT_MODEL .

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Kedalaman pohon properti per model aset	10	Tidak	Misalnya, model dengan properti transformasi C yang mengkonsumsi properti transformasi B yang mengkonsumsi properti pengukuran A memiliki kedalaman 3.
Jumlah model aset per pohon hierarki	100	Ya	

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Jumlah properti yang bergantung langsung per model aset	20	Tidak	Kuota ini membatasi berapa banyak properti yang dapat secara langsung bergantung pada satu properti, seperti yang didefinisikan dalam ekspresi rumus properti. Jumlah properti dependen untuk model aset harus lebih besar dari Jumlah properti yang bergantung langsung per model aset. Anda harus meminta peningkatan kuota untuk keduanya jika batas untuk Jumlah properti yang bergantung langsung per model aset lebih besar dari batas untuk Jumlah properti dependen per model aset.

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Jumlah properti dependen per model aset	30	Tidak	Kuota ini membatasi berapa banyak properti yang dapat secara langsung atau tidak langsung bergantung pada satu properti, seperti yang didefinisikan dalam ekspresi rumus properti.
Jumlah model komposit per model aset	50	Ya	Jumlah maksimum model komposit diperbolehkan pada model aset tunggal.
Kedalaman model komposit	2	Ya	Kedalaman maksimum pohon model komposit per model aset, termasuk model inline dan component-model-based komposit.
Jumlah model aset unik yang menggunakan model komponen yang sama	20	Ya	Jumlah maksimum model aset unik yang memiliki setidaknya satu model component-model-based komposit yang secara langsung mereferensikan model aset tertentu dari tipe COMPONENT_MODEL.

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Jumlah variabel properti per ekspresi rumus properti	10	Tidak	Misalnya, ada dua variabel properti, <code>power</code> dan <code>temp</code> , dalam ekspresi <code>avg(power) + max(temp)</code> . Ini juga berlaku untuk hasil komputasi transformasi.
Jumlah fungsi per ekspresi rumus properti	10	Tidak	Misalnya, ada dua fungsi, <code>avg</code> dan <code>max</code> , dalam ekspresi <code>avg(power) + max(temp)</code> .

Kuota untuk data properti aset

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Tingkat permintaan untuk operasi API data properti aset	1000 permintaan per detik per Wilayah per AWS akun	Ya	Kuota ini berlaku untuk operasi API seperti <code>GetAssetPropertyValue</code> dan <code>BatchPutAssetPropertyValue</code> .
Jumlah titik data per detik per kualitas data per properti aset	10 titik data	Tidak	Kuota ini berlaku untuk jumlah maksimum titik data timestamp-quality-value (TQV) dengan stempel waktu yang sama dalam hitungan

Sumber daya	Kuota	Dapat disesuaikan	Catatan
			detik per kualitas data untuk setiap properti aset. Anda dapat menyimpan hingga jumlah titik data berkualitas baik, tidak pasti, dan berkualitas buruk ini untuk setiap detik tertentu untuk setiap properti aset.
Jumlah BatchPutAssetPropertyEntry yang dicerna per detik per properti aset per Wilayah per AWS Akun.	10 entri per properti aset	Tidak	Kuota ini berlaku untuk BatchPutAssetPropertyEntry entri dari semua sumber, termasuk gateway SiteWise Edge, AWS IoT Core aturan, dan panggilan API.
Tingkat titik data tertelan	5000 titik data per detik per Wilayah per AWS akun	Ya	Titik data TimestampQualityValue (TQV).
Tarif permintaan untuk BatchGetAssetPropertyAggregates	200	Ya	Jumlah maksimum BatchGetAssetPropertyAggregates permintaan per detik yang dapat Anda lakukan di akun ini di Wilayah saat ini.

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Tarif permintaan untuk BatchGetAssetPropertyValue	500	Ya	Jumlah maksimum BatchGetAssetPropertyValue permintaan per detik yang dapat Anda lakukan di akun ini di Wilayah saat ini.
Tarif permintaan untuk BatchGetAssetPropertyValueHistory	200	Ya	Jumlah maksimum BatchGetAssetPropertyValueHistory permintaan per detik yang dapat Anda lakukan di akun ini di Wilayah saat ini.
Jumlah BatchPutAssetPropertyValue entri yang dicerna per detik per properti aset per Wilayah per AWS Akun.	10 entri per properti aset	Tidak	Kuota ini berlaku untuk BatchPutAssetPropertyValue entri dari semua sumber, termasuk gateway SiteWise Edge, AWS IoT Core aturan, dan panggilan API.

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Tingkat GetAssetPropertyAggregates permintaan dan kueri BatchGetAssetPropertyAggregates entri per properti aset	50	Tidak	Jumlah maksimum total GetAssetPropertyAggregates permintaan dan BatchGetAssetPropertyAggregates entri untuk setiap properti aset per detik di akun ini di Wilayah saat ini.
Tingkat GetAssetPropertyValue permintaan dan kueri BatchGetAssetPropertyValue entri per properti aset	500	Tidak	Jumlah maksimum total GetAssetPropertyValue permintaan dan BatchGetAssetPropertyValue entri untuk setiap properti aset per detik di akun ini di Wilayah saat ini.

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Tingkat GetAssetPropertyValueHistory permintaan dan kueri BatchGetAssetPropertyValueHistory entri per properti aset	30	Tidak	Jumlah maksimum total GetAssetPropertyValueHistory permintaan dan BatchGetAssetPropertyValueHistory entri untuk setiap properti aset per detik di akun ini di Wilayah saat ini.
Permintaan tarif GetInterpolatedAssetPropertyValues	500	Ya	Jumlah maksimum GetInterpolatedAssetPropertyValues permintaan per detik yang dapat Anda lakukan di akun ini di Wilayah saat ini.
Jumlah hasil per GetInterpolatedAssetPropertyValues permintaan	10	Ya	Jumlah maksimum hasil yang akan dikembalikan per GetInterpolatedAssetPropertyValues permintaan paginasi.

Sumber daya	Kuota	Dapat disesuaikan	Catatan
Tingkat titik data yang diambil dari dan GetAssetPropertyValueHistory BatchGetAssetPropertyValueHistory	Respons baca 100MB per detik per Wilayah per AWS akun.	Ya	<p>Kecepatan byte maksimum (MB/detik) titik data yang diambil per detik per Wilayah per akun di seluruh dan. AWS GetAssetPropertyValueHistory BatchGetAssetPropertyValueHistory Payload respons yang dievaluasi untuk kuota ini menggunakan bidang Timestamp-Quality-Value (TQV) untuk setiap titik data dan membulatkan ukuran byte untuk setiap permintaan API ke kenaikan 4KB berikutnya.</p> <p>Titik data Timestamp-quality-value (TQV) yang diambil per detik bervariasi sesuai tipe data:</p> <ul style="list-style-type: none"> • Integer - hingga 5 Juta TQV per detik • Ganda - hingga 4 Juta TQV per detik

Sumber daya	Kuota	Dapat disesuaikan	Catatan
			<ul style="list-style-type: none"> • Boolean — hingga 6 Juta TQV per detik • String - bervariasi berdasarkan setiap ukuran nilai string.

Kuota untuk gateway SiteWise Edge

Sumber daya	Kuota	Dapat disesuaikan
Jumlah gateway SiteWise Edge per Wilayah per akun AWS	100	Ya
Jumlah sumber OPC-UA per SiteWise gateway Edge	100	Tidak

Kuota untuk AWS IoT SiteWise Monitor

Sumber daya	Kuota	Dapat disesuaikan
Jumlah portal per Wilayah per akun AWS	100	Ya
Jumlah proyek per portal	100	Ya
Jumlah dasbor per proyek	100	Ya
Jumlah aset root per proyek	1	Tidak
Jumlah visualisasi per dasbor	10	Ya
Jumlah metrik per visualisasi dasbor	5	Ya

Sumber daya	Kuota	Dapat disesuaikan
Jumlah ambang batas per visualisasi dasbor	12	Tidak

Kuota untuk impor AWS IoT SiteWise massal dan ekspor metadata

Sumber Daya	Deskripsi	Kuota	Dapat disesuaikan
Jumlah pekerjaan transfer metadata dalam antrian	Jumlah maksimum pekerjaan transfer PENDING metadata dalam antrian.	10	Ya
Ukuran file impor pekerjaan transfer metadata	Ukuran maksimum file yang diimpor (dalam MB).	100 MB	Ya
AWS IoT SiteWise kuota sumber daya untuk pekerjaan transfer metadata	Jumlah maksimum sumber daya yang diimpor atau diekspor dalam satu pekerjaan . Sumber daya mencakup aset, dan model aset.	5000	Tidak

Kuota untuk impor data AWS IoT SiteWise massal

Sumber daya	Kuota	Dapat disesuaikan
Jumlah pekerjaan impor massal yang berjalan	100	Tidak
Ukuran file CSV	10 GB	Tidak
Ukuran file parket yang tidak terkompresi	270 MB	Tidak

Sumber daya	Kuota	Dapat disesuaikan
Ukuran CSV file untuk konsumsi buffer	270 MB	Tidak
Ukuran kelompok baris paket yang tidak terkompresi	64 MB	Tidak
Jumlah pengukuran unik per kelompok baris paket	2000	Ya
Jumlah hari antara stempel waktu di masa lalu dan hari ini untuk konsumsi buffer	30	Ya
Tarif permintaan <code>CreateBulkImportJobs</code> untuk setiap Wilayah di setiap AWS akun	10	Ya
Tarif permintaan <code>ListBulkImportJobs</code> untuk setiap Wilayah di setiap AWS akun	50	Ya
Tarif permintaan <code>DescribeBulkImportJobs</code> untuk setiap Wilayah di setiap AWS akun	50	Ya

Kuota untuk deteksi anomali

Kuota untuk deteksi anomali dibagi antara dan AWS IoT SiteWise Amazon Lookout for Equipment. Untuk informasi selengkapnya, lihat [Kuota untuk menggunakan Lookout for Equipment](#).

Riwayat dokumen untuk Panduan AWS IoT SiteWise Pengguna

Tabel berikut menjelaskan dokumentasi untuk rilis ini AWS IoT SiteWise.

- Versi API: 2019-12-02

Perubahan	Deskripsi	Tanggal
Menambahkan dukungan untuk menjalankan SiteWise Edge di Siemens Industrial Edge	AWS IoT SiteWise sekarang mendukung menjalankan SiteWise Edge pada perangkat Siemens Industrial Edge.	26 November 2023
Menambahkan dukungan untuk penyimpanan tingkat hangat	AWS IoT SiteWise sekarang mendukung penyimpanan hangat, tingkat penyimpanan yang dikelola sepenuhnya yang memudahkan pelanggan untuk menyimpan dan mengakses data industri dengan aman.	15 November 2023
Menambahkan dukungan untuk pengidentifikasi unik yang ditentukan pengguna	AWS IoT SiteWise sekarang mendukung penggunaan pengidentifikasi unik yang ditentukan pengguna untuk aset, model aset, properti, dan hierarki.	15 November 2023
Menambahkan dukungan untuk deteksi anomali multi variat aset industri	AWS IoT SiteWise sekarang mendukung deteksi anomali multi variasi aset industri dengan integrasi data peralatan historis dan waktu	15 November 2023

	nyata dengan Amazon Lookout for Equipment.	
<u>Menambahkan dukungan untuk konsumsi data deret waktu yang hemat biaya dan terukur di AWS IoT SiteWise</u>	AWS IoT SiteWise sekarang mendukung konsumsi data deret waktu yang efisien dan terukur yang diperlukan untuk kasus penggunaan analitis.	15 November 2023
<u>Ditambahkan dukungan untuk impor massal, ekspor, dan update</u>	AWS IoT SiteWise sekarang mendukung impor massal, ekspor, dan pembaruan metadata peralatan industri.	15 November 2023
<u>Menambahkan dukungan untuk komponen model aset</u>	AWS IoT SiteWise sekarang mendukung komponen model Aset untuk membantu pelanggan industri membuat komponen yang dapat digunakan kembali.	15 November 2023
<u>Menambahkan dukungan untuk aplikasi dasbor IoT</u>	AWS IoT SiteWise sekarang mendukung aplikasi dashboard open source di mana Anda dapat memvisualisasikan dan berinteraksi dengan data operasional.	15 November 2023
<u>Memperbarui peran terkait layanan untuk AWS IoT SiteWise</u>	AWS IoT SiteWise memiliki peran terkait layanan baru, dan dapat menjalankan kueri penelusuran metadata, terhadap database. AWS IoT TwinMaker	6 November 2023
<u>Penandaan yang diperbarui untuk sumber AWS IoT SiteWise daya aliran data</u>	Menambahkan dukungan untuk menandai sumber daya aliran data.	18 Agustus 2022

Gerbang SiteWise Edge yang Diperbarui	Anda sekarang dapat mengonfigurasi penerbit untuk mengontrol data apa yang dikirim dari tepi ke cloud dan urutan yang dikirim ke cloud.	12 Januari 2022
Diperbarui AWS IoT SiteWise demo	Anda sekarang dapat menggunakan demo untuk membuat portal SiteWise Monitor.	10 Januari 2022
Manajemen penyimpanan yang diperbarui	Anda sekarang dapat menentukan periode retensi untuk mengontrol berapa lama data Anda disimpan di tingkat panas.	29 November 2021
Menambahkan dukungan untuk manajemen aliran data	Anda sekarang dapat mengambil data AWS IoT SiteWise sebelum Anda membuat model aset dan aset.	24 November 2021
Hirarki model aset yang diperbarui	Model aset anak sekarang dapat dikaitkan dengan beberapa model aset induk.	28 Oktober 2021
Peluncuran wilayah	Diluncurkan AWS IoT SiteWise di AWS GovCloud (AS-Barat).	29 September 2021

Fungsi yang diperbarui	Ditambahkan fitur berikut <ul style="list-style-type: none">• Dalam metrik, Anda dapat menggunakan ekspresi bersarang dalam fungsi agregasi dan fungsi temporal.• Dalam transformasi, Anda dapat menggunakan fungsi pretrigger () untuk mengambil nilai variabel sebelum pembaruan properti yang memicu perhitungan transformasi saat ini.	Agustus 10, 2021
Interval waktu metrik khusus	Menambahkan dukungan untuk interval waktu khusus dan offset dalam metrik.	Agustus 3, 2021
Menggunakan AWS IoT SiteWise di tepi	Fitur pemrosesan tepi sekarang tersedia secara umum.	29 Juli 2021
Mengekspor data ke Amazon S3	AWS IoT SiteWise sekarang dapat mengekspor data ke Amazon S3.	27 Juli 2021
Titik akhir VPC (AWS PrivateLink)	Titik akhir VPC antarmuka untuk operasi API bidang kontrol sekarang tersedia secara umum.	15 Juli 2021
Mengubah	Transformasi sekarang dapat memasukkan beberapa variabel properti aset.	8 Juli 2021

Memperbarui fungsi timestamp()	Dalam transformasi, Anda sekarang dapat memberikan variabel sebagai argumen untuk timestamp() fungsi.	16 Juni 2021
Alarm ketersediaan umum	Fitur alarm sekarang tersedia secara umum.	27 Mei 2021
Adaptor Protokol Modbus-TCP versi 2 dirilis	Versi 2 dari konektor Adaptor Protokol Modbus-TCP tersedia . Rilis ini menambahkan dukungan untuk string sumber yang dikodekan ASCII, UTF8, dan ISO8859.	24 Mei 2021
Kuota layanan yang diperbarui	Menambahkan kuota berikut untuk GetInterpolatedAssetPropertyValues API: tingkat GetInterpolatedAssetPropertyValues permintaan, jumlah hasil per GetInterpolatedAssetPropertyValues permintaan, dan jumlah hari antara tanggal mulai di masa lalu dan hari ini untuk GetInterpolatedAssetPropertyValues .	29 April 2021

[Ekspresi rumus yang diperbarui](#)

Ditambahkan operator dan fungsi berikut:

22 April 2021

- Ditambahkan [operator](#) berikut: <, >, <=, >=, ==, !=, !=, and, or, dan not.
- Ditambahkan [fungsi perbandingan](#) berikut: neq(x, y).
- Ditambahkan [fungsi string](#) berikut: join(), format(), dan f''.

[Titik akhir VPC \(AWS PrivateLink\)](#)

Menambahkan informasi tentang cara membuat koneksi pribadi antara virtual private cloud (VPC) Anda dan API bidang AWS IoT SiteWise kontrol dengan membuat titik akhir VPC antarmuka.

16 Maret 2021

[Federasi IAM](#)

Administrator dan pengguna portal SiteWise Monitor Anda sekarang dapat masuk ke portal yang ditetapkan dengan kredensial IAM mereka.

16 Maret 2021

[Peluncuran wilayah](#)

Diluncurkan AWS IoT SiteWise di China (Beijing).

3 Februari 2021

SiteWise Konektor IoT versi 10 dirilis	Versi 10 dari SiteWise konektor IoT tersedia. Rilis ini dikonfigurasi StreamManager untuk meningkatkan penanganan ketika koneksi sumber terputus dan dibuat kembali. Versi ini juga menerima nilai OPC-UA dengan sebuah ServerTimestamp ketika tidak SourceTimestamp tersedia.	22 Januari 2021
Fungsi tanggal dan waktu	AWS IoT SiteWise sekarang mendukung fungsi tanggal dan waktu.	21 Januari 2021
Sintaks fungsi	Anda sekarang dapat menggunakan Uniform Function Call Syntax (UFCS) untuk fungsi. AWS IoT SiteWise	11 Januari 2021
Integrasi dengan Grafana	Menambahkan informasi tentang cara memvisualisasikan AWS IoT SiteWise data di dasbor Grafana.	15 Desember 2020

[AWS IoT SiteWise rilis fitur](#)

15 Desember 2020

Anda sekarang dapat memantau data Anda dengan alarm, memproses data industri di edge, menggunakan sumber Modbus TCP dan Ethernet/IP ke gateway SiteWise Edge Anda, memfilter data yang masuk dengan deadband, dan banyak lagi.

- Menambahkan bagian [Monitoring data dengan alarm](#) yang dapat Anda gunakan untuk menentukan, mengonfigurasi, dan merespons alarm. AWS IoT SiteWise
- Menambahkan bagian [pemrosesan Edge](#) yang dapat Anda gunakan untuk mengonfigurasi pemrosesan data industri Anda di perangkat edge Anda.
- Menambahkan bagian [Modbus TCP dan Ethernet/IP](#) ke dokumentasi sumber gateway Edge. SiteWise
- Menambahkan bagian [tujuan sumber](#) yang dapat Anda gunakan untuk menyesuaikan tempat Anda mengirim data industri yang masuk.
- Menambahkan bagian [pemfilteran OPC-UA](#) yang

dapat Anda gunakan untuk mengontrol frekuensi dan jenis data yang dikirim ke gateway SiteWise Edge Anda dari server lokal industri Anda.

[AWS IoT SiteWise sekarang mendukung CMK yang dikelola pelanggan.](#)

AWS IoT SiteWise sekarang mendukung enkripsi dengan CMK yang dikelola pelanggan.

24 November 2020

[SiteWise Konektor IoT versi 8 dirilis](#)

Versi 8 dari SiteWise konektor IoT tersedia. Rilis ini meningkatkan stabilitas ketika konektor mengalami konektivitas jaringan intermiten.

19 November 2020

[Menggunakan string dan kondisional dalam ekspresi rumus](#)

Menambahkan informasi tentang cara string dan fungsi kondisional dalam ekspresi rumus untuk transformasi dan metrik.

16 November 2020

[Menelan data menggunakan pengelola AWS IoT Greengrass aliran](#)

Menambahkan informasi tentang cara menelan data IoT volume tinggi dari sumber data lokal menggunakan perangkat edge. AWS IoT Greengrass

16 September 2020

[Titik akhir VPC \(AWS PrivateLink\)](#)

Menambahkan informasi tentang cara membuat koneksi pribadi antara virtual private cloud (VPC) Anda dan API AWS IoT SiteWise data dengan membuat titik akhir VPC antarmuka.

Selasa, 04 September 2020

SiteWise Konektor IoT versi 7 dirilis	Versi 7 dari SiteWise konektor IoT tersedia. Rilis ini memperbaiki masalah dengan metrik gateway SiteWise Edge.	14 Agustus 2020
Membuat pengguna IAM Identity Center dari konsol AWS IoT SiteWise	Menambahkan informasi tentang cara membuat pengguna IAM Identity Center di AWS IoT SiteWise konsol. Anda sekarang dapat membuat pengguna IAM Identity Center ketika Anda menetapkan pengguna ke portal baru atau yang sudah ada. Memperbarui tutorial Visualisasi dan berbagi data ladang angin untuk menggunakan fitur ini. Perubahan ini mengurangi jumlah langkah dalam tutorial.	4 Agustus 2020
Pemecahan masalah gateway SiteWise Edge yang ditingkatkan	Menambahkan informasi tambahan tentang cara memecahkan masalah gateway SiteWise Edge dan cara mengeksport sertifikat klien OPC-UA untuk sumber.	18 Juni 2020
Dokumentasi tugas konsol	Menambahkan dokumentasi tugas konsol untuk Pemodelan aset industri , Kueri data properti aset , dan Berinteraksi dengan layanan lain . Anda dapat mengikuti petunjuk ini untuk menyelesaikan tugas di AWS IoT SiteWise konsol.	11 Juni 2020

[Menganalisis tutorial data yang diekspor](#)

[Menambahkan tutorial yang dapat Anda ikuti untuk mempelajari cara menggunakan Amazon Athena untuk menganalisis data aset yang Anda ekspor ke S3 dengan template fitur ekspor. AWS CloudFormation](#)

27 Mei 2020

[Peningkatan menggunakan ekspresi rumus](#)

Menambahkan informasi rinci tentang perilaku properti AWS IoT SiteWise rumus dan menambahkan contoh cara menghitung titik data yang difilter.

18 Mei 2020

[SiteWise Konektor IoT versi 6 dirilis](#)

Versi 6 dari SiteWise konektor IoT tersedia. Rilis ini menambahkan dukungan untuk CloudWatch metrik dan penemuan otomatis tag OPC-UA baru. Ini berarti Anda tidak perlu memulai ulang gateway SiteWise Edge Anda saat tag berubah untuk sumber OPC-UA Anda. Versi konektor ini memerlukan manajer aliran dan perangkat lunak AWS IoT Greengrass Core v1.10.0 atau lebih tinggi.

29 April 2020

[AWS IoT SiteWise rilis fitur](#)

29 April 2020

AWS IoT SiteWise rilis fitur. Sekarang Anda dapat mengelola gateway SiteWise Edge dengan API, menambahkan logo Anda ke portal, melihat metrik gateway SiteWise Edge, dan banyak lagi.

- Menambahkan bagian [Mengekspor data ke Amazon S3](#) dengan templat AWS CloudFormation yang dapat Anda gunakan untuk mengekspor nilai data baru ke bucket S3.
- Menambahkan bagian [Mengonfigurasi sumber data](#) yang meningkatkan dokumentasi sumber gateway SiteWise Edge dan menyertakan API gateway SiteWise Edge baru.
- Menambahkan bagian [metrik gateway SiteWise Edge](#) yang menjelaskan CloudWatch metrik yang diterbitkan oleh gateway SiteWise Edge.
- Menambahkan bagian [Mengonfigurasi gateway SiteWise Edge di Amazon EC2](#) dengan templat AWS CloudFormation yang dapat Anda gunakan untuk SiteWise mengonfigurasi

dependensi gateway Edge dengan cepat pada instans Amazon EC2.

- Menambahkan bagian [peran layanan portal](#) yang menjelaskan fitur izin baru dari portal SiteWise Monitor.
- [Dokumentasi portal](#) yang diperbarui untuk peran layanan portal dan logo portal.
- Menambahkan bagian [Menandai AWS IoT SiteWise sumber daya Anda](#).
- Memperbarui bagian [Membuat dasbor \(CLI\)](#) untuk struktur definisi dasbor baru.
- Menambahkan bagian [Keamanan](#).

[Menelan data dari AWS IoT Events](#)

Menambahkan informasi tentang cara menelan data dari AWS IoT Events saat suatu peristiwa terjadi.

20 April 2020

[Memvisualisasikan dan berbagi data ladang angin di SiteWise Monitor tutorial](#)

Menambahkan tutorial yang dapat Anda ikuti untuk mempelajari cara menggunakan AWS IoT SiteWise Monitor untuk memvisualisasikan dan berbagi data aset.

12 Maret 2020

AWS IoT SiteWise konsep	Menambahkan glosarium AWS IoT SiteWise konsep yang dapat Anda gunakan untuk mempelajari tentang layanan dan istilah umum.	5 Maret 2020
Instruksi AWS IoT Greengrass instalasi dihapus	Menghapus instruksi penginstalan perangkat lunak AWS IoT Greengrass Inti dari Panduan AWS IoT SiteWise Pengguna. Panduan AWS IoT Greengrass Pengembang menawarkan skrip penyiapan perangkat dan instruksi untuk disiapkan AWS IoT Greengrass di platform lain seperti Amazon EC2 dan Docker.	14 Februari 2020
Peningkatan menelan data menggunakan aturan AWS IoT Core	Menambahkan informasi terperinci tentang cara menggunakan dan cara memecahkan masalah tindakan AWS IoT SiteWise aturan, yang dapat Anda gunakan untuk menelan data dari pesan MQTT. AWS IoT Core	14 Februari 2020
SiteWise Konektor IoT versi 5 dirilis	Versi 5 dari SiteWise konektor IoT tersedia. Rilis ini memperbaiki masalah kompatibilitas dengan perangkat lunak AWS IoT Greengrass Core v1.9.4.	12 Februari 2020

SiteWise Konektor IoT versi 4 dirilis	Versi 4 dari SiteWise konektor IoT tersedia. Rilis ini memperbaiki masalah dengan koneksi ulang server OPC-UA.	7 Februari 2020
Aset industri pemodelan yang direstrukturisasi	<p>Merestrukturisasi bagian Memperbarui Aset dan Model menjadi beberapa topik dalam Pemodelan Aset Industri.</p> <ul style="list-style-type: none">• Status aset dan model• Memetakan aliran data industri ke properti aset• Memperbarui nilai atribut• Mengasosiasikan dan memisahkan aset• Memperbarui aset dan model• Menghapus aset dan model	4 Februari 2020
Menelan data dari AWS IoT hal-hal tutorial	Menambahkan tutorial yang dapat Anda ikuti untuk mempelajari cara mengonfigurasi tindakan AWS IoT SiteWise aturan untuk menelan data dari armada baru atau yang sudah ada AWS IoT .	4 Februari 2020
Merestrukturisasi mengambil data dari AWS IoT SiteWise	Merestrukturisasi bagian Mengambil Data menjadi dua bagian tingkat atas: Mengkueri nilai properti aset dan agregat dan Berinteraksi dengan layanan lain. AWS	21 Januari 2020

Menerbitkan pembaruan nilai properti ke tutorial Amazon DynamoDB	Menambahkan tutorial yang dapat Anda ikuti untuk mempelajari cara menggunakan pemberitahuan nilai properti untuk menyimpan data aset di DynamoDB.	8 Januari 2020
Menggunakan ekspresi rumus	Menambahkan referensi ekspresi rumus untuk mengatur konstanta dan fungsi yang tersedia untuk digunakan dalam properti transformasi dan metrik. Properti Aset yang direstrukturisasi menjadi topik terpisah untuk setiap jenis properti.	Januari 7, 2020
Menggunakan filter node OPC-UA	Menambahkan informasi tentang cara menggunakan filter node OPC-UA untuk meningkatkan kinerja gateway SiteWise Edge saat menambahkan sumber gateway SiteWise Edge.	3 Januari 2020
Memutakhirkan konektor	Menambahkan informasi tentang cara memutakhirkan gateway SiteWise Edge saat versi konektor baru dirilis.	Desember 30, 2019
SiteWise Konektor IoT versi 3 dirilis	Versi 3 dari SiteWise konektor IoT tersedia. Rilis ini menghapus persyaratan izin iot: *.	17 Desember 2019

[SiteWise Konektor IoT versi 2 dirilis](#)

Versi 2 dari SiteWise konektor IoT tersedia. Rilis ini menambahkan dukungan untuk beberapa sumber daya rahasia OPC-UA.

10 Desember 2019

[Membuat dasbor \(\)AWS CLI](#)

Menambahkan informasi tentang cara membuat dasbor dalam AWS IoT SiteWise Monitor menggunakan AWS CLI.

Desember 6, 2019

[AWS IoT SiteWise versi 2 dirilis](#)

Pratinjau yang dirilis untuk versi 2 dari AWS IoT SiteWise. Anda sekarang dapat menelan data melalui OPC-UA, MQTT, dan HTTP, memodelkan data Anda dalam hierarki aset, dan memvisualisasikan data Anda dengan Monitor. SiteWise

2 Desember 2019

- Menulis ulang bagian [pemodelan aset](#) untuk perubahan aset, model aset, dan hierarki aset.
- Memperbarui bagian [konsumsi data untuk menyertakan langkah-langkah AWS IoT Greengrass konektor dan bagian konsumsi data non-gateway](#).
- Menambahkan [AWS IoT SiteWise Monitor](#) bagian dan [panduan aplikasi terpisah](#) yang menunjukkan cara menggunakan aplikasi web SiteWise Monitor.
- Ditambahkan [Data kueri dari AWS IoT SiteWise dan Berinteraksi dengan layanan lain AWS](#) bagian.
- Menulis ulang bagian [memulai](#) agar sesuai dengan pengalaman demo yang diperbarui.

[AWS IoT SiteWise versi 1 dirilis](#)

Merilis pratinjau awal untuk versi 1 dari AWS IoT SiteWise.

25 Februari 2019

Daftar istilah AWS

Untuk terminologi AWS terbaru, lihat [Daftar istilah AWS](#) di Referensi Glosarium AWS.

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.