Securing Internet of Things (IoT) with AWS

AWS Whitepaper

Securing Internet of Things (IoT) with AWS: AWS Whitepaper

Table of Contents

Securing Internet of Things (IoT) with AWS

Publication date: **December 20, 2021** (*Document history* (p. 23))

This whitepaper is a detailed look at how customers can use AWS security services to secure their Internet of Things (IoT) workloads in consumer and industrial environments. This paper is intended for senior-level program owners, decision makers, and security practitioners considering secure enterprise adoption of consumer and industrial IoT (IIoT) solutions.

Introduction

IoT technology allows organizations to optimize processes, enhance product offerings, and transform customer experiences in a variety of ways. Although business leaders are excited about the way in which their businesses can benefit from this technology, it is important for them to consider the complexity and security risks associated with deploying IoT solutions. This is due, in part, to a lack of understanding of how to adopt security best practices to the new technologies, as well as a struggle with disparate, incompatible, and sometimes immature security offerings that fail to properly secure deployments, leading to an increased risk for customer or business owner data. This paper provides guidance on how to understand, approach and meet your security, risk and compliance objectives when deploying IoT solutions with AWS.

Organizations are eager to deliver smart services that can drastically improve the quality of life for populations, business operations and intelligence, quality of care from service providers, smart city resilience, environmental sustainability, and a host of scenarios yet to be imagined. Most recently, AWS has seen an increase in IoT adoption from manufacturing, the healthcare sector and municipalities, with other industries expected to follow in the near term. Many municipalities are early adopters and are taking the lead when it comes to integrating modern technologies, such as IoT. For example:

- **Kansas City, Missouri** – Kansas City created a unified smart city platform to manage new systems operating along its KC streetcar corridor. Video sensors, pavement sensors, connected street lights, a public Wi-Fi network, and parking and traffic management have supported a 40% reduction in energy costs, $1.7 billion in new downtown development, and 3,247 new residential units.
- **City of Chicago, Illinois** – Chicago is installing sensors and cameras in intersections to detect pollen count and air quality for its citizens.
- **City of Catania, Italy** – Catania developed an application to let commuters know where the closest open parking spot is on the way to their destination.
- **City of Recife, Brazil** – Recife uses tracking devices placed on each waste collection truck and cleaning trolley. The city was able to reduce cleaning costs by $250,000 per month, while improving service reliability and operational efficiency.
- **City of Newport, Wales, UK** – Newport deployed smart city IoT solutions to improve air quality, flood control, and waste management in just a few months.
- **Jakarta, Indonesia** – Being a city of 28 million residents that often deals with flooding, Jakarta is harnessing IoT to detect water levels in canals and lowlands, and is using social media to track citizen sentiment. Jakarta is also able to provide early warning and evacuation to targeted neighborhoods so that the government and first responders know which areas are most in need and can coordinate the evacuation process.

At AWS, security is our highest priority, and this mandate includes supporting AWS IoT services and customers. AWS invests significant resources into ensuring that security is incorporated into every layer of its services, extending that security out to devices with IoT. Helping to protect the confidentiality, integrity, and availability of customer systems and data, while providing a safe, scalable, and secure platform for IoT solutions is a priority for AWS. AWS also provides design principles for deploying IoT securely on AWS. Found in the Security pillar of the AWS IoT Lens for the Well-Architected Framework, the design principles are:

- **Manage device security lifecycle holistically** – Data security starts at the design phase, and ends with the retirement and destruction of the hardware and data. It is important to take a complete approach to the security lifecycle of your IoT solution to maintain your competitive advantage and retain customer trust.
- **Ensure least privilege permissions** – Devices should all have fine-grained access permissions that limit which topics a device can use for communication. By restricting access, one compromised device will have fewer opportunities to impact any other devices.
- **Secure device credentials at rest** – Devices should securely store credential information at rest using mechanisms such as a dedicated crypto element or secure flash.
- **Implement device identity lifecycle management** – Devices maintain a device identity from creation through end of life. A well-designed identity system will keep track of a device's identity, track the validity of the identity, and proactively extend or revoke IoT permissions over time.
- **Take a holistic view of data security** – IoT deployments involving a large number of remotely deployed devices present a significant attack surface for data theft and privacy loss. Use a model such as the Open Trusted Technology Provider Standard to systemically review your supply chain and solution design for risk and then apply appropriate mitigations.

Although the IoT Lens provides a checklist and some examples for these design principles, it does not offer prescriptive guidance for securing industrial and consumer IoT applications, which this whitepaper will do.

Security challenges and focus areas

Security risks and vulnerabilities have the potential to compromise the security and privacy of customer data in an IoT application. Coupled with the growing number of connected devices, and the data generated, the potential for security events raises questions about how to address security risks posed by IoT devices and device communication to and from the cloud. Common customer concerns regarding risks focus on the security and encryption of data while in transit to and from the cloud, or in transit from edge services to and from the device, along with patching of devices, device and user authentication, and access control. Another class of security risks stem from protecting physical devices. Hardware-based security, such as using Trusted Platform Modules (TPMs), can protect the unique identities and sensitive data on a device and protect it from manipulative events such as probing of open interfaces on the device.

Addressing these risks by securing IoT devices is essential, not only to maintain data integrity, but to also protect against security events that can impact the reliability of devices. As devices can send large amounts of sensitive data over the internet, and end users are empowered to directly control a device, the security of “things” must permeate every layer of the solution. This whitepaper walks through the ability to integrate security into each of these layers using cloud-native tools and services.

The foundation of an IoT solution must involve security throughout the process or else risking costly recalls or expensive retrofitting when poor security implementations lead to customer issues or downtimes. Getting the right foundations in place makes it easier to adjust to changing conditions and makes it possible to layer on services capable of continuously auditing IoT configurations to ensure that they do not deviate from security best practices and respond if they do. After a deviation is detected, alerts should be raised so appropriate corrective action can be implemented—ideally, automatically.

To keep up with the entry of connected devices into the marketplace, as well as the threats coming from online, it is best to implement services that address each part of the IoT ecosystem and overlap in their capability to secure and protect, audit and remediate, and manage fleet deployments of IoT devices (with or without connection to the cloud). In addition, with the accelerated adoption of Industrial IoT (IIoT) connecting operational technologies (OT) such as industrial control systems (ICS) to the internet, new security challenges have arisen. OT environments are leveraging more IT solutions to improve productivity and efficiency of production operations. This convergence of IT and OT systems creates risk management difficulties that need to be controlled. Operational technology controls physical assets and equipment such that if there is unintended access, it could impact outages of critical services. To address these emerging concerns, customers must evaluate the unique considerations these bring, and apply the appropriate security considerations. In later sections, this whitepaper provides prescriptive guidance on addressing the security concerns related to various IoT use cases including consumer, enterprise, and industrial.

AWS IoT services and compliance

AWS serves a variety of customers, including those in regulated industries. Providing highly secure and resilient infrastructure and services to our customers is a top priority for AWS. Customers can use the tools, services and guidance which AWS offers to manage their risk appropriately and understand how to achieve compliance in the AWS Cloud. Through our shared responsibility model, we help customers to manage risk effectively and efficiently in the IT environment, and provide assurance of effective risk management through our compliance with established, widely recognized, frameworks, and programs. AWS has integrated a risk and compliance program throughout the organization, including AWS IoT services. This program aims to manage risk in all phases of service design and deployment and continually improve and reassess the organization's risk-related activities. AWS regularly undergoes independent third-party attestation audits to provide assurance that control activities are operating as intended. More specifically, AWS is audited against a variety of global and regional security frameworks dependent on region and industry. AWS participates in over 50 different audit programs such as International Standards Organization 27001 (ISO), Payment Card Industry Data Security Standard (PCI), and the Service Organization Control (SOC) reports, among other international, national, and sectoral accreditations.

AWS is sensitive to the fact that customers might have specific compliance requirements that must be demonstrated and complied with. Keeping this in mind, AWS continually adds services that align with compliance programs based on customer demand. For more information, refer to AWS Services in Scope by Compliance Program and AWS Artifact for on-demand access to AWS' compliance reports.

Using provable security to enhance IoT – An industry differentiator

New security services and technologies are being built at AWS to help enterprises secure their IoT and edge devices. In particular, AWS has recently launched checks within AWS IoT Device Defender, powered by an AI technology known as automated reasoning, which uses mathematical proofs for formal verification to determine if there is unintended access to the devices. The AWS IoT Device Defender is an example of how customers can directly use automated reasoning to audit and monitor their own devices. Internally, AWS has used automated reasoning to verify the memory integrity of code running on FreeRTOS and to protect against malware. Investment in automated reasoning to provide scalable assurance of secure software, referred to as *provable security*, allows customers to operate sensitive workloads on AWS.

AWS Zelkova uses automated reasoning to prove that customer data access controls are operating as intended. The access control checks in AWS IoT Device Defender are powered by Zelkova, allowing customers to ensure their data is appropriately protected. An AWS IoT policy is overly permissive if it grants access to resources outside of a customer's intended security configuration. The Zelkova-powered controls integrated into AWS IoT Device Defender verify that policies don't allow actions restricted by the customer's security configuration and that intended resources have permissions to perform certain actions.

Other automated reasoning tools have been used to help secure the AWS IoT infrastructure. The open source formal verification tool CBMC has been used to strengthen the foundations of the AWS IoT infrastructure by proving the memory safety of critical components of the FreeRTOS operating system. A proof of memory safety minimizes the potential of certain security issues, allowing customers and developers to focus on securing other areas in their environment. The memory safety proofs are automatically checked every time a code change is made to FreeRTOS, providing both customers and AWS developers ongoing confidence in the security of these critical components.

Automated reasoning continues to be implemented across a variety of AWS services and features, providing heightened levels of security assurance for critical components of the AWS Cloud. AWS continues to deploy automated reasoning to develop tools for customers as well as internal infrastructure verification technology for the AWS IoT stack.

Implementing IoT security using AWS services

As noted in the previous sections, IoT implementations can have some very unique challenges not present in traditional IT deployments. For example, deploying a consumer IoT device, such as what iRobot has done using AWS to handle scale and spikes, can introduce a new classification of threats to be addressed. Industrial deployments of IoT (IIoT) devices (such as how SKF and Volkswagen have used AWS IoT to optimize its production processes, reduce costs, and provide a better experience to its customers) offer another unique set of security considerations. Lastly, operational technology (OT) or SCADA-based IoT deployments, such as Enel using AWS IoT to get electricity to their customers can require more thought around reliability and anomaly detection. And this is not an exhaustive list. For these use cases there are some common security best practices that can be addressed using AWS services. How enterprises choose to invest in each of these will be based on their risk model.

The following are 10 best practices to build a secure IoT deployment.

Best practices

1. Conduct a formal security risk assessment using a common framework

Conduct a formal security risk assessment using a common framework (such as MITRE ATT&CK). Use this to inform system design.

Whether you're deploying consumer devices, industrial workloads, or operational technologies, it is important to first evaluate the risks and threats associated with your deployment. For example, one common threat to IoT devices listed in the MITRE ATT&CK framework is a Network Denial of Service (T1498). A denial-of-service (DoS) attack against an IoT device can be defined as disallowing status or command and control communication to and from an IoT device and its controllers. In the case of a consumer IoT device, such as a smart bulb, not having the ability to communicate status or receive updates from a central control place could create problems, but would likely not necessarily have dramatic consequences. However, in an OT system managing a water treatment facility, losing the ability to receive commands to open or shut key valves could create a larger impact to people and the environment. So, it's important to look at the impact of various common threats, how they apply to different IoT use cases, and ways to mitigate them. Key steps include:

- Identify, manage, and track gaps and vulnerabilities. Create and maintain an up-to-date threat model that can be monitored against.
- Segment systems based on their risk assessment. Some IoT and IT systems may share the same risks, so use a predefined zoning model with appropriate controls between them.
- Follow a micro segmentation approach to isolate the impact of an event.
- Use appropriate security mechanisms to control information flow between network segments.
- Regularly identify and review security event minimization opportunities as your IoT system evolves.

Supporting AWS resources

When building your environment inside of AWS, foundational services such as Amazon Virtual Private Cloud (VPC), VPC security groups (SGs), and network access control lists (network ACLs) should be used to implement the micro segmentation. AWS recommends using multiple accounts, which helps to isolate IoT applications, data, and business processes across your environment and use AWS Organizations for better manageability and centralized insight. Additional information can be found in the Security Pillar of AWS Well-Architected Framework and Organizing Your AWS Environment Using Multiple Accounts whitepaper.

2. Maintain an asset inventory of all IoT assets

Maintain an asset inventory of all IoT assets, including IT assets required to maintain IoT operations. Categorize them by safety, criticality, ability to patch, and other actionable criteria.

A critical aspect of a good security program is having visibility into your system. It's also important that you create visibility with actionable outcomes in mind, so you can automate operations and maintenance of these devices after deployment.

- Create and maintain an asset inventory for all IoT assets along with their major characteristics that you may want to action upon. This includes things such as deployed certificates and software or hardware versions.
- Segment devices into categories or apply appropriate tags to be able to manage them programmatically. Focus on actionable data such criticality of the devices, location, whether the device can or should be updated, or important contact and owner information.

Supporting AWS resources

AWS provides the following services to help you create and maintain a connected asset inventory:

- AWS IoT Device Management – For devices connected to AWS IoT.
- AWS Systems Manager – For cloud and on-premises computers.
- Security Pillar of AWS Well-Architected and IoT Lens

3. Provision IoT devices and systems with unique identities and credentials

Provision IoT devices and systems with unique identities and credentials. Apply authentication and access control mechanisms at each system interface.

Strong identity controls are key to operational excellence. However, IoT implementation considerations around physical control of devices range widely. Therefore, not only is it important to ensure devices receive unique identities and credentials, but also that those credentials are appropriately protected on the device, and monitoring and automated remediation plans are put in place when there's deviation from expected standards.

- Assign unique identities to IoT devices such as X.509 certificates to each device. Monitor that the identity does not change on devices or that certificates are not reused.
- Create mechanisms to facilitate the generation, distribution, rotation, and revocation of credentials.
- When appropriate, use hardware-protected modules such as TPMs for storing credentials and performing authentication operations.
- Avoid hardcoding credentials or storing secrets that are not unique to the device on IoT devices.

Supporting AWS resources

AWS provides the following assets, services, and capabilities to help you identify, sort, and secure your IoT assets:

- Security and identity for AWS IoT
- Device manufacturing and provisioning with X.509 certificates in AWS IoT Core – Goes over various mechanisms to securely provision identities to your IoT devices.
- AWS Certificate Manager Private Certificate Authority – For provisioning your own certificates.
- Amazon Cognito – A service that provides authentication, authorization, and user management for your web and mobile apps.
- AWS Identity and Access Management (IAM) – A service that enables you to manage access to AWS services and resources securely.
- Device authentication and authorization for AWS IoT Greengrass
- AWS Secrets Manager – A service that can be used to securely store and manage secrets in the cloud and encrypts the secrets using AWS Key Management Service (AWS KMS).
- AWS KMS – Allows you to easily create and control the keys used for cryptographic operations in the cloud.
- Security Pillar of AWS Well-Architected and IoT Lens

4. Define appropriate update mechanisms for software and firmware updates.

Whether it's deploying patches to individual packages, updating local firmware, or wholesale replacing the software on an IoT device, patching is critical during the IoT device's lifecycle. Although different use cases will have different tradeoffs, common things to consider include rolling out patches gradually to catch defects and ensuring all devices of the same type aren't brought down simultaneously, being responsive to vulnerabilities, and ensuring the patch delivery mechanism can't be used by unauthorized actors. Some additional considerations include:

- Begin with having a mechanism to push software and firmware to devices in the field to patch security vulnerabilities and improve device functionality.
- Apply and verify digital signatures on distributed deployment artifacts.
- Verify the integrity of the software on the device before starting to run it ensuring that it comes from a reliable source (signed by the vendor) and that it is obtained in a secure manner.

- Monitor status of deployments throughout your ecosystem and investigate any failed or stalled deployments.
- Use rolling patches using asset tags or other segmentation mechanism based on the impact of a latent issue.
- Include patch status in your inventory of the deployed devices.
- Use version control mechanisms to prevent unauthorized actors from forcing firmware or software downgrades.
- Maintain notification mechanisms to immediately alert the appropriate stakeholders when security updates are required or fail.
- Create mechanisms to identify, isolate into a different network segment, or replace IoT devices that are outside of compliance.
- Create detection and response mechanisms to handle unauthorized changes in deployed software or firmware.

Supporting AWS resources

AWS provides the following capabilities and services to help you organize and maintain a continuous development and deployment pipeline:

- FreeRTOS over-the-air updates
- OTA updates of AWS IoT Greengrass Core software
- AWS IoT Jobs – Defines a set of remote operations that you send to and run on one or more devices connected to AWS IoT.
- AWS Systems Manager Patch Manager – Automates the process of patching managed instances with both security related and other types of updates, such as operating systems and applications.
- Security Pillar of AWS Well-Architected and IoT Lens

5. Encrypt persistent data at rest

For devices such as sensors or cameras, information stored on deployed devices may seem innocuous, but when physical control of a device is not guaranteed that information can be a target for unauthorized actors. Whether in the consumer space like cached videos on cameras, industrial application with proprietary machine learning (ML) models, or even some configuration data for operational environments, the best course of action is to encrypt all data (even transitive data) stored at rest when possible. Some additional considerations include:

- Identify and classify data collected throughout your IoT ecosystem and learn their corresponding business use case.
- Categorize data based on the earlier risk analysis, including impact to other stakeholders.
- Identify opportunities to stop collecting unused data or reducing granularity and retention time, then implement improvements.
- Ensure integrity of data used to operate devices through cryptographic mechanisms.
- Apply access controls using least privilege principle to encryption keys, and monitor and audit data access.
- When necessary, follow least privilege and need-to-know principles when granting access to third parties.
- Consider privacy and transparency expectations of your customers and corresponding legal requirements.

Supporting AWS resources

AWS provides the following assets and services to help you secure IoT data at the edge and cloud:

- AWS Shared Responsibility Model – For security and compliance.
- AWS Data Privacy
- AWS Privacy Notice
- AWS Compliance programs and offerings
- AWS Compliance Solutions Guide
- AWS Key Management Service (AWS KMS) – Can be used to create and control the keys used for cryptographic operations in the cloud.
- Security Pillar of AWS Well-Architected and IoT Lens

6. Encrypt all data in transit

Encrypt all data in transit, including sensor and device data, administration, provisioning, and deployments.

Nearly all modern IoT devices have the power to perform encryption of network traffic, so take advantage of that and protect both the data plane and control plane communications. This not only ensures confidentiality of the data, but also the integrity of monitoring signals. For protocols that can't be encrypted, consider if a second device closer to the IoT asset can accept the communication and convert it to something more secure to then send outside the local perimeter. Some additional considerations include:

- Protect the confidentiality and integrity of inbound and outbound network communication channels that you use for data transfers, monitoring, administration, provisioning, and deployments by selecting modern internet native cryptographic network protocols.
- If possible, limit the number of protocols implemented within a given environment and disable default network services that are unused.
- If over-the-air updates are implemented, network-related vulnerabilities that affect the integrity of the over-the-air process should be addressed first.
- If possible, implement mechanisms to identify when an insecure network environment is being used. For example, if the certificate used for TLS encryption doesn't match a known certificate on the device such as in a man-in-the-middle event.

Supporting AWS resources

AWS provides the following assets, capabilities, and services to help you encrypt your networks:

- AWS IoT SDKs – Help you securely and quickly connect your devices to AWS IoT.
- FreeRTOS libraries – Provide additional functionality to the FreeRTOS kernel and its internal libraries.
- AWS Certificate Manager Private Certificate Authority – Provision your own certificates.
- Security best practices for AWS IoT SiteWise
- Security Pillar of AWS Well-Architected and IoT Lens

7. Secure both the IoT environment and supporting IT environments to the same level of criticality

Secure both the IoT environment and supporting IT environments to the same level of criticality following a well-documented standard. This is especially true for gateways that serve as boundaries between systems.

Often, IoT systems still have a dependency on traditional IT systems to operate. Whether that's for identity and authorization, billing, monitoring and remediation, or maintenance, having these systems become unavailable to the IoT system can cause cascading failures. Therefore, you should use the risk assessment and asset inventory to document these critical dependencies and architect all relevant systems to the same level of resiliency and security. Some ways to do this include:

- Plan and manage security lifecycle of devices.
- Consistently harden internet-connected network resources such as edge gateways.
- Avoid hardcoding or storing credentials and secrets locally on devices.
- Use device certificates and temporary credentials instead of long-term credentials to access AWS cloud services.
- Limit the number of listening ports on IoT devices, and ensure access only from authorized systems.
- Create allow lists for access with a management mechanism similar to that of software updates.
- Disable unused sensors, actuators, services, or software on the IoT device.
- Establish secure connections to cloud services, and monitor these connections.

Supporting AWS resources

AWS provides the following assets, capabilities, and services to help secure cloud connected network resources and securely manage on-premises computing resources:

- NIST Guide to General Server Security – For general guidance on security devices (such as edge gateways).
- AWS IoT Greengrass hardware security
- Working with secrets at the Edge.
- AWS IoT SiteWise Gateway – Securely configuring edge gateways.
- AWS Systems Manager – Provides you with a centralized and consistent way to gather operational insights and carry out routine management tasks.
- AWS IoT Device Management – A service that allows you to securely register, organize, monitor, and remotely manage IIoT devices at scale throughout their lifecycle.
- AWS IoT secure tunneling – Accesses IIoT devices behind restricted firewalls at remote sites for troubleshooting, configuration updates, and other operational tasks.
- Plant network to Amazon Virtual Private Cloud connectivity options
- AWS IoT Greengrass - Connect on port 443 or through a network proxy
- Security Pillar of AWS Well-Architected and IoT Lens

8. Deploy security auditing and monitoring mechanisms across your IoT environment and relevant IT systems.

As we've discussed, it's important to ensure the proper configuration of IoT devices when they are put into production and that they are updated. But, it's also important to monitor their behavior and security posture on an ongoing basis.

- Deploy auditing and monitoring mechanisms to continuously collect and report activity metrics and logs.
- Monitor on-device and related off-device activities such as network traffic and entry points, process implementation, and system interactions for any unexpected behavior.
- Continuously check that your security controls and systems are intact by explicitly testing them.
- Implement a monitoring solution to create a traffic baseline, and monitor anomalies and adherence to the baseline.
- Collect security logs and analyze them in real time using automated tooling.
- Monitor availability of your IoT devices in real time, where technically feasible.

Supporting AWS resources

AWS provides the following capabilities and services to help you monitor your security at varying levels:

- AWS IoT Device Defender – Monitors and audits your fleet of IoT devices.
- Monitor AWS IoT with CloudWatch Logs – Centralizes the logs from all of your systems, applications, and AWS services that you use, in a single, highly scalable service.
- Log AWS IoT API Calls with AWS CloudTrail – Provides a record of actions taken by a user, a role, or an AWS service in AWS IoT.
- Monitoring with AWS IoT Greengrass Logs
- AWS Config – Assess, audit, and evaluate the configurations of your AWS resources.
- Amazon GuardDuty – Continuously monitors for malicious activity and unauthorized behavior to protect your AWS accounts and workloads.
- AWS Security Hub – Automates AWS security checks and centralizes security alerts.
- Security Pillar of AWS Well-Architected and IoT Lens

9. Create incident response playbooks, and build automation as your security response matures

Management systems must build continuous health checks before the devices get shipped. It's also important to create incident response playbooks for when those checks identify anomalies, and, as processes mature, automate the containing of events and returning to a known good state. Although it may seem daunting, this doesn't have to happen all at the same time. This is a process that will continue throughout the lifecycle of the IoT environment, with the complexity and maturity of the program growing over time.

- Maintain and regularly exercise a security incident response plan to test monitoring functionality.

- Collect security logs and analyze them in real time using automated tooling. Build playbooks in response to unexpected findings.
- Create an incident response playbook with clearly understood roles and responsibilities.
- Test incident response procedures on a periodic basis.
- As procedures become more stable, automate their implementation but maintain human interaction. As the automated procedures are validated, automate what triggers their implementation.

Supporting AWS resources

AWS provides the following assets and services to help you monitor your security and create incident response playbooks:

- AWS Security Incident Response Guide
- AWS Systems Manager – Provides a centralized and consistent way to gather operational insights and carry out routine management tasks.
- Security Pillar of AWS Well-Architected and **IoT Lens**

10. Create and test business continuity and recovery plans

During an event, different IoT systems could behave in different ways. Before those events occur, you must define parameters relevant to your use case (should a system fail open or fail shut, does the system attempt recovery automatically or require human intervention, do you need to enable or disable manual controls) and then test those rigorously. Again, use the risk assessment and criticality assignments performed early in this process to ensure you apply the right amount of scrutiny and resources to this phase. Don't forget about defining when to return to the baseline state in your recovery plans.

- Define important parameters (such as overall availability) for your stakeholders.
- Define the resilience requirements for the system and analyze failure modes to ensure adherence.
- Test recovery plans periodically and adapt them according to lessons learned from tests and actual security incidents.
- Perform threat and risk assessment of supporting IT systems and develop written procedures on how to return to the normal, well-defined, state of operation tailored to the assessment's results.
- Include third-party aspects (such as network communications, software, and support).
- Use resiliency features at the edge to support data resiliency and backup needs.
- Use cloud services for backup and business continuity.

Supporting AWS resources

AWS provides the following assets and services to help you create and test business continuity and recovery plans:

- AWS IoT Lens for AWS Well-Architected Framework – A document that covers commonly encountered IoT use cases and identified key solution elements to ensure that your workload architecture uses established best practices.
- Resilience in AWS IoT Greengrass
- Backup and Restore Use Cases with AWS
- CloudEndure Disaster Recovery

- AWS Backup

These general best practices apply across all IoT deployments, but as mentioned previously, different industries will have different threat and risk models. In the next section we will dive into examples across these industries and demonstrate prescriptive approaches that are more targeted.

Augmenting security practices for industrial control systems, operational technology, and industrial IoT

Industrial IoT is driving changes to the operational technology (OT) landscape, making it more connected. OT such as industrial control systems (ICS) and supervisory control and data acquisition systems (SCADA) is the use of hardware and software to monitor and control physical assets and production operation. Industrial internet of things (IIoT) is the connection of ICS with enterprise systems, business processes, and analytics, and is a key enabler for smart manufacturing and Industry 4.0. The convergence of IT and OT systems is creating a mix of technologies that were designed for remote network environments and ones that were not, which creates risk management difficulties that need to be addressed. This OT and IT convergence introduces new security risks and challenges in the industrial environment which need to be properly managed.

Although general best practices still apply, there are some additional considerations that should be put in place to support the often times higher criticality and larger impact of OT and IIoT systems. To help companies plan their industrial digital transformation safely and securely, AWS recommends augmenting general best practices with these fundamentals in ICS and OT, and IIoT security.

1. **Conduct a formal security risk assessment using a common framework (such as MITRE ATT&CK for ICS). Use this to inform system design.**
   - Segment industrial plants networks based on a predefined zoning model that includes establishment of demilitarized zones and control of traffic between zones (for example, according to the Purdue Model).
   - Use application-specific firewalls, unidirectional gateways, and data diodes to control information flow between network segments.
   - Use protocol converters to convert insecure industrial protocols to secure protocols as close to the device as possible.
   - If possible, isolate safety networks from business and control networks.
   - If you are unable to protect insecure industrial assets, isolate or disconnect them from the network.
2. **Maintain an asset inventory of all IIoT assets, including IT assets required to maintain IIoT operations. Categorize them by safety, criticality, ability to patch, and other actionable criteria.**
   - Maintain an updated inventory of devices that don't support modern security controls. Isolate them from the rest of the other OT and IIoT devices by network segmentation. Create a plan to replace them with devices that do support modern security controls.
   - Conduct security architecture reviews as assets move or become dependent on new systems.
   - Consider if integrating the IIoT asset information into your enterprise asset management system provides any benefit. Assess the business risk of having a segmented inventory system.
   - Create and maintain an up-to-date OT and IIoT network architecture showing how these assets are interconnected along their relationships (asset hierarchies).
3. **Provision modern IIoT devices and systems with unique identities and credentials. Apply authentication and access control mechanisms.**

- Assign unique identities to modern IIoT devices so that when a device connects to other devices or cloud services, it must establish trust by authenticating using principals such as X.509 certificates, security tokens, or other credentials.
- Create mechanisms to facilitate the generation, distribution, rotation, and revocation of credentials.
- Establish root of trust by using hardware-protected modules such as TPMs if available on the device.
- Ensure least privilege access controls for IIoT devices, edge gateways, and agent software accessing local and cloud resources.
- Avoid hardcoding or storing credentials and secrets locally on OT and IIoT devices.

4. **Define appropriate update mechanisms for software and firmware updates.**
   - Maintain an inventory of the deployed software across your OT and IIoT ecosystem, including versions and patch status.
   - Create mechanisms to identify, network isolate, and replace legacy devices and IIoT systems that are not capable of receiving updates.
   - Perform deployment of patches for the OT and IIoT devices only after testing the patches in a test environment before implementing them in production.
   - Create a plan to validate firmware, patches, or any other software, from software providers in the supply chain to ensure their authenticity and validity.
   - For OT and IIoT systems that cannot be updated, apply compensating measures such as network isolation and continuous monitoring.
5. **Encrypt persistent data at rest.**
   - Monitor the production data at rest and in transit to identify potential unauthorized data modification.
   - When appropriate, based on risk, access controls should also be applied at the connectivity layer using security appliances such as unidirectional network devices or data diodes.
   - Identify and consider the unique capabilities of your OT and IIoT devices. This could include mobility, actuation, sensory data collection and transmission, and ownership transfers that impact your regulatory and legal compliance.
   - Create mechanisms for secure IIoT data sharing, governance, and sovereignty.
6. **Encrypt all data in transit, including sensor and device data, administration, and provisioning and deployments.**
   - Ensure security capabilities and interoperability between industrial protocols when implementing different protocols for various devices within the same system.
   - Select the newer version of industrial protocols which offer security features, and configure the highest level of encryption available when using ICS protocols such as CIP Security, Modbus Secure, OPC UA, and so on.
   - When secure industrial protocols are not an option and you use legacy insecure industrial protocols, then tighten the trust boundary using a protocol converter to translate the insecure protocol to a secure protocol as close to the data source as possible. Otherwise, segregate the plant network into smaller cell or area zones by grouping ICS devices into functional areas to limit the scope and area of insecure communications. Use specialized firewall and inspection products that understand ICS protocols to inspect traffic entering and leaving cell or area zones and can detect anomalous behavior in the control network.
   - Have a mechanism to identify and disable vulnerable wireless networks in the local environment which get installed during proof of concepts, often without the necessary security approvals.
7. **Secure both the IoT environment and supporting IT environments to the same level of criticality following a well-documented standard. This is especially true for gateways that serve as boundaries between systems.**
   - Configure, monitor, and securely manage IIoT devices, edge gateways, and virtual machines.
   - Use secure enclosures to protect OT and IIoT assets.
   - Establish a mechanism for bidirectional, secure communication to remote devices, which are often behind firewalls.

- Provision your IIoT devices and field gateways with credentials that grant only the required privileges.
- Regularly review and identify attack surface minimization opportunities as your IIoT ecosystem evolves.

8. **Deploy security auditing and monitoring mechanisms across your IIoT environment and relevant IT systems.**
   - Verify that security controls prevent unauthorized access and maintain their integrity in the event of external dependency or internal system failures.
   - Implement a monitoring solution in the OT and IIoT environments to create an industrial network traffic baseline and monitor anomalies and adherence to the baseline.
   - Perform periodic reviews of network logs, access control privileges, and asset configurations.
9. **Create incident response playbooks, and build automation as your security response matures.**
   - Maintain and regularly exercise a security incident response plan along with containment and recovery mechanisms. This should be in correspondence to the technical skill level of operators of your OT and IIoT elements and their deployment and ownership model.
   - Ensure that your security operations center is trained and knowledgeable on OT and IIoT security logs, and alerts from the automated tooling.
10. **Create and test business continuity and recovery plans.**
    - Focus on ensuring resilience of Industry 4.0 systems by creating a business continuity plan and disaster recovery plan. Test the plans periodically and adapt them according to lessons learned from tests and actual security incidents.
    - Perform threat and risk assessment of OT and IIoT, and supporting IT systems, and develop written procedures on how to return to the normal, well-defined, state of operation tailored to the assessment's results.
    - In business continuity and recovery plans, include third-party aspects.
    - Conduct ongoing security testing across OT and IIoT periodically to test devices and OT systems, edge gateways, networks, and communication and cloud services.

Government contributions to IoT security

Although private sector organizations are actively deploying IoT in use cases such as healthcare, industrial construction, and low-power consumer goods, governments at the national and local levels are beginning to address IoT adoption and security. Some key players and their roles include:

- **National Institute of Standards and Technology** – Spearheading multiple whitepapers and industry efforts to define and reduce risk of IoT environments.
- **US Department of Defense** – Providing policy recommendations for agencies addressing IoT risks.
- **Federal Trade Commission** – Pursuing action against device manufacturers who fail to meet the reasonable data security bar.

In the United States, some states such as California are enacting their own rules, and globally, other countries such as the United Kingdom are advancing regulation as well. For more details on these developments, refer to Appendix 2 – Government involvement in IoT.

Key IoT security takeaways

Despite the number of best practices available, there is no one-size-fits-all approach to mitigating the risks to IoT solutions. Depending on the device, system, service, and environment in which the devices are deployed, different threats, vulnerabilities, and risk tolerances exist for customers to consider. Here are key takeaways to help incorporate complete security across data, devices, and cloud services:

1. **Incorporate security in the design phase.**

   The foundation of an IoT solution starts and ends with security. Because devices may send large amounts of sensitive data, and end users of IoT applications may also have the ability to directly control a device, the security of *things* must be a pervasive design requirement. Security is not a static formula; IoT applications must be able to continuously model, monitor, and iterate on security best practices.

   A challenge for IoT security is the lifecycle of a physical device and the constrained hardware for sensors, microcontrollers, actuators, and embedded libraries. These constrained factors may limit the security capabilities each device can perform. With these additional dynamics, IoT solutions must continuously adapt their architecture, firmware, and software to stay ahead of the changing security landscape. Although the constrained factors of devices can present increased risks, hurdles, and potential tradeoffs between security and cost, building a secure IoT solution must be the primary objective for any organization.
2. **Build on recognized IT security and cybersecurity frameworks.**

   AWS supports an open, standards-based approach to promote secure IoT adoption. When considering the billions of devices and connection points necessary to support a robust IoT ecosystem for consumer, industrial, and public sector use, interoperability is vital. Thus, AWS IoT services adhere to industry standard protocols and best practices. Additionally, AWS IoT Core supports other industry-standard and custom protocols, allowing devices to communicate with each other even if they are using different protocols. AWS is a strong proponent of interoperability so that developers can build on top of existing platforms to support evolving customer needs. AWS also supports a thriving partner ecosystem to expand the menu of choices and stretch the limits of what is possible for customers. Applying globally recognized best practices carries a number of benefits across all IoT stakeholders including:
   - Repeatability and reuse, instead of re-starting and re-doing
   - Consistency and consensus to promote the compatibility of technology and interoperability across geographical boundaries
   - Maximizing efficiencies to accelerate IT modernization and transformation
3. **Focus on impact to prioritize security measures.**

   Attacks or abnormalities are not identical and may not have the same impact on people, business operations, and data. Understanding customer IoT ecosystems and where devices will operate within this ecosystem informs decisions on where the greatest security risks are—within the device as part of the network or physical component. Focusing on the risk impact assessment and consequences is critical for determining where security efforts should be directed along with who is responsible for those efforts in the IoT ecosystem.
4. **Start with using zero-trust security principles.**

   Zero-trust principles are intended for an organization's infrastructure, which includes operational technology (OT), IT systems, IoT, and Industrial Internet of Things (IIoT). Traditional security models rely heavily on network segmentation and give high levels of trust to devices based on their network presence. In comparison, zero trust requires your users, devices, and systems to prove their trustworthiness, and it enforces fine-grained, identity-based rules that govern access to applications,

data, and other assets. AWS provides guidance on how to implement zero trust IoT solutions with AWS IoT.

Conclusion

Along with an exponential growth in connected devices, each *thing* in IoT communicates packets of data that require reliable connectivity, storage, and security. With IoT, an organization is challenged with managing, monitoring, and securing immense volumes of data and connections from dispersed devices. But this challenge doesn't have to be a roadblock in a cloud-based environment. In addition to scaling and growing a solution in one location, cloud computing enables IoT solutions to scale globally and across different physical locations while lowering communication latency and allowing for better responsiveness from devices in the field. AWS offers a suite of IoT services with complete security, including services to operate and secure endpoints, gateways, platforms, and applications as well as the traffic traversing across these layers. This integration simplifies secure use and management of devices and data that continually interact with each other, allowing organizations to benefit from the innovation and efficiencies IoT can offer while maintaining security as a priority. AWS offers customers a defense in depth approach with multiple security services and an easier, faster and more cost-effective path towards comprehensive, continuous and scalable IoT security, compliance and governance solutions.

Contributors

Contributors to this document include:

- Ryan Dsouza, Principal IoT Solutions Architect
- Michael Wasielewski, Principal Security and Compliance Specialist

Document history

To be notified about updates to this whitepaper, subscribe to the RSS feed.

Change	Description	Date
Whitepaper updated (p. 23)	Content updates.	December 20, 2021
Initial publication (p. 23)	Whitepaper first published.	April 1, 2019

Note
To subscribe to RSS updates, you must have an RSS plug-in enabled for the browser that you are using.

Appendix 1 – AWS IoT services and security capabilities

AWS offers a suite of IoT services to help customers secure their devices, connectivity, and data. These services enable customers to use complete security from device protection to data in transit and at rest. They also provide security features that enable the application and implementation of security policies required to meet their security benchmark.

AWS IoT provides broad and deep functionality; customers can build IoT solutions for virtually any use case across a wide range of devices. AWS IoT integrates with artificial intelligence services so customers can make devices smarter—even without internet connectivity. Built on the AWS Cloud, and used by millions of customers in 245 countries as of September 2021, AWS IoT can easily scale as customers' device fleets grow and their business requirements evolve. AWS IoT also offers comprehensive security features so customers can create preventative security policies and respond immediately to potential security issues.

AWS IoT provides cloud services and edge software, enabling customers to securely connect devices, gather data, and take intelligent actions locally, even when internet connectivity is interrupted. Cloud services allow customers to quickly onboard and securely connect large and diverse fleets, maintain fleet health, keep fleets secure, and detect and respond to events across IoT sensors and applications. AWS IoT can also be used to analyze data and build sophisticated ML models. These models can be deployed in the cloud or locally on customer devices to make devices smarter.

Although current AWS IoT services range widely to allow for innovative and comprehensive IoT solutions, this whitepaper focuses on the following six services, which are foundational for IoT security:

- **FreeRTOS** is an open source operating system for microcontrollers that makes small, low-power edge devices easy to program, deploy, secure, connect, and manage.
- **AWS IoT Greengrass** is software that lets customers run local compute, messaging, data caching, sync, and ML inference capabilities on connected devices.
- **AWS IoT Core** is a managed cloud service that lets connected devices easily and securely interact with cloud applications and other devices.
- **AWS IoT Device Management** is a cloud-based device management service that makes it easy to securely onboard, organize, monitor, and remotely manage IoT devices at scale.
- **AWS IoT Device Defender** is an IoT security service that continuously monitors and audits customers' IoT configurations to ensure that they do not deviate from security best practices.
- **AWS IoT SiteWise** is a managed service that enables industrial enterprises to collect, store, organize, and visualize thousands of sensor data streams across multiple industrial facilities.

Service descriptions and security features are further discussed in the following sections.

FreeRTOS – Device software

FreeRTOS is an open source operating system for microcontrollers that makes small, low-power edge devices easy to program, deploy, secure, connect, and manage. FreeRTOS is a popular open source operating system for microcontrollers that has been extended with software libraries that make it easy to securely connect customers' small, low-power devices directly to AWS Cloud services (such as AWS IoT Core) or to more powerful edge devices running AWS IoT Greengrass.

Security capabilities

FreeRTOS comes with libraries to help secure device data and connections, including support for data encryption and key management. FreeRTOS includes support for Transport Layer Security (TLS v1.2) to help devices connect securely to the cloud. FreeRTOS also has a code signing feature to ensure customer device code is not compromised during deployment as well as capabilities for OTA updates to remotely update devices with feature enhancements or security patches.

AWS IoT Greengrass – Software for edge computing

AWS IoT Greengrass is software that lets customers run local compute, messaging, data caching, sync, and ML inference capabilities for connected devices, allowing connected devices to operate even with intermittent connectivity to the cloud. After the device reconnects, AWS IoT Greengrass synchronizes the data on the device with AWS IoT Core, providing constant functionality regardless of connectivity. AWS IoT Greengrass seamlessly extends AWS to devices so they can act locally on the data they generate, while still using the cloud for management, analytics, and durable storage.

Security capabilities

AWS IoT Greengrass authenticates and encrypts device data for both local and cloud communications, and data is never exchanged between devices and the cloud without proven identity. The service uses security and access management similar to what customers are familiar with in AWS IoT Core, with mutual device authentication and authorization, and secure connectivity to the cloud.

More specifically, AWS IoT Greengrass uses X.509 certificates, managed subscriptions, AWS IoT policies, and AWS Identity and Access Management (IAM) policies and roles to ensure that AWS IoT Greengrass applications are secure. AWS IoT devices require an AWS IoT thing, a device certificate, and an AWS IoT policy to connect to the AWS IoT Greengrass service. This allows AWS IoT Greengrass core devices to securely connect to the AWS IoT cloud service. It also allows the AWS IoT Greengrass cloud service to deploy configuration information, AWS Lambda functions, and managed subscriptions to AWS IoT Greengrass core devices. In addition, AWS IoT Greengrass provides hardware root of trust private key storage for edge devices.

Other important security capabilities of AWS IoT Greengrass are monitoring and logging. For example, core software in the service can write logs to Amazon CloudWatch (which also functions for AWS IoT Core) and to the local file system of customers' core devices. Logging is configured at the group level and all AWS IoT Greengrass log entries include a time stamp, log level, and information about the event. AWS IoT Greengrass is integrated with AWS CloudTrail—a service that provides a record of actions taken by a user, role, or an AWS service in AWS IoT Greengrass—and if activated by the customer, it captures application programming interface (API) calls for AWS IoT Greengrass as events. This includes calls from the AWS IoT Greengrass console and code calls to the AWS IoT Greengrass API operations. For example, customers can create a trail and calls can enable continuous delivery of AWS CloudTrail events to an Amazon Simple Storage Service (Amazon S3) bucket, including events for AWS IoT Greengrass. If customers don’t want to create a trail, they can view the most recent events in the AWS CloudTrail console in event history. This information can be used to do a number of things, such as determining when a request was made to AWS IoT Greengrass and the IP address from which the request was made.

Best practice options are available to secure customers’ data on the device and should be utilized whenever possible. For AWS IoT Greengrass, all IoT AWS IoT Greengrass devices should enable full disk encryption and follow key management best practices. Customers can utilize full disk encryption, using AES 256-bit keys based on NIST FIPS 140-2 validated algorithms and follow key management best practices. For low-power devices such as those using FreeRTOS, customers can follow NIST 8114 lightweight cryptography recommendations.

The previous sections covered microcontrollers and edge use cases. The following sections will focus on IoT services that operate in the cloud.

AWS IoT Core – Cloud-based IoT gateway

AWS IoT Core is a managed cloud service that lets connected devices easily and securely interact with cloud applications and other devices. AWS IoT Core provides secure communication and data processing across different kinds of connected devices and locations so customers can build IoT applications. Examples of customer use cases include industrial solutions and connected home solutions, with the ability to support billions of devices and trillions of messages that can be processed and routed to AWS endpoints and other devices reliably and securely.

Security capabilities

AWS IoT Core offers a number of solutions to customers that help enable and maintain security. AWS Cloud security mechanisms protect data as it moves between AWS IoT and other devices or AWS services. Devices can connect using a variety of identity options (X.509 certificates, IAM users and groups, Amazon Cognito identities, or custom authentication tokens) over a secure connection. Although customers perform the client-side validations (such as chain of trust validation, hostname verification, secure storage, and distribution of their private keys), AWS IoT Core provides secure transportation channels using TLS. The AWS IoT rules engine also forwards device data to other devices and AWS services according to customer-defined rules. AWS access management systems are used to securely transfer data to its final destination. Another AWS IoT authorization feature worth noting is AWS IoT policy variables, which helps avoid the provisioning of over-privileged credentials to a device. These features, used in conjunction with general cybersecurity best practices, work to protect customer data.

AWS IoT Device Management – Cloud-based IoT device management service

AWS IoT Device Management helps customers onboard, organize, monitor, and remotely manage IoT devices at scale. AWS IoT Device Management integrates with AWS IoT Core to easily connect devices to the cloud and other devices so customers can remotely manage their fleets of devices. AWS IoT Device Management helps customers onboard new devices by using AWS IoT within the AWS Management Console or an API to upload templates that they populate with information like device manufacturer and serial number, X.509 identity certificates, or security policies. Following this, customers can then configure the entire fleet of devices with this information with a few clicks in AWS IoT within the AWS Management Console.

Security capabilities

With AWS IoT Device Management, customers can group their device fleet into a hierarchical structure based on function, security requirements, or similar categories. They can group a single device in a room, multiple devices on the same floor, or all the devices that operate within a building. These groups can then be used to manage access policies, view operational metrics, or perform actions across the entire group. Additionally, a feature known as dynamic thing groups can automatically add devices that meet the customer-defined criteria and remove devices that no longer match the requirements. This securely streamlines the process while maintaining operational integrity. Dynamic thing groups also makes it easy to find device records based on any combination of device attributes and allows customers to perform bulk updates.

With AWS IoT Device Management, customers can also push software and firmware to devices in the field to patch security vulnerabilities and improve device functionality; implement bulk updates; control

deployment velocity; set failure thresholds; and define continuous jobs to update device software automatically so that they are always running the latest version of software. Customers can remotely send actions (such as device reboots or factory resets) to fix software issues in the device or restore the device to its original settings. Customers can also digitally sign files that are sent to their devices, helping to ensure the devices are not compromised.

The ability to push software updates isn't limited to cloud services. OTA update jobs in FreeRTOS allow customers to use AWS IoT Device Management to schedule software updates. Similarly, customers can also create an AWS IoT Greengrass core update job for one or more AWS IoT Greengrass core devices using AWS IoT Device Management to deploy security updates, bug fixes, and new AWS IoT Greengrass features to connected devices.

With the secure tunneling feature, customers can establish a secure remote communications session to a device. This provides secure connectivity to individual devices, which you can then use to diagnose issues and act to solve in just a few clicks. You can also make multiple, concurrent client connections over a single secure tunnel, enabling you to perform more advanced device troubleshooting, such as issuing remote shell commands to a device while simultaneously debugging a web application on the same device.

AWS IoT Device Defender – Cloud-based IoT device security service

AWS IoT Device Defender is a fully managed service that helps customers secure their fleet of devices. The service continuously audits IoT configurations to ensure that configurations aren't deviating from security best practices—such as ensuring device identity, authenticating and authorizing devices, and encrypting device data. The service can send an alert if there are any gaps in a customer's IoT configuration that might create a security risk, such as identity certificates being shared across multiple devices or a device with a revoked identity certificate trying to connect to AWS IoT Core.

AWS IoT Device Defender also lets customers continuously monitor security metrics from devices and AWS IoT Core for deviations from the expected behaviors for each device. Customers can define the appropriate behavior for their devices or use ML to model the regular device behavior based on historical data. If something doesn't look right according to defined behaviors or ML models, AWS IoT Device Defender pushes an alarm so customers can act to mitigate the issue. For example, spikes in outbound traffic might indicate that a device is participating in a distributed denial of service (DDoS) attack. Additionally, AWS IoT Greengrass and FreeRTOS automatically integrate with AWS IoT Device Defender to provide security metrics from the devices for evaluation.

Security capabilities

AWS IoT Device Defender audits IoT configurations associated with customers' devices against a set of defined IoT security best practices so customers know exactly where they have security gaps. Customers can run audits on a continuous or one-time basis. AWS IoT Device Defender comes with security best practices that customers can select and run as part of the audit. For example, customer can create an audit to check for identity certificates that are inactive, revoked, expiring, or pending transfer in less than seven days. Audits make it possible for customer to receive alerts while their IoT configuration is updated.

AWS IoT Device Defender detects anomalies in device behavior that may indicate a compromised device by monitoring high-value security metrics from the cloud and AWS IoT Core and comparing them against expected device behavior that customers define. For example, AWS IoT Device Defender lets customers define how many ports are open on the device, who the device can talk to, where it is connecting from, and how much data it sends or receives. AWS IoT Device Defender also allows customers to use ML models to set device normal behavior (for example, the number of times customers' devices connect with

AWS IoT cloud every five minutes). Then, it monitors the device communication and traffic and alerts customers if something looks wrong according to defined behaviors or ML models (such as traffic from devices to a known malicious IP or a spike in connection attempts).

AWS IoT Device Defender publishes security alarms to the AWS IoT console, Amazon CloudWatch, and Amazon Simple Notification Service when an audit fails or when behavior anomalies are detected so customers can investigate and determine the root cause. For example, AWS IoT Device Defender can alert customers when device identities are accessing sensitive APIs. AWS IoT Device Defender also provides built-in mitigation actions customers can take to minimize the impact of security issues such as adding a thing to a thing group (for example, quarantine), updating a device certificate, replacing default policy version, and enabling IoT logging.

AWS IoT SiteWise – Edge and Cloud processing for industrial data

AWS IoT SiteWise is a managed service that allows industrial enterprises to collect, store, organize, and visualize thousands of sensor data streams across multiple industrial facilities. AWS IoT SiteWise includes software that runs on a gateway device that resides onsite in a facility, continuously collects the data from a historian or a specialized industrial server, and sends it to the AWS Cloud. Industrial companies can use AWS IoT SiteWise to monitor and improve processes in a single industrial site or across multiple facilities, understand and resolve equipment issues efficiently, and visualize operational data of devices and equipment with the SiteWise Monitor feature.

Security capabilities

AWS IoT SiteWise gateway supports connectivity over the OPC-UA, Modbus TCP, or Ethernet/IP (EIP) protocols. AWS IoT SiteWise offers additional security when supported in the protocols, such as using encryption and server authentication secrets to authenticate between OPC-UA data sources securing your industrial data as it moves from your servers to the gateway. If your gateway has a hardware security module, you can configure AWS IoT Greengrass to secure your gateway. For AWS IoT SiteWise Monitor, customers can follow the principle of least privilege by using the minimum set of access policy permissions for their portal users and implement a healthy password rotation policy by configuring an appropriate expiration for passwords.

Additionally, AWS IoT SiteWise Edge now offers many of these capabilities on-premises in support of low latency and network fault intolerant applications.

Appendix 2 – Government involvement in IoT

Countries

United States

The National Institute of Standards and Technology – Department of Commerce

The United States Department of Commerce is spearheading multiple efforts to address IoT security. The National Institute of Standards and Technology (NIST) published a whitepaper that brings to light topics that customers and government agencies alike consider when assessing the security of data and devices. In the whitepaper, readers are invited to assess these concerns and are provided recommendations on how to mitigate the problems. NIST also released NIST Internal Report (NISTIR) 8228, which identifies risks that may negatively impact IoT adoption. The document also offers recommendations for mitigating or reducing the effects of these concerns. NIST is also convening public and private partnerships, soliciting comments, and hosting workshops related to smart cities and international standardization of IoT, among a host of other initiatives. Though in its infancy, early indicators point to potential cybersecurity and privacy risks as serious challenges to the gains that governments and consumers can harness through IoT.

Department of Defense

Another example within the government is found in the defense community. In 2016, the Chief Information Officer of the United States Department of Defense (DoD) issued policy recommendations to address the vulnerabilities and risks to IoT. According to the policy recommendations, DoD already provisions millions of IoT devices and sensors across DoD facilities, vehicles, and medical devices and is considering incorporating them into weapons and intelligence systems. The complexity of securing IoT stems from the limited processing power of the devices to run firewalls and anti-malware, as well as the vast number of devices. This compounds vulnerability exposure to a different level than traditional mobile devices.

DoD's recommended approach and policy action to address IoT security risks include:

1. A security and privacy risk analysis supporting each IoT implementation and associated data streams
2. Encryption at every point, where costs are commensurate with risk and value
3. Monitoring IoT networks to identify anomalous traffic and emergent threat

Federal Trade Commission

The Federal Trade Commission (FTC) has been an important participant in IoT security conversations, pursuing action against device manufacturers who have misrepresented or demonstrated negligence

in their security commitments. The FTC has set its bar to *reasonable data security* and identified the following repeated security deficiencies in device manufacturers:

- Security not built into devices
- Developers are not training their employees on good security practices
- Not ensuring downstream security and compliance (by contracts)
- Lack of defense in depth strategies
- Lack of reasonable access controls (customers can bypass or guess default passwords)
- Lack of a data security program

State of California

California is among the first states within the United States to pass legislation on IoT. The current bills address issues such as security of device design and data protection, but do not have specific requirements of IoT manufacturers. Instead, lawmakers have focused on security at the design phase, writing in SB-327 Information privacy: connected devices that protection of data must be "appropriate to the nature and function of the device" and "appropriate to the information it may collect, contain, or transmit."

United Kingdom

The UK's Department for Digital, Culture, Media and Sport (DCMS) published the final version of its Code of Practice for Consumer IoT Security in October 2018. This Code of Practice was jointly drafted with the National Cyber Security Centre and included input from consumer associations, industry, and academia. The document provides 13 guidelines on how to achieve a "secure by design" approach for all organizations involved in developing, manufacturing, and retailing consumer IoT products.

The Code of Practice emphasizes three leading practices for enabling users to achieve the greatest and most immediate security benefits, and urges IoT stakeholders to prioritize them:

- **No default passwords** – Many users do not change the default password, which has been the source of many IoT security issues.
- **Implement a vulnerability disclosure policy** – IoT device, service, and app developers should have a vulnerability disclosure policy and public point of contact to allow for the reporting (and remediation) of vulnerabilities in a timely manner.
- **Keep software updated** – Software updates need to be timely, easy to implement, and not disruptive to the functioning of the device.

As evidenced by the approaches outlined by both the US and UK, the security of IoT will continue to be top of mind for governments. Efforts are also underway by national and international standards bodies to develop standards, guidelines, and best practices for securing IoT, including the International Organization for Standardization (ISO) IoT Reference Architecture and the International Telecommunication Union (ITU) study group on IoT and smart cities.

In the context of IoT, customers should have the flexibility of using existing, time-tested practices already in use in what's considered more traditional network cybersecurity. For example, when trying to identify vulnerabilities, detect irregularities, respond to potential incidents, and recover from damage or disruption to IoT devices, customers can use the cybersecurity controls mapped against the NIST Cybersecurity Framework (CSF). This foundational set of cybersecurity disciplines is recognized globally and has been supported by governments and industries as a recommended baseline for use by any organization, regardless of its sector or size. The advantage of utilizing the NIST CSF is not just in its reputation, but also in the flexibility it allows for applying cybersecurity while keeping in mind its effect

on physical, cyber, and people dimensions. Along with the human aspect, the framework applies to organizations relying on technology, whether the focus is primarily on information technology, ICS, cyber-physical systems, or IoT.

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

AWS glossary

For the latest AWS terminology, see the AWS glossary in the *AWS General Reference*.