Amazon EventBridge

사용 설명서

Amazon EventBridge: 사용 설명서

Table of Contents

Amazon EventBridge란 무엇입니까?

Amazon EventBridge는 다양한 소스의 데이터와 애플리케이션을 쉽게 연결할 수 있는 서버리스 이벤트 버스 서비스입니다. EventBridge는 자체 애플리케이션, SaaS(Software-as-a-Service) 애플리케이션 및 AWS 서비스의 실시간 데이터 스트림을 제공한 다음, 해당 데이터를 AWS Lambda 등의 대상으로 라우팅합니다. 데이터를 전송할 대상을 결정하는 라우팅 규칙을 설정하여 모든 데이터 원본에 실시간으로 대응하는 애플리케이션 아키텍처를 구축할 수 있습니다. EventBridge를 사용하면 느슨하게 결합되고 분산된 이벤트 중심 아키텍처를 구축할 수 있습니다.

EventBridge은 이전에 Amazon CloudWatch Events였습니다. SaaS 파트너 및 자체 애플리케이션에서 이벤트를 수신할 수 있는 새로운 기능이 포함되어 있습니다. 기존 CloudWatch 이벤트 사용자는 새 EventBridge 콘솔과 CloudWatch 이벤트 콘솔에서 기존 기본 버스, 규칙 및 이벤트에 액세스할 수 있습니다. EventBridge는 동일한 CloudWatch 이벤트 API를 사용하므로 모든 기존 CloudWatch 이벤트 API 사용량은 동일하게 유지됩니다.

EventBridge를 위한 대상으로서 다음과 같은 AWS 리소스를 구성할 수 있습니다.

- Lambda 함수
- Amazon EC2 인스턴스
- Amazon Kinesis Data Streams의 스트림
- Amazon Kinesis Data Firehose의 전송 스트림
- Amazon CloudWatch Logs의 로그 그룹
- Amazon ECS 작업
- 시스템 관리자 Run Command
- 시스템 관리자 자동화
- AWS Batch 작업
- AWS Step Functions 상태 시스템
- AWS CodePipeline의 파이프라인
- AWS CodeBuild 프로젝트
- Amazon Inspector 평가 템플릿
- Amazon SNS 주제
- Amazon SQS 대기열
- 기본 제공 대상 - `EC2 CreateSnapshot API call`, `EC2 RebootInstances API call`, `EC2 StopInstances API call` 및 `EC2 TerminateInstances API call`.
- 다른 AWS 계정의 기본 이벤트 버스

개념

EventBridge 사용을 시작하기 전에 다음 개념을 이해해야 합니다.

- 이벤트 – 이벤트는 환경의 변화를 나타냅니다. 즉, AWS 환경, SaaS 파트너 서비스나 애플리케이션 또는 사용자 지정 애플리케이션이나 서비스일 수 있습니다. 예를 들어 Amazon EC2는 EC2 인스턴스의 상태가 보류에서 실행으로 변경될 때 이벤트를 생성하며, Amazon EC2 Auto Scaling은 인스턴스가 시작 또는 종료될 때 이벤트를 생성합니다. AWS CloudTrail은 API가 호출될 때 이벤트를 게시합니다. 또한 정기적으로 생성되는 예약 이벤트를 설정할 수도 있습니다. 각 서비스에서 이벤트와 샘플 이벤트를 생성하는 서비스의 목록은 EventBridge 지원되는 AWS 서비스의 이벤트 예제 (p. 57) 단원을 참조하십시오.
- 규칙 – 규칙은 들어오는 이벤트에서 일치하는 것을 찾아서 대상으로 라우팅하여 처리합니다. 단일 규칙으로 여러 개의 대상으로 라우팅을 할 수 있으며, 이들은 모두 병렬 처리됩니다. 규칙이 처리되는 특정한 순

서는 없습니다. 따라서 한 조직의 서로 다른 부분들이 자신이 관심 있는 이벤트를 찾아서 처리할 수 있습니다. 규칙은 특정 부분만 전달하거나 상수로 덮어쓰기를 해서 대상에 전송된 JSON을 사용자 지정할 수 있습니다.

- 대상 – 대상은 이벤트를 처리합니다. Amazon EC2 인스턴스, Lambda 함수, Kinesis 스트림, Amazon ECS 작업, Step Functions 상태 시스템, Amazon SNS 주제, Amazon SQS 대기열, 기본 제공 대상 등이 여기에 해당됩니다. 대상은 JSON 형식으로 이벤트를 수신합니다.

  규칙의 대상은 규칙과 같은 리전에 있어야 합니다.
- 이벤트 버스 – 이벤트 버스는 이벤트를 수신합니다. 규칙을 생성할 때 해당 규칙을 특정 이벤트 버스와 연결해야 하며, 규칙은 해당 이벤트 버스가 수신한 이벤트에만 일치하게 됩니다.

  계정에는 AWS 서비스로부터 이벤트를 수신하는 하나의 기본 이벤트 버스가 있습니다. 사용자 정의 애플리케이션에서 이벤트를 수신할 사용자 지정 이벤트 버스를 생성할 수 있습니다. SaaS 파트너 애플리케이션에서 이벤트를 수신할 파트너 이벤트 버스를 생성할 수도 있습니다.
- Partner event sources(파트너 이벤트 소스) – 파트너 이벤트 소스는 AWS 파트너가 AWS 고객 계정으로 이벤트를 보내는 데 사용됩니다. 이러한 이벤트를 수신하려면 고객이 이벤트 버스를 파트너 이벤트 소스와 연결해야 합니다.

관련 AWS 서비스

EventBridge에서 다음 서비스를 사용할 수 있습니다.

- AWS CloudTrail은 AWS Management 콘솔, AWS CLI 및 기타 서비스를 통해 이루어진 호출을 포함하여 계정에서 EventBridge API에 대한 호출을 모니터링할 수 있도록 합니다. CloudTrail 로깅이 활성화되면 EventBridge는 S3 버킷에 로그 데이터를 기록합니다. 각 로그 파일에는 하나 이상의 레코드가 포함이 되며, 요청을 충족하기 위해 수행되는 작업의 수에 따라 그 수가 결정됩니다. 자세한 내용은 Amazon EventBridge의 로깅 및 모니터링 (p. 147) 단원을 참조하십시오.
- AWS CloudFormation은 AWS 리소스를 모델링 및 설정할 수 있도록 해줍니다. 필요한 모든 AWS 리소스를 설명하는 템플릿을 생성하면 AWS CloudFormation이 해당 리소스의 프로비저닝과 구성을 담당합니다. AWS CloudFormation 형식의 EventBridge 규칙을 사용할 수 있습니다. 자세한 내용은 AWS CloudFormation 사용 설명서의 AWS::Events::Rule을 참조하십시오.
- AWS Config는 AWS 리소스에 대한 구성 변경을 기록할 수 있도록 해줍니다. 여기에는 리소스 간의 관계와 과거 리소스가 구성된 방법이 포함되므로 시간이 지나면서 구성과 관계가 어떻게 변하는지 확인할 수 있습니다. 또한 AWS Config 규칙을 생성해서 규칙을 준수하거나 준수하지 않는 리소스 유형과 조직의 정책을 확인할 수 있습니다. 자세한 내용은 AWS Config 개발자 안내서를 참조하십시오.
- AWS Identity and Access Management(IAM)은 사용자를 위해 AWS 리소스에 대한 액세스를 안전하게 제어할 수 있도록 해줍니다. IAM을 사용하여 AWS 리소스를 사용할 수 있는 사람을 제어(인증)하고 해당 사용자가 사용할 수 있는 리소스 및 사용 방법을 제어(권한 부여)할 수 있습니다. 자세한 내용은 Amazon EventBridge의 자격 증명 및 액세스 관리 (p. 120) 단원을 참조하십시오.
- Amazon Kinesis Data Streams는 빠르고 지속적인 데이터 인테이크 및 집계를 지원합니다. 사용되는 데이터 유형으로는 IT 인프라 로그 데이터, 애플리케이션 로그, 소셜 미디어, 시장 데이터 피드, 웹 클릭스트림 데이터 등이 있습니다. 데이터 인테이크 및 처리에 대한 응답이 실시간으로 이루어지기 때문에 일반적으로 간소화된 방식으로 처리가 됩니다. 자세한 내용은 Amazon Kinesis Data Streams 개발자 안내서를 참조하십시오.
- AWS Lambda는 새 정보에 신속하게 응답하는 애플리케이션을 구축할 수 있도록 해줍니다. Lambda 함수 형태로 애플리케이션 코드를 업로드하면 Lambda는 고가용성 컴퓨팅 인프라에서 코드를 실행합니다. Lambda는 서버 및 운영 체제 유지 관리, 용량 프로비저닝 및 자동 조정, 코드 및 보안 패치 배포, 코드 모니터링 및 로깅 등 모든 컴퓨팅 리소스 관리를 수행합니다. 자세한 내용은 AWS Lambda Developer Guide 단원을 참조하십시오.

Amazon EventBridge 설정

Amazon EventBridge를 사용하려면 AWS 계정이 있어야 합니다. AWS 계정이 있어야 서비스(예: Amazon EC2)를 사용해 웹 기반 인터페이스인 CloudWatch 콘솔에서 확인 가능한 이벤트를 생성할 수 있습니다. 뿐만 아니라 AWS Command Line Interface(AWS CLI)를 설치 및 구성하여 명령줄 인터페이스를 사용할 수 있습니다.

Amazon Web Services(AWS)에 가입

AWS 계정을 생성하면 모든 AWS 서비스에 자동으로 계정이 등록됩니다. 사용한 서비스에 대해서만 지불하면 됩니다.

이미 AWS 계정이 있다면 다음 단계로 건너뛰십시오. AWS 계정이 없는 경우에는 아래 단계를 수행하여 계정을 만드십시오.

AWS 계정에 가입하려면 다음을 수행합니다.

1. https://portal.aws.amazon.com/billing/signup을 엽니다.
2. 온라인 지시 사항을 따릅니다.

   등록 절차 중 전화를 받고 전화 키패드를 사용하여 확인 코드를 입력하는 과정이 있습니다.

Amazon EventBridge 콘솔에 로그인

Amazon EventBridge 콘솔에 로그인하려면

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.

계정 자격 증명

EventBridge 액세스에 루트 사용자 자격 증명을 사용할 수는 있지만 AWS Identity and Access Management(IAM) 계정을 사용하는 것이 좋습니다. EventBridge 액세스에 IAM 계정을 사용하고 있는 경우에는 반드시 다음 권한이 있어야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "events:*",
        "iam:PassRole"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

자세한 내용은 인증 (p. 121) 단원을 참조하십시오.

명령행 인터페이스 설정

AWS CLI를 사용하여 EventBridge 작업을 수행할 수 있습니다.

AWS CLI의 설치 및 구성 방법에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의 AWS Command Line Interface 초기 설정을 참조하십시오.

리전 엔드포인트

EventBridge를 사용하려면 리전 엔드포인트(기본)를 활성화해야 합니다. 자세한 내용은 IAM 사용 설명서의 AWS 리전에서 AWS STS 활성화 및 비활성화를 참조하십시오.

Amazon EventBridge 시작하기

이 단원에 나오는 절차를 사용하여 EventBridge 규칙 및 이벤트 버스를 만들고 삭제합니다. 시나리오와 대상별 자습서는 Amazon EventBridge 자습서 (p. 12)를 참조하십시오.

목차

제한 사항

- 규칙과 연결하는 대상은 규칙과 동일한 리전에 있어야 합니다.
- 모든 리전에서 사용할 수 없는 대상 유형도 일부 있습니다. 자세한 정보는 Amazon Web Services 일반 참조의 리전 및 엔드포인트를 참조하십시오.
- 기본 제공 대상을 통한 규칙 생성은 AWS Management 콘솔에서만 지원됩니다.
- 암호화된 Amazon SQS 대기열이 포함된 규칙을 대상으로 생성한 경우 AWS Key Management Service 키 정책에 다음 섹션이 포함되어야 합니다. 그러면 암호화된 대기열로 이벤트를 성공적으로 전달할 수 있습니다.

```
{
                "Sid": "Allow CWE to use the key",
                "Effect": "Allow",
                "Principal": {
                                "Service": "events.amazonaws.com"
                },
                "Action": [
                                "kms:Decrypt",
                                "kms:GenerateDataKey"
                ],
                "Resource": "*"
}
```

AWS 리소스의 이벤트에서 트리거되는 EventBridge 규칙 생성

다음 단계를 사용하여 AWS 서비스에서 발생하는 이벤트에서 트리거되는 EventBridge 규칙을 생성합니다.

계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다. 계정의 AWS 서비스에서 이벤트를 트리거하는 규칙을 작성하려면 기본 이벤트 버스와 연결해야 합니다. AWS 서비스의 이벤트를 찾는 사용자 지정 이벤트 버스에서 규칙을 생성할 수 있지만, 이 규칙은 교차 계정 이벤트 제공을 통해 다른 계정에서 이러한 이벤트를 수신할 때만 트리거됩니다. 자세한 내용은 AWS 계정 간 이벤트 전송 및 수신 (p. 104) 단원을 참조하십시오.

이벤트에서 트리거되는 규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.

   규칙은 동일한 리전과 동일한 이벤트 버스의 다른 규칙과 동일한 이름을 가질 수 없습니다.
5. Define pattern(패턴 정의)에 대해 이벤트 패턴을 선택하십시오.
6. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
7. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
8. 서비스 이름에 대해 이벤트를 출력하는 서비스 이름을 선택하십시오.
9. 이벤트 유형에 대해 모든 이벤트를 선택하거나 이 규칙에 사용할 이벤트 유형을 선택하십시오. 모든 이벤트를 선택하면 이 AWS 서비스에서 출력한 모든 이벤트가 규칙과 일치합니다.

   이벤트 패턴을 사용자 지정하려면 편집을 선택하고 변경한 후 저장을 선택하십시오.
10. Select event bus(이벤트 버스 선택)에 대해 이 규칙과 연결할 이벤트 버스를 선택하십시오. 이 규칙이 자신의 AWS 계정에서 오는 발생하는 해당 이벤트에서 트리거되도록 하려면 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
11. 대상 추가에서 선택한 유형의 이벤트가 감지되면 작동할 AWS 서비스를 선택합니다.
12. 필요할 경우 이 섹션의 다른 필드에 이 대상 유형에 관련된 정보를 입력합니다.
13. 여러 대상 유형에 대해 EventBridge에서는 대상에 이벤트를 보낼 권한이 필요합니다. 이 경우 EventBridge는 규칙 실행에 필요한 IAM 역할을 생성할 수 있습니다.

    - IAM 역할을 자동으로 생성하려면 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
    - 이전에 생성한 IAM 역할을 사용하려면 기존 역할 사용을 선택합니다.
14. (선택 사항) 이 규칙에 다른 대상을 추가하려면 Add target(대상 추가)을 선택하십시오.
15. (선택 사항) 규칙에 대해 하나 이상의 태그를 입력하십시오. 자세한 내용은 Amazon EventBridge 리소스에 태그 지정 (p. 152) 단원을 참조하십시오.
16. Create를 선택합니다.

AWS CloudTrail을 사용하여 AWS API 호출에서 트리거되는 EventBridge 규칙 생성

이벤트를 출력하지 않는 AWS 서비스의 작업에서 트리거되는 규칙을 생성하려면 해당 서비스에서 한 API 호출을 기반으로 규칙을 생성하면 됩니다. API 호출은 AWS CloudTrail에 의해 기록됩니다. 규칙의 트리거로 사용할 수 있는 API 호출에 대한 자세한 정보는 CloudTrail 이벤트 기록에서 지원하는 서비스를 참조하십시오.

EventBridge의 규칙은 해당 규칙이 생성된 리전에서만 작동합니다. 여러 리전에서 API 호출을 추적하도록 CloudTrail을 구성했는데, 해당하는 각 리전에서 CloudTrail을 기반으로 규칙이 트리거되도록 하려면 추적하려는 각 리전에서 별도의 규칙을 생성해야 합니다.

CloudTrail을 통해 전달되는 모든 이벤트는 `detail-type`의 값으로 `AWS API Call via CloudTrail`을 보유합니다.

> **Note**
>
> 우연히 무한 루프, 즉 반복 실행되는 규칙이 생성될 수 있습니다. 예를 들어 규칙이 S3 버킷에서 ACL이 바뀐 것을 감지할 경우 소프트웨어를 트리거하여 ACL을 원하는 상태로 변경합니다. 이때 규칙을 부주의하게 작성하면 ACL에 대한 변경이 이어져 규칙을 다시 실행하면서 무한 루프에 빠지게 됩니다.

이를 방지하려면 트리거된 작업이 동일한 규칙을 다시 실행하지 못하도록 규칙을 작성해야 합니다. 예를 들어 ACL이 변경 이후가 아니고 잘못된 상태일 때만 규칙이 실행되도록 할 수 있습니다. 무한 루프는 예상보다 높은 요금을 빠르게 야기할 수 있습니다. 따라서 요금이 지정된 한도를 초과할 경우 이를 알려줄 수 있는 예산 관리를 사용하는 것이 좋습니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 예산을 통해 비용 관리를 참조하십시오.

CloudTrail을 통해 API 호출에서 트리거되는 규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.

   규칙은 동일한 리전과 동일한 이벤트 버스의 다른 규칙과 동일한 이름을 가질 수 없습니다.
5. Define pattern(패턴 정의)에 대해 이벤트 패턴을 선택하십시오.
6. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
7. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
8. 서비스 이름에 대해 이벤트를 출력하는 서비스 이름을 선택하십시오.
9. 이벤트 유형에 대해 CloudTrail을 통해 AWS API 호출을 선택합니다.

   이벤트 패턴을 사용자 지정하려면 편집을 선택하고 변경한 후 저장을 선택하십시오.
10. Select event bus(이벤트 버스 선택)에 대해 이 규칙과 연결할 이벤트 버스를 선택하십시오.
11. Select event bus(이벤트 버스 선택)에 대해 이 규칙과 연결할 이벤트 버스를 선택하십시오. 이 규칙이 자신의 AWS 계정에서 오는 발생하는 해당 이벤트에서 트리거되도록 하려면 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
12. 필요할 경우 이 섹션의 다른 필드에 이 대상 유형에 관련된 정보를 입력합니다.
13. 여러 대상 유형에 대해 EventBridge에서는 대상에 이벤트를 보낼 권한이 필요합니다. 이 경우 EventBridge는 규칙 실행에 필요한 IAM 역할을 생성할 수 있습니다.

    - IAM 역할을 자동으로 생성하려면 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
    - 이전에 생성한 IAM 역할을 사용하려면 기존 역할 사용을 선택합니다.
14. (선택 사항) 이 규칙에 다른 대상을 추가하려면 Add target(대상 추가)을 선택하십시오.
15. (선택 사항) 규칙에 대해 하나 이상의 태그를 입력하십시오. 자세한 내용은 Amazon EventBridge 리소스에 태그 지정 (p. 152) 단원을 참조하십시오.
16. Create를 선택합니다.

일정에 따라 트리거되는 EventBridge 규칙 생성

다음 단계를 사용하여 정기적인 일정으로 트리거되는 EventBridge 규칙을 생성합니다. 기본 이벤트 버스를 사용해야만 예약된 규칙을 생성할 수 있습니다.

정기적인 일정에 따라 트리거되는 규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.

   규칙은 동일한 리전과 동일한 이벤트 버스의 다른 규칙과 동일한 이름을 가질 수 없습니다.
5. Define pattern(패턴 정의)에 대해 일정을 선택하십시오.

6. 고정 비율을 선택하고 작업이 실행되는 빈도를 지정하거나 Cron 표현식을 선택하고 작업이 트리거되는 시기를 정의하는 Cron 표현식을 지정합니다. Cron 식 구문에 대한 자세한 내용은 규칙에 대한 예약 표현식 (p. 32)을 참조하십시오.
7. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 예약된 규칙은 기본 이벤트 버스에만 지원됩니다.
8. Add target(대상 선택)에 대해 지정된 일정에 따라 작동할 AWS 서비스를 선택하십시오.
9. 필요할 경우 이 섹션의 다른 필드에 이 대상 유형에 관련된 정보를 입력합니다.
10. 여러 대상 유형에 대해 EventBridge에서는 대상에 이벤트를 보낼 권한이 필요합니다. 이 경우 EventBridge는 규칙 실행에 필요한 IAM 역할을 생성할 수 있습니다.

    - IAM 역할을 자동으로 생성하려면 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
    - 이전에 생성한 IAM 역할을 사용하려면 기존 역할 사용을 선택합니다.
11. (선택 사항) 이 규칙에 다른 대상을 추가하려면 Add target(대상 추가)을 선택하십시오.
12. (선택 사항) 규칙에 대해 하나 이상의 태그를 입력하십시오. 자세한 내용은 Amazon EventBridge 리소스에 태그 지정 (p. 152) 단원을 참조하십시오.
13. Create를 선택합니다.

SaaS 파트너로부터 이벤트 수신

SaaS 파트너 애플리케이션과 서비스에서 이벤트를 수신하려면 파트너로부터 제공 받은 파트너 이벤트 소스가 있어야 합니다. 그런 다음 파트너 이벤트 버스를 생성하여 해당 파트너 이벤트 소스와 일치시킬 수 있습니다.

파트너 이벤트 소스는 다음 리전에서 사용 가능합니다.

Code	이름
us-east-1	미국 동부(버지니아 북부)
us-east-2	미국 동부(오하이오)
us-west-1	미국 서부(캘리포니아 북부 지역)
us-west-2	미국 서부(오레곤)
ca-central-1	캐나다(중부)
eu-central-1	유럽(프랑크푸르트)
eu-west-1	유럽(아일랜드)
eu-west-2	유럽(런던)
eu-west-3	유럽(파리)
eu-north-1	유럽(스톡홀름)
ap-east-1	아시아 태평양(홍콩)
ap-northeast-1	아시아 태평양(도쿄)
ap-northeast-2	아시아 태평양(서울)
ap-southeast-1	아시아 태평양(싱가포르)
ap-southeast-2	아시아 태평양(시드니)

Code	이름
ap-south-1	아시아 태평양(뭄바이)
sa-east-1	남아메리카(상파울루)

SaaS 파트너로부터 이벤트를 받기 시작하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 Partner event sources(파트너 이벤트 소스)를 선택하십시오.
3. 원하는 파트너를 찾고 해당 파트너에 대해 Set up(설정)을 선택하십시오.
4. 계정 ID를 클립보드에 복사하려면 복사를 선택하십시오.
5. 탐색 창에서 Partner event sources(파트너 이벤트 소스)를 선택하십시오.
6. 파트너 웹사이트로 이동하여 지침에 따라 파트너 이벤트 소스를 생성하십시오. 이를 위해 계정 ID를 사용하십시오. 생성한 이벤트 소스는 본인 계정에서만 사용할 수 있습니다.
7. EventBridge 콘솔로 돌아가서 탐색 창에서 Partner event sources(파트너 이벤트 소스)를 선택하십시오.
8. 파트너 이벤트 소스 옆에 있는 버튼을 선택하고 Associate with event bus(이벤트 버스와 연결)를 선택하십시오.

   해당 이벤트 소스의 상태가 `Pending`에서 `Active`로 변경되고 파트너 이벤트 소스 이름과 일치하도록 이벤트 버스 이름이 업데이트됩니다. 이제 해당 파트너 이벤트 소스에서 이벤트에서 트리거되는 규칙 생성을 시작할 수 있습니다. 자세한 내용은 SaaS 파트너 이벤트에서 트리거되는 규칙 생성 (p. 9) 단원을 참조하십시오.

SaaS 파트너 이벤트에서 트리거되는 규칙 생성

SaaS 파트너 애플리케이션 및 서비스의 이벤트에 대한 규칙을 생성하려면 파트너 이벤트 버스를 생성하여 해당 파트너 이벤트 소스와 일치시켜야 합니다. 자세한 내용은 SaaS 파트너로부터 이벤트 수신 (p. 8) 단원을 참조하십시오.

SaaS 파트너의 이벤트에서 트리거되는 규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 이벤트 패턴을 선택하십시오.
6. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
7. Service provider(서비스 제공업체)에 대해 Service partners(서비스 파트너)를 선택하십시오.
8. 서비스 이름에 대해 파트너 이름을 선택하십시오.
9. 이벤트 유형에 대해 모든 이벤트를 선택하거나 이 규칙에 사용할 이벤트 유형을 선택하십시오. 모든 이벤트를 선택하면 이 파트너 이벤트 소스가 출력한 모든 이벤트가 규칙과 일치합니다.

   이벤트 패턴을 사용자 지정하려면 편집을 선택하고 변경한 후 저장을 선택하십시오.
10. Service event bus(서비스 이벤트 버스)의 경우 이 파트너에 해당하는 이벤트 버스가 선택되었는지 확인하십시오.
11. 대상 추가에서 선택한 유형의 이벤트가 감지되면 작동할 AWS 서비스를 선택합니다.
12. 필요할 경우 이 섹션의 다른 필드에 이 대상 유형에 관련된 정보를 입력합니다.
13. 여러 대상 유형에 대해 EventBridge에서는 대상에 이벤트를 보낼 권한이 필요합니다. 이 경우 EventBridge는 규칙 실행에 필요한 IAM 역할을 생성할 수 있습니다.

- IAM 역할을 자동으로 생성하려면 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
- 이전에 생성한 IAM 역할을 사용하려면 기존 역할 사용을 선택합니다.

14. (선택 사항) 이 규칙에 다른 대상을 추가하려면 Add target(대상 추가)을 선택하십시오.
15. (선택 사항) 규칙에 대해 하나 이상의 태그를 입력하십시오. 자세한 내용은 Amazon EventBridge 리소스에 태그 지정 (p. 152) 단원을 참조하십시오.
16. Create를 선택합니다.

이벤트 버스 생성

계정에는 하나의 기본 이벤트 버스가 포함되어 있으며, 이는 AWS 서비스에서 출력한 이벤트를 수신합니다. 이벤트를 기본 이벤트 버스로 보내도록 사용자 지정 애플리케이션을 구성할 수도 있습니다.

계정에서 두 가지 유형의 추가 이벤트 버스를 생성할 수 있습니다.

- Partner event buses(파트너 이벤트 버스): AWS SaaS(Software as a Service) 파트너가 생성한 애플리케이션 및 서비스로부터 이벤트를 수신할 수 있습니다. SaaS 파트너로부터 이벤트를 수신하려면 이벤트를 수신하려는 각 파트너 이벤트 소스에 대한 파트너 이벤트 버스를 생성해야 합니다.

  자세한 내용은 SaaS 파트너로부터 이벤트 수신 (p. 8) 단원을 참조하십시오.
- Custom event buses(사용자 지정 이벤트 버스): 사용자 지정 애플리케이션 및 서비스로부터 이벤트를 수신할 수 있습니다.

  계정의 각 이벤트 버스에는 최대 100개의 EventBridge 규칙이 연결될 수 있으므로, 계정에 많은 규칙이 있는 경우 사용자 지정 애플리케이션 이벤트의 일부 규칙과 연결할 사용자 지정 이벤트 버스를 생성할 수 있습니다. 사용자 지정 이벤트 버스를 생성하는 또 한 가지 이유는 서로 다른 이벤트 버스에 서로 다른 권한을 적용하려는 것입니다. 이벤트 버스에 권한을 설정할 때, 어떤 다른 계정 또는 전체 조직이 이벤트 버스에 이벤트를 보낼 수 있는지 지정할 수 있습니다.

사용자 지정 이벤트 버스 생성

사용자 정의 애플리케이션에서 이벤트를 수신할 사용자 지정 이벤트 버스를 생성할 수 있습니다. 애플리케이션에서 기본 이벤트 버스로 이벤트를 보낼 수도 있습니다.

사용자 지정 이벤트 버스를 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 이벤트 버스를 선택합니다.
3. Create event bus(이벤트 버스 생성)를 선택하십시오.
4. 새 이벤트 버스의 이름을 입력하십시오.
5. 다른 계정 또는 전체 조직이 이 이벤트 버스로 이벤트를 보낼 수 있도록 하려면 Other AWS account(다른 AWS 계정), 조직 또는 둘 다 선택하십시오.

   a. Other AWS account(다른 AWS 계정)을 선택한 경우 Individual AWS account ID(개별 AWS 계정 ID) 또는 All AWS accounts(모든 AWS 계정)를 선택하십시오. Individual AWS account ID(개별 AWS 계정 ID)를 선택한 경우 계정 ID를 입력하십시오. 더 많은 계정을 추가하려면 계정 추가를 선택하십시오.

      All AWS accounts(모든 AWS 계정)를 선택할 경우 다른 계정으로부터 수신하는 이벤트만 일치시키는 규칙을 생성하도록 주의하십시오. 더욱 안전한 규칙을 생성하려면 이벤트를 수신할 계정 하나 이상의 계정 ID가 입력되는 `Account` 필드가 각 규칙의 모든 패턴에 포함되어야 합니다. 이벤트 패턴에 `Account` 필드가 포함되는 규칙은 다른 계정에서 전송된 이벤트와 일치하지 않습니다.

b. 조직을 선택할 경우에는 내 조직을 선택하여 계정이 속한 조직의 모든 계정에게 권한을 부여합니다. 또는 Other organization(다른 조직)을 선택하고 o- 접두사를 포함하여 조직 ID를 입력하십시오. My organization(내 조직)은 계정이 조직의 구성원인 경우에만 사용할 수 있습니다.

   Other organization(다른 조직)을 선택하고 더 많은 조직을 추가하려면 Add organization(조직 추가)을 선택하십시오.

6. Create를 선택합니다.

EventBridge 규칙 삭제 또는 비활성화

다음 단계를 사용하여 EventBridge 규칙을 삭제하거나 비활성화합니다.

규칙을 삭제하거나 비활성화하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.

   Event bus(이벤트 버스)에서 규칙과 연결된 이벤트 버스를 선택하십시오.
3. 다음 중 하나를 수행하십시오.

   a. 규칙을 삭제하려면 규칙 옆의 버튼을 선택하고 작업, 삭제, 삭제를 선택합니다.

      규칙이 관리형 규칙인 경우 해당 규칙이 관리형 규칙이고, 규칙을 삭제하면 규칙을 생성한 서비스에서 기능이 멈출 수 있음을 확인하도록 규칙 이름을 입력해야 합니다. 계속하려면 규칙 이름을 입력하고 강제 삭제를 선택합니다. 자세한 내용은 Amazon EventBridge 관리형 규칙 (p. 118) 단원을 참조하십시오.

   b. 규칙을 일시적으로 비활성화하려면 규칙 옆의 버튼을 선택하고 비활성화, 비활성화를 선택합니다.

      관리형 규칙은 비활성화할 수 없습니다.

Amazon EventBridge 자습서

다음 자습서에서는 특정 작업과 대상의 EventBridge 규칙을 생성하는 방법을 보여줍니다.

자습서:

자습서: EventBridge를 사용하여 AWS 시스템 관리자 Run Command로 이벤트 릴레이

Amazon EventBridge를 사용하여 AWS 시스템 관리자 Run Command를 호출하고 특정 이벤트가 발생할 때 Amazon EC2 인스턴스에서 작업을 수행할 수 있습니다. 이 자습서에서는 시스템 관리자 Run Command를 사용하여 셸 명령을 실행하고 Amazon EC2 Auto Scaling 그룹에서 시작되는 각각의 인스턴스를 새로 구성합니다. 이 자습서에서는 키로 `environment`를, 값으로 `production`을 지정하는 등 Amazon EC2 Auto Scaling 그룹에 태그를 이미 지정했다고 가정합니다.

EventBridge 규칙을 만들려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.
   a. [Event pattern]을 선택합니다.
   b. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
   c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
   d. 서비스 이름에서 Auto Scaling을 선택합니다.
   e. 이벤트 유형에서 인스턴스 시작 및 종료를 선택합니다.

f. 특정 인스턴스 이벤트와 EC2 인스턴스-시작 수명 주기 작업을 선택합니다.

g. 기본적으로 규칙은 리전의 모든 Amazon EC2 Auto Scaling 그룹과 일치합니다. 규칙을 특정 그룹과 일치시키려면 특정 그룹 이름을 선택하고 그룹을 하나 이상 선택합니다.

6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에 대해 SSM Run Command를 선택하십시오.
8. 문서에서 AWS-RunShellScript를 선택합니다.

   대상 키에 **`tag:environment`**를 입력합니다. 대상 값에 **`production`**를 입력하고 추가를 선택하십시오.
9. Configure automation parameter(s)에서 다음을 수행하십시오.

   a. 상수를 선택하십시오.

   b. 명령에 셸 명령을 입력하고 추가를 선택합니다. 인스턴스가 시작될 때 모든 명령이 실행되도록 이 단계를 반복합니다.

   c. 필요할 경우 WorkingDirectory와 ExecutionTimeout에 해당 정보를 입력합니다.
10. EventBridge는 이벤트 실행에 필요한 IAM 역할을 생성할 수 있습니다.

    - IAM 역할을 자동으로 생성하려면 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
    - 이전에 생성한 IAM 역할을 사용하려면 기존 역할 사용을 선택합니다.
11. [Create rule]을 선택합니다.

자습서: EventBridge를 사용하여 Amazon EC2 인스턴스의 상태 기록

Amazon EC2 인스턴스의 상태 변경을 기록하는 AWS Lambda 함수를 생성할 수 있습니다. 상태 전환이 있거나 관심이 있는 하나 이상의 상태로 전환될 때마다 함수를 실행하도록 규칙을 생성할 수 있습니다. 이 자습서에서는 새 인스턴스의 시작을 기록합니다.

1단계: AWS Lambda 함수 만들기

Lambda 함수를 생성하여 상태 변경 이벤트를 기록합니다. 규칙을 생성할 때 이 함수를 지정합니다.

Lambda 함수를 만들려면

1. https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.
2. Lambda를 처음 사용하는 경우 시작 페이지가 표시됩니다. 지금 시작을 선택합니다. 그렇지 않으면 Lambda 함수 생성을 선택합니다.
3. 블루프린트 선택 페이지에서 필터에 `hello`를 입력하고 hello-world 블루프린트를 선택합니다.
4. 트리거 구성 페이지에서 다음을 선택합니다.
5. 함수 구성 페이지에서 다음을 수행합니다.

   a. Lambda 함수의 이름과 설명을 입력합니다. 예를 들어, 함수 이름을 `LogEC2InstanceStateChange`로 지정합니다.

   b. Lambda 함수에 대한 샘플 코드를 편집합니다. 예:

```
'use strict';

exports.handler = (event, context, callback) => {
```

```
    console.log('LogEC2InstanceStateChange');
    console.log('Received event:', JSON.stringify(event, null, 2));
    callback(null, 'Finished');
};
```

c. 역할에서 기존 역할 선택을 선택합니다. 기존 역할의 경우 기본 실행 역할을 선택합니다. 그렇지 않다면 기본 실행 역할을 만듭니다.

d. [Next]를 선택합니다.

6. 검토 페이지에서 함수 생성을 선택합니다.

2단계: 규칙 생성

Amazon EC2 인스턴스를 시작할 때마다 Lambda 함수를 실행하는 규칙을 생성합니다.

EventBridge 규칙을 만들려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.

   a. [Event pattern]을 선택합니다.

   b. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.

   c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.

   d. 서비스 이름에서 EC2를 선택합니다.

   e. 이벤트 유형에서 EC2 인스턴스 상태 변경 알림을 선택합니다.

   f. 특정 상태, 실행을 선택합니다.

   g. 기본적으로 규칙은 리전의 모든 인스턴스 그룹과 일치합니다. 규칙을 특정 인스턴스와 일치시키려면 특정 인스턴스를 선택하고 인스턴스 ID를 하나 이상 입력합니다.

6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에 대해 Lambda함수를 선택하십시오.
8. 함수에서 생성한 Lambda 함수를 선택합니다.
9. Create를 선택합니다.

3단계: 규칙 테스트

규칙을 테스트하려면 Amazon EC2 인스턴스를 시작합니다. 인스턴스가 시작 및 초기화될 때까지 몇 분 기다린 후에 Lambda 함수가 호출되었는지 확인합니다.

인스턴스를 시작하여 규칙을 테스트하려면

1. https://console.aws.amazon.com/ec2/에서 Amazon EC2 콘솔을 엽니다.
2. 인스턴스 시작. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 인스턴스 시작을 참조하십시오.
3. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
4. 탐색 창에서 규칙을 선택하고 생성한 규칙의 이름을 선택한 후 규칙에 대한 지표를 선택합니다.
5. Lambda 함수에서 출력을 보려면 다음을 수행합니다.

a. https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
b. 탐색 창에서 로그를 선택합니다.
c. Lambda 함수에 대한 로그 그룹 이름(`/aws/lambda/function-name`)을 선택합니다.
d. 로그 스트림 이름을 선택하여 시작한 인스턴스에서 함수를 통해 제공된 데이터를 확인합니다.

6. (선택 사항) 작업이 완료되면 Amazon EC2 콘솔을 열고 시작한 인스턴스를 중지 또는 종료할 수 있습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 인스턴스 종료를 참조하십시오.

자습서: EventBridge를 사용하여 Auto Scaling 그룹의 상태 기록

Auto Scaling 그룹이 Amazon EC2 인스턴스를 시작하거나 종료할 때마다 이벤트를 기록하고 시작 이벤트가 성공했는지 종료 이벤트가 성공했는지를 기록하도록 AWS Lambda 함수를 실행할 수 있습니다.

Amazon EC2 Auto Scaling 이벤트를 사용한 추가 EventBridge 시나리오에 대한 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Auto Scaling 그룹 조정 시 CloudWatch 이벤트 수신을 참조하십시오.

1단계: AWS Lambda 함수 만들기

Lambda 함수를 생성하여 Auto Scaling 그룹에서 이벤트의 확장 및 축소를 기록합니다. 규칙을 생성할 때 이 함수를 지정합니다.

Lambda 함수를 만들려면

1. https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.
2. Lambda를 처음 사용하는 경우 시작 페이지가 표시됩니다. 지금 시작을 선택합니다. 그렇지 않으면 Lambda 함수 생성을 선택합니다.
3. 블루프린트 선택 페이지에서 필터에 `hello`를 입력하고 hello-world 블루프린트를 선택합니다.
4. 트리거 구성 페이지에서 다음을 선택합니다.
5. 함수 구성 페이지에서 다음을 수행합니다.

   a. Lambda 함수의 이름과 설명을 입력합니다. 예를 들어 함수 이름을 `LogAutoScalingEvent`로 지정합니다.
   b. Lambda 함수에 대한 샘플 코드를 편집합니다. 예:

```
'use strict';

exports.handler = (event, context, callback) => {
    console.log('LogAutoScalingEvent');
    console.log('Received event:', JSON.stringify(event, null, 2));
    callback(null, 'Finished');
};
```

   c. 역할에서 기존 역할 선택을 선택합니다. 기존 역할의 경우 기본 실행 역할을 선택합니다. 그렇지 않다면 기본 실행 역할을 만듭니다.
   d. [Next]를 선택합니다.

6. 함수 생성을 선택합니다.

2단계: 규칙 생성

Auto Scaling 그룹이 인스턴스를 시작 또는 종료할 때마다 Lambda 함수를 실행하는 규칙을 생성합니다.

규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.
   a. 이벤트 패턴을 선택합니다.
   b. Pre-defined by service(서비스별 사전 정의됨)을 선택하십시오.
   c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
   d. 서비스 이름에서 Auto Scaling을 선택합니다.
   e. 이벤트 유형에서 인스턴스 시작 및 종료를 선택합니다.
   f. 성공 및 실패한 모든 인스턴스 시작 및 종료 이벤트를 포착하려면 모든 인스턴스 이벤트를 선택합니다.
   g. 기본적으로 규칙은 리전의 모든 Auto Scaling 그룹과 일치합니다. 규칙을 특정 그룹과 일치시키려면 특정 그룹 이름을 선택하고 그룹을 하나 이상 선택합니다.
   h. 기본적으로 규칙은 리전의 모든 Auto Scaling 그룹과 일치합니다. 규칙을 특정 Auto Scaling 그룹과 일치시키려면 특정 그룹 이름을 선택하고 Auto Scaling 그룹을 하나 이상 선택합니다.
6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에 대해 Lambda함수를 선택하십시오.
8. 함수에서 생성한 Lambda 함수를 선택합니다.
9. Create를 선택합니다.

3단계: 규칙 테스트

인스턴스가 시작되도록 Auto Scaling 그룹을 수동으로 조정함으로써 규칙을 테스트할 수 있습니다. 확장 이벤트가 발생하도록 몇 분 기다린 후에 Lambda 함수가 호출되었는지 확인할 수 있습니다.

Auto Scaling 그룹을 사용하여 규칙을 테스트하려면

1. Auto Scaling 그룹의 크기를 늘리려면 다음을 수행합니다.
   a. https://console.aws.amazon.com/ec2/에서 Amazon EC2 콘솔을 엽니다.
   b. 탐색 창에서 Auto Scaling과 Auto Scaling 그룹을 선택합니다.
   c. Auto Scaling 그룹에 대한 확인란을 선택합니다.
   d. 세부 정보 탭에서 편집을 선택합니다. Desired(원하는)에서 한 개씩 원하는 용량을 늘립니다. 예를 들어, 현재 값이 2인 경우 3을 입력합니다. 원하는 용량은 그룹의 최대 크기보다 작거나 같아야 합니다. Desired(원하는)의 새 값이 최대보다 큰 경우 최대를 업데이트해야 합니다. 작업을 마쳤으면 [Save]를 선택합니다.
2. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
3. 탐색 창에서 규칙을 선택하고 생성한 규칙의 이름을 선택한 후 규칙에 대한 지표를 선택합니다.
4. Lambda 함수에서 출력을 보려면 다음을 수행합니다.
   a. https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
   b. 탐색 창에서 로그를 선택합니다.
   c. Lambda 함수에 대한 로그 그룹 이름(`/aws/lambda/`*`function-name`*)을 선택합니다.
   d. 로그 스트림 이름을 선택하여 시작한 인스턴스에서 함수를 통해 제공된 데이터를 확인합니다.

5. (선택 사항) 완료되면 Auto Scaling 그룹이 이전 크기로 돌아가도록 한 개씩 원하는 용량을 줄일 수 있습니다.

자습서: EventBridge를 사용하여 Amazon S3 객체 수준 작업 기록

S3 버킷에서 객체 수준 API 작업을 기록할 수 있습니다. Amazon EventBridge를 이러한 이벤트와 일치시키려면 먼저 AWS CloudTrail을 사용하여 이들 이벤트를 수신하기 위해 구성된 추적을 설정해야 합니다.

1단계: AWS CloudTrail 추적 구성

AWS CloudTrail 및 EventBridge에 S3 버킷의 데이터 이벤트를 기록하려면 추적을 생성합니다. 추적은 계정에서 API 호출과 관련 이벤트를 캡처하고 지정된 S3 버킷에 로그 파일을 전달합니다. 기존 추적을 업데이트하거나 생성할 수 있습니다.

추적을 생성하려면

1. https://console.aws.amazon.com/cloudtrail/에서 CloudTrail 콘솔을 엽니다.
2. 탐색 창에서 Trails(추적), Create trail(추적 생성)을 선택합니다.
3. 추적 이름에 추적 이름을 입력합니다.
4. Data events(데이터 이벤트)에 버킷 이름과 접두사(선택 사항)를 입력합니다. 각 추적의 경우 최대 250개의 Amazon S3 객체를 추가할 수 있습니다.
   - 버킷의 모든 Amazon S3 객체에 대한 데이터 이벤트를 로깅하려면 S3 버킷과 빈 접두사를 지정합니다. 이벤트가 해당 버킷의 개체에서 발생하면 추적이 해당 이벤트를 처리하고 기록합니다.
   - 특정 Amazon S3 객체에 대한 데이터 이벤트를 로깅하려면 S3 버킷 추가를 선택한 후 S3 버킷을 지정하고 객체 접두사(선택 사항)를 지정합니다. 이벤트가 해당 버킷의 개체에서 발생하고 개체가 지정된 접두사로 시작하면 추적이 이벤트를 처리하고 기록합니다.
5. 각 리소스에 대해 읽기 이벤트를 로깅할지, 쓰기 이벤트를 로깅할지 또는 둘 다 로깅할지를 지정합니다.
6. Storage location(스토리지 위치)에 대해 로그 파일 스토리지에 지정할 기존 S3 버킷을 생성하거나 선택합니다.
7. Create를 선택합니다.

자세한 내용은 AWS CloudTrail User Guide의 데이터 이벤트를 참조하십시오.

2단계: AWS Lambda 함수 만들기

Lambda 함수를 생성하여 S3 버킷에서 데이터 이벤트를 기록합니다. 규칙을 생성할 때 이 함수를 지정합니다.

Lambda 함수를 만들려면

1. https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.
2. Lambda를 처음 사용하는 경우 시작 페이지가 표시됩니다. 함수 만들기를 선택합니다. 그렇지 않은 경우에는 함수 생성을 선택합니다.
3. 새로 작성을 선택합니다.
4. 새로 작성에서 다음 작업을 수행합니다.
   a. Lambda 함수 이름을 입력합니다. 예를 들어 함수 이름을 `LogS3DataEvents`로 지정합니다.
   b. 역할에서 사용자 지정 역할 생성을 선택합니다.

새 창이 열립니다. 필요할 경우 역할 이름을 변경하고 허용을 선택합니다.

c. Lambda 콘솔로 돌아가서 함수 생성을 선택합니다.

5. Lambda 함수의 코드를 다음과 같이 편집하고 저장을 선택합니다.

```
'use strict';

exports.handler = (event, context, callback) => {
    console.log('LogS3DataEvents');
    console.log('Received event:', JSON.stringify(event, null, 2));
    callback(null, 'Finished');
};
```

3단계: 규칙 생성

Amazon S3 데이터 이벤트에 대한 응답으로 Lambda 함수를 실행하는 규칙을 생성합니다.

규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.

   a. 이벤트 패턴을 선택합니다.

   b. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.

   c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.

   d. 서비스 이름에서 Simple Storage Service(S3)를 선택합니다.

   e. 이벤트 유형에서 객체 수준 작업을 선택합니다.

   f. 특정 작업, PutObject를 선택합니다.

   g. 기본적으로 규칙은 리전의 모든 버킷에 대한 데이터 이벤트와 일치합니다. 특정 버킷에 대한 데이터 이벤트와 일치시키려면 Specify bucket(s) by name(이름 기준 특정 버킷)을 선택하고 버킷을 하나 이상 입력합니다.

6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에서 Lambda 함수를 선택합니다.
8. 함수에서 생성한 Lambda 함수를 선택합니다.
9. Create를 선택합니다.

4단계: 규칙 테스트

규칙을 테스트하려면 S3 버킷에 개체를 배치합니다. Lambda 함수가 호출되었는지 확인할 수 있습니다.

Lambda 함수에 대한 로그를 보려면

1. https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 로그를 선택합니다.
3. Lambda 함수에 대한 로그 그룹 이름(`/aws/lambda/function-name`)을 선택합니다.
4. 로그 스트림 이름을 선택하여 시작한 인스턴스에서 함수를 통해 제공된 데이터를 확인합니다.

추적에 지정한 S3 버킷의 CloudTrail 로그 내용을 확인할 수도 있습니다. 자세한 내용은 AWS CloudTrail User Guide의 CloudTrail 로그 파일 가져오기 및 보기를 참조하십시오.

자습서: 입력 변환기를 사용하여 이벤트 대상에 전달된 것을 사용자 지정

EventBridge의 입력 변환기 기능을 사용하여 이 규칙의 대상에 대한 입력이기 전에 이벤트로부터 가져온 텍스트를 사용자 지정할 수 있습니다.

이벤트로부터 여러 JSON 경로를 정의하고 그 출력을 다른 변수에 할당할 수 있습니다. 그런 다음 <*variable-name*>으로 입력 템플릿 내에서 이러한 변수를 사용할 수 있습니다. < 및 > 문자는 이스케이프되지 않습니다.

변수를 지정하여 이벤트에 존재하지 않는 JSON 경로를 일치시키는 경우 변수가 생성되지 않기 때문에 출력에 나타나지 않습니다.

이 자습서에서 인스턴스 상태 변경 이벤트에서 Amazon EC2 인스턴스의 `instance-id`와 `state`를 추출합니다. 입력 변환기를 사용하여 Amazon SNS 주제에 전송된 읽기 쉬운 메시지에 데이터를 포함합니다. 인스턴스 상태가 변경될 때 규칙이 트리거됩니다. 예를 들어 이 규칙을 사용하여 다음 Amazon EC2 인스턴스 상태-변경 알림 이벤트는 The EC2 instance i-1234567890abcdef0 has changed state to stopped라는 Amazon SNS 메시지를 생산합니다.

```
{
   "id":"7bf73129-1428-4cd3-a780-95db273d1602",
   "detail-type":"EC2 Instance State-change Notification",
   "source":"aws.ec2",
   "account":"123456789012",
   "time":"2015-11-11T21:29:54Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ec2:us-east-1:123456789012:instance/ i-1234567890abcdef0"
   ],
   "detail":{
      "instance-id":" i-1234567890abcdef0",
      "state":"stopped"
   }
}
```

`instance` 변수를 이벤트의 `$.detail.instance-id` JSON 경로로, `state` 변수를 `$.detail.state` JSON 경로로 매핑함으로써 이를 달성합니다. 그런 다음 입력 템플릿을 "The EC2 instance <instance> has changed state to <state>"로 설정합니다.

> **Note**
>
> 이벤트 변환기에 대한 자세한 내용은 대상 입력 변환 (p. 47) 단원을 참조하십시오.

규칙 생성

입력 변환기를 사용하여 대상으로 전송된 인스턴스 상태 변경 정보를 사용자 지정하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙의 이름과 설명을 입력합니다.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.

a. [Event pattern]을 선택합니다.
b. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
d. 서비스 이름에서 EC2를 선택합니다.
e. 이벤트 유형에서 EC2 인스턴스 상태 변경 알림을 선택합니다.
f. 모든 상태와 모든 인스턴스를 선택합니다.

6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에서 SNS 주제를 선택합니다.
8. 주제에서 Amazon EC2 인스턴스 상태 변경 시 알림을 받고자 하는 Amazon SNS 주제를 선택합니다.
9. 입력 구성, 입력 변환기를 선택합니다.
10. Input Path(입력 경로)에 {"state" : "$.detail.state", "instance" : "$.detail.instance-id"}를 입력합니다.
11. Input Template(입력 템플릿)에 "The EC2 instance <instance> has changed state to <state>."를 입력합니다.
12. Create를 선택합니다.

자습서: EventBridge를 사용하여 AWS API 호출 기록

각 AWS API 호출을 기록하는 AWS Lambda 함수를 사용할 수 있습니다. 예를 들어 Amazon EC2 내에서 어떤 작업이든 기록하도록 규칙을 생성하거나 특정 API 호출만 기록하도록 규칙을 제한할 수 있습니다. 이 자습서에서는 Amazon EC2 인스턴스가 중지될 때마다 기록을 합니다.

사전 조건

이러한 이벤트와 일치시키려면 먼저 AWS CloudTrail을 사용하여 추적을 설정해야 합니다. 추적이 없는 경우에는 다음 절차를 완료합니다.

추적을 생성하려면

1. https://console.aws.amazon.com/cloudtrail/에서 CloudTrail 콘솔을 엽니다.
2. Trails(추적), Create trail(추적 생성)을 선택합니다.
3. Trail name(추적 이름)에 추적 이름을 입력합니다.
4. 스토리지 위치의 경우 Create a new S3 bucket(새 S3 버킷 생성)에 CloudTrail에서 로그를 전송할 새 버킷의 이름을 입력합니다.
5. Create를 선택합니다.

1단계: AWS Lambda 함수 만들기

Lambda 함수를 생성하여 API 호출 이벤트를 기록합니다. 규칙을 생성할 때 이 함수를 지정합니다.

Lambda 함수를 만들려면

1. https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.
2. Lambda를 처음 사용하는 경우 시작 페이지가 표시됩니다. 지금 시작을 선택합니다. 그렇지 않으면 Lambda 함수 생성을 선택합니다.
3. 블루프린트 선택 페이지에서 필터에 `hello`를 입력하고 hello-world 블루프린트를 선택합니다.

4. 트리거 구성 페이지에서 다음을 선택합니다.
5. 함수 구성 페이지에서 다음을 수행합니다.

   a. Lambda 함수의 이름과 설명을 입력합니다. 예를 들어, 함수 이름을 `LogEC2StopInstance`로 지정합니다.
   b. Lambda 함수에 대한 샘플 코드를 편집합니다. 예:

```
'use strict';

exports.handler = (event, context, callback) => {
    console.log('LogEC2StopInstance');
    console.log('Received event:', JSON.stringify(event, null, 2));
    callback(null, 'Finished');
};
```

   c. 역할에서 기존 역할 선택을 선택합니다. 기존 역할의 경우 기본 실행 역할을 선택합니다. 그렇지 않다면 기본 실행 역할을 만듭니다.
   d. [Next]를 선택합니다.
6. 검토 페이지에서 함수 생성을 선택합니다.

2단계: 규칙 생성

Amazon EC2 인스턴스를 중지할 때마다 Lambda 함수를 실행하는 규칙을 생성합니다.

규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.

   a. [Event pattern]을 선택합니다.
   b. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
   c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
   d. 서비스 이름에서 EC2를 선택합니다.
   e. 이벤트 유형에 대해 CloudTrail을 통해 AWS API 호출을 선택합니다.
   f. Specific operations(s)(특정 작업)를 선택하고 상자에 `StopInstances`를 입력하십시오.
   g. 기본적으로 규칙은 리전의 모든 Amazon EC2 Auto Scaling 그룹과 일치합니다. 규칙을 특정 그룹과 일치시키려면 특정 그룹 이름을 선택하고 그룹을 하나 이상 선택합니다.
6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에서 대상 추가, Lambda 함수를 선택합니다.
8. 함수에서 생성한 Lambda 함수를 선택합니다.
9. Create를 선택합니다.

3단계: 규칙 테스트

Amazon EC2 콘솔을 사용하여 Amazon EC2 인스턴스를 중지함으로써 규칙을 테스트할 수 있습니다. 인스턴스가 중지되도록 몇 분 기다린 다음 CloudWatch 콘솔에서 AWS Lambda 지표를 통해 함수가 호출되었는지 확인합니다.

인스턴스를 중지시켜 규칙을 테스트하려면

1. https://console.aws.amazon.com/ec2/에서 Amazon EC2 콘솔을 엽니다.
2. 인스턴스 시작. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 인스턴스 시작을 참조하십시오.
3. 인스턴스를 중지합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 인스턴스 중지 및 시작을 참조하십시오.
4. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
5. 탐색 창에서 규칙을 선택하고 생성한 규칙의 이름을 선택한 후 규칙에 대한 지표를 선택합니다.
6. Lambda 함수에서 출력을 보려면 다음을 수행합니다.
   a. https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
   b. 탐색 창에서 로그를 선택합니다.
   c. Lambda 함수에 대한 로그 그룹 이름(`/aws/lambda/function-name`)을 선택합니다.
   d. 로그 스트림 이름을 선택하여 중지한 인스턴스에서 함수를 통해 제공된 데이터를 확인합니다.
7. (선택 사항) 작업이 완료되면 중지된 인스턴스를 종료합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 인스턴스 종료를 참조하십시오.

자습서: EventBridge를 사용하여 자동화된 Amazon EBS 스냅샷 생성 예약

예약된 일정에 따라 EventBridge 규칙을 실행할 수 있습니다. 이 자습서에서는 예약된 일정에 따라 기존 Amazon Elastic Block Store(Amazon EBS) 볼륨에 대한 자동화된 스냅샷을 생성합니다. 고정된 속도를 선택하여 몇 분마다 스냅샷을 생성하거나 Cron 식을 사용하여 특정 시간대에 스냅샷이 생성되도록 지정할 수 있습니다.

> Important
>
> 기본 제공 대상을 통한 규칙 생성은 AWS Management 콘솔에서만 지원됩니다.

1단계: 규칙 생성

일정에 따라 스냅샷을 가져오는 규칙을 생성합니다. Rate 식이나 Cron 식을 사용하여 예약을 지정할 수 있습니다. 자세한 내용은 규칙에 대한 예약 표현식 (p. 32) 단원을 참조하십시오.

규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.
   a. 일정을 선택합니다.
   b. 고정 비율을 선택하고 예약 간격(예: 5분)을 지정합니다. 또는 Cron 표현식을 선택하고 Cron 식을 지정합니다(예: 현재 시간부터 월요일에서 금요일까지 15분마다).
6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 예약된 규칙은 기본 이벤트 버스에만 지원됩니다.
7. 대상에 대해 EC2 CreateSnapshot API call을 선택하십시오.
8. 볼륨 ID에 대상 Amazon EBS 볼륨의 ID를 입력합니다.

9. 이 특정 리소스에 대해 새 역할 생성을 선택합니다. 새로운 역할이 사용자를 대신해 리소스에 액세스 할 대상의 권한을 부여합니다.
10. Create를 선택합니다.

2단계: 규칙 테스트

첫 번째로 가져온 스냅샷을 확인하여 규칙을 확인할 수 있습니다.

규칙을 테스트하려면

1. https://console.aws.amazon.com/ec2/에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Elastic Block Store와 스냅샷을 선택합니다.
3. 목록에 첫 번째 스냅샷이 나타나는지 확인합니다.
4. (선택 사항) 작업이 완료되면 추가 스냅샷을 가져오지 않도록 규칙을 비활성화할 수 있습니다.

    a. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
    b. 탐색 창에서 [Rules]를 선택합니다.
    c. 규칙 옆에 있는 버튼을 선택하고 비활성화를 선택하십시오.
    d. 확인 메시지가 나타나면 Disable을 선택합니다.

자습서: EventBridge를 사용하여 AWS Lambda 함수 예약

예약된 일정에 따라 AWS Lambda 함수를 실행하도록 규칙을 설정할 수 있습니다. 이 자습서에서는 AWS Management 콘솔 또는 AWS CLI를 사용하여 규칙을 생성하는 방법을 보여줍니다. AWS CLI를 사용하고 싶은데 설치가 되지 않은 경우는 AWS Command Line Interface 사용 설명서를 참조하십시오.

EventBridge는 일정 표현식에 초 단위의 정밀성을 제공하지 않습니다. Cron 표현식을 사용해 가장 정밀하게 설정할 수 있는 단위가 1분입니다. EventBridge와 대상 서비스가 분산되어 있기 때문에 예약된 규칙이 트리거 되는 시간과 대상 서비스가 대상 리소스 실행을 인식하는 시간 간에는 몇 초의 지연이 있을 수 있습니다. 예약된 규칙은 지정한 시간(분) 이내에 트리거되지만 정확한 초 단위로 트리거되지는 않습니다.

1단계: AWS Lambda 함수 만들기

Lambda 함수를 생성하여 예약된 이벤트를 기록합니다. 규칙을 생성할 때 이 함수를 지정합니다.

Lambda 함수를 만들려면

1. https://console.aws.amazon.com/lambda/에서 AWS Lambda 콘솔을 엽니다.
2. Lambda를 처음 사용하는 경우 시작 페이지가 표시됩니다. 지금 시작을 선택합니다. 그렇지 않으면 Lambda 함수 생성을 선택합니다.
3. 블루프린트 선택 페이지에서 필터에 `hello`를 입력하고 hello-world 블루프린트를 선택합니다.
4. 트리거 구성 페이지에서 다음을 선택합니다.
5. 함수 구성 페이지에서 다음을 수행합니다.

    a. Lambda 함수의 이름과 설명을 입력합니다. 예를 들어 함수 이름을 `LogScheduledEvent`로 지정합니다.
    b. Lambda 함수에 대한 샘플 코드를 편집합니다. 예:

```
'use strict';
```

```
exports.handler = (event, context, callback) => {
    console.log('LogScheduledEvent');
    console.log('Received event:', JSON.stringify(event, null, 2));
    callback(null, 'Finished');
};
```

c. 역할에서 기존 역할 선택을 선택합니다. 기존 역할의 경우 기본 실행 역할을 선택합니다. 그렇지 않다면 기본 실행 역할을 만듭니다.

d. [Next]를 선택합니다.

6. 검토 페이지에서 함수 생성을 선택합니다.

2단계: 규칙 생성

예약된 일정에 따라 Lambda 함수를 실행하도록 규칙을 생성합니다.

콘솔을 사용하여 규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.

   a. 일정을 선택합니다.

   b. 고정 비율을 선택하고 예약 간격(예: 5분)을 지정합니다.
6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 예약된 규칙은 기본 이벤트 버스에만 지원됩니다.
7. 대상에 대해 Lambda 함수를 선택하십시오.
8. 함수에서 생성한 Lambda 함수를 선택합니다.
9. Create를 선택합니다.

원할 경우, AWS CLI를 사용하여 규칙을 생성할 수 있습니다. 먼저, Lambda 함수를 호출할 수 있는 권한을 규칙에 부여해야 합니다. 그런 다음 규칙을 생성하고 Lambda 함수를 대상으로서 추가할 수 있습니다.

AWS CLI를 사용하여 규칙을 생성하려면

1. 예약된 일정에 따라 자체 트리거되는 규칙을 생성하려면 아래 put-rule 명령을 사용하십시오.

```
aws events put-rule \
--name my-scheduled-rule \
--schedule-expression 'rate(5 minutes)'
```

이 규칙이 트리거되면 이 규칙의 대상에 대한 입력 역할을 하는 이벤트가 생성됩니다. 다음은 이벤트 예제입니다.

```
{
    "version": "0",
    "id": "53dc4d37-cffa-4f76-80c9-8b7d4a4d2eaa",
    "detail-type": "Scheduled Event",
    "source": "aws.events",
    "account": "123456789012",
    "time": "2015-10-08T16:53:06Z",
    "region": "us-east-1",
```

```
    "resources": [
        "arn:aws:events:us-east-1:123456789012:rule/my-scheduled-rule"
    ],
    "detail": {}
}
```

2. EventBridge 서비스 원칙(`events.amazonaws.com`)을 따르고 지정된 Amazon 리소스 이름(ARN)에서 규칙에 대한 권한 범위를 설정하려면 아래 add-permission 명령을 사용하십시오.

```
aws lambda add-permission \
--function-name LogScheduledEvent \
--statement-id my-scheduled-event \
--action 'lambda:InvokeFunction' \
--principal events.amazonaws.com \
--source-arn arn:aws:events:us-east-1:123456789012:rule/my-scheduled-rule
```

3. 이 규칙에 생성한 Lambda 함수를 추가하여 5분마다 실행이 되도록 하려면 아래 put-targets 명령을 사용하십시오.

```
aws events put-targets --rule my-scheduled-rule --targets file://targets.json
```

다음 콘텐츠가 포함된 `targets.json` 파일을 생성합니다.

```
[
  {
    "Id": "1",
    "Arn": "arn:aws:lambda:us-east-1:123456789012:function:LogScheduledEvent"
  }
]
```

3단계: 규칙 확인

2단계 완료 후 최소 5분 후 Lambda 함수가 호출되었는지 확인할 수 있습니다.

규칙을 테스트하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.

   생성한 규칙의 이름을 선택한 후 규칙에 대한 지표를 선택합니다.
3. Lambda 함수에서 출력을 보려면 다음을 수행합니다.

   a. https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
   b. 탐색 창에서 로그를 선택합니다.
   c. Lambda 함수에 대한 로그 그룹 이름(`/aws/lambda/function-name`)을 선택합니다.
   d. 로그 스트림 이름을 선택하여 시작한 인스턴스에서 함수를 통해 제공된 데이터를 확인합니다.

자습서: AWS 시스템 관리자 자동화를 EventBridge 대상으로 설정

EventBridge를 사용해 정기적으로 또는 지정된 이벤트가 감지될 때마다 AWS 시스템 관리자 자동화를 호출할 수 있습니다. 이 자습서에서는 특정 이벤트를 기준으로 시스템 관리자 자동화를 호출하는 것으로 가정합니다.

EventBridge 규칙을 만들려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.
   a. 이벤트 패턴을 선택합니다.
   b. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
   c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
   d. 서비스 이름 및 이벤트 유형에 대해 트리거로 사용할 서비스 및 이벤트 유형을 선택합니다. 선택한 서비스 및 이벤트 유형에 따라 추가 옵션을 지정해야 할 수도 있습니다.
6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에 대해 SSM Automation을 선택하십시오.
8. 문서에서 대상이 트리거될 때 실행할 시스템 관리자 문서를 선택합니다.
9. (선택 사항) 특정 버전의 문서를 지정하려면 Configure document version(문서 버전 구성)을 선택합니다.
10. Configure automation parameter(s) 아래에서 파라미터 없음 또는 상수를 선택합니다.

    상수를 선택할 경우 문서 실행에 전달할 상수를 지정합니다.
11. EventBridge는 이벤트 실행에 필요한 IAM 역할을 생성할 수 있습니다.
    - IAM 역할을 자동으로 생성하려면 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
    - 이전에 생성한 IAM 역할을 사용하려면 기존 역할 사용을 선택합니다.
12. Create를 선택합니다.

자습서: EventBridge를 사용하여 Amazon Kinesis 스트림으로 이벤트 릴레이

EventBridge의 AWS AWS API 호출 이벤트를 Amazon Kinesis의 스트림으로 릴레이할 수 있습니다.

사전 조건

AWS CLI를 설치합니다. 자세한 내용은 AWS Command Line Interface 사용 설명서 단원을 참조하십시오.

1단계: Amazon Kinesis 스트림 생성

스트림을 생성하려면 아래 create-stream 명령을 사용하십시오.

```
aws kinesis create-stream --stream-name test --shard-count 1
```

스트림 상태가 ACTIVE이면 스트림이 준비되었다는 뜻입니다. 스트림 상태를 확인하려면 아래 describe-stream 명령을 사용하십시오.

```
aws kinesis describe-stream --stream-name test
```

2단계: 규칙 생성

한 예로 Amazon EC2 인스턴스를 중지할 때 스트림으로 이벤트를 전송하는 규칙을 생성합니다.

규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.
   a. [Event pattern]을 선택합니다.
   b. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
   c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
   d. 서비스 이름에서 EC2를 선택합니다.
   e. 이벤트 유형에서 인스턴스 상태 변경 알림을 선택합니다.
   f. 특정 상태, 실행을 선택합니다.
6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에서 Kinesis 스트림을 선택합니다.
8. 스트림에서 생성한 스트림을 선택합니다.
9. 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
10. Create를 선택합니다.

3단계: 규칙 테스트

규칙을 테스트하려면 Amazon EC2 인스턴스를 중지합니다. 인스턴스가 중지되도록 몇 분 기다린 다음 CloudWatch 지표를 통해 함수가 호출되었는지 확인합니다.

인스턴스를 중지시켜 규칙을 테스트하려면

1. https://console.aws.amazon.com/ec2/에서 Amazon EC2 콘솔을 엽니다.
2. 인스턴스 시작. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 인스턴스 시작을 참조하십시오.
3. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
4. 탐색 창에서 [Rules]를 선택합니다.

   생성한 규칙의 이름을 선택한 후 규칙에 대한 지표를 선택합니다.
5. (선택 사항) 작업이 완료되면 인스턴스를 종료합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 인스턴스 종료를 참조하십시오.

4단계: 이벤트가 릴레이되었는지 확인

스트림에서 레코드를 가져와서 이벤트가 릴레이되었는지 확인할 수 있습니다.

레코드를 가져오려면

1. Kinesis 스트림에서 읽기를 시작하려면 아래 get-shard-iterator 명령을 사용하십시오.

```
aws kinesis get-shard-iterator --shard-id shardId-000000000000 --shard-iterator-type
 TRIM_HORIZON --stream-name test
```

다음은 예제 출력입니다.

```
{
    "ShardIterator": "AAAAAAAAAAHSywljv0zEgPX4NyKdZ5wryMzP9yALs8NeKbUjp1IxtZs1Sp
+KEd9I6AJ9ZG4lNR1EMi+9Md/nHvtLyxpfhEzYvkTZ4D9DQVz/mBYWRO6OTZRKnW9gd
+efGN2aHFdkH1rJl4BL9Wyrk+ghYG22D2T1Da2EyNSH1+LAbK33gQweTJADBdyMwlo5r6PqcP2dzhg="
}
```

2. 레코드를 얻으려면 다음 `get-records` 명령을 사용합니다. 샤드 반복자는 이전 단계에서 얻은 것입니다.

```
aws kinesis get-records --shard-
iterator AAAAAAAAAAHSywljv0zEgPX4NyKdZ5wryMzP9yALs8NeKbUjp1IxtZs1Sp+KEd9I6AJ9ZG4lNR1EMi
+9Md/nHvtLyxpfhEzYvkTZ4D9DQVz/mBYWRO6OTZRKnW9gd+efGN2aHFdkH1rJl4BL9Wyrk
+ghYG22D2T1Da2EyNSH1+LAbK33gQweTJADBdyMwlo5r6PqcP2dzhg=
```

이 명령이 성공하면 지정된 샤드에서 스트림에서 나온 레코드를 요청합니다. 0개 이상의 레코드를 수신할 수 있습니다. 반환된 레코드가 스트림의 모든 레코드를 나타내는 것은 아닙니다. 기대한 데이터를 수신하지 못한 경우에는 계속해서 `get-records`를 호출합니다.

Kinesis의 레코드는 Base64로 인코딩됩니다. 그러나 AWS CLI에서 지원되는 스트림은 Base64 디코딩을 제공하지 않습니다. Base64 디코더를 사용하여 데이터를 수동으로 디코딩하면 이벤트가 JSON 형식으로 스트림에 릴레이되었다는 것을 알 수 있습니다.

자습서: 파일이 Amazon S3 버킷에 업로드되면 Amazon ECS 작업 실행

EventBridge를 사용하여 특정 AWS 이벤트가 발생할 때 Amazon ECS 작업을 실행할 수 있습니다. 이 자습서에서는 Amazon S3 PUT 작업을 사용하여 파일이 특정 Amazon S3 버킷에 업로드될 때마다 Amazon ECS 작업을 실행하는 EventBridge 규칙을 설정합니다.

이 자습서에서는 Amazon ECS에 작업 정의를 이미 만들었다고 가정합니다.

PUT 작업을 사용하여 파일이 S3 버킷에 업로드될 때마다 Amazon ECS 작업을 실행하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.
   a. [Event pattern]을 선택합니다.
   b. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
   c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
   d. 서비스 이름에서 Simple Storage Service(S3)를 선택합니다.
   e. 이벤트 유형에서 객체 수준 작업을 선택합니다.
   f. 특정 작업, Put Object를 선택합니다.
   g. 이름 기준 특정 버킷을 선택하고 버킷 이름을 입력합니다.

6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에서 다음을 수행합니다.

   a. ECS 작업을 선택하십시오.
   b. 클러스터 및 작업 정의에서 생성한 리소스를 선택합니다.
   c. 시작 유형에서 `FARGATE` 또는 `EC2`를 선택합니다. `FARGATE`는 AWS Fargate가 지원되는 리전에만 표시됩니다.
   d. (선택 사항) 작업 그룹에 값을 지정합니다. 시작 유형이 `FARGATE`인 경우, 필요에 따라 플랫폼 버전을 지정합니다. 플랫폼 버전의 숫자 부분만 지정합니다(예: 1.1.0).
   e. (선택 사항) 작업 정의 개정 및 작업 수를 지정합니다. 작업 정의 개정을 지정하지 않은 경우 최신이 사용됩니다.
   f. 작업 정의가 awsvpc 네트워크 모드를 사용하는 경우 서브넷과 보안 그룹을 지정해야 합니다. 모든 서브넷과 보안 그룹이 동일한 VPC에 있어야 합니다.

      보안 그룹 또는 서브넷을 둘 이상 지정하는 경우, 공백이 아닌 쉼표로 구분하십시오.

      서브넷에 다음 예와 같이 각 서브넷의 전체 `subnet-id` 값을 지정합니다.

      ```
      subnet-123abcd,subnet-789abcd
      ```

   g. 퍼블릭 IP 주소가 자동 할당되도록 할지 여부를 선택합니다.
   h. EventBridge는 작업 실행에 필요한 IAM 역할을 생성할 수 있습니다.

      - IAM 역할을 자동으로 생성하려면 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
      - 이전에 생성한 IAM 역할을 사용하려면 기존 역할 사용을 선택합니다. 이 역할은 빌드를 호출할 수 있는 충분한 권한을 이미 가지고 있는 역할이어야 합니다. EventBridge는 선택한 역할에 추가 권한을 부여하지 않습니다.
8. Create를 선택합니다.

자습서: AWS CodeBuild를 사용하여 자동화된 빌드 예약

이 자습서의 예에서는 CodeBuild가 매일 밤 20:00(GMT)시에 빌드를 실행하도록 예약합니다. 또한 이 예약된 빌드에 사용할 상수를 CodeBuild에 전달합니다.

매일 밤 20:00(GMT)시에 CodeBuild 프로젝트 빌드를 예약하는 규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.

   a. 일정을 선택합니다.
   b. Cron 식을 선택하고 표현식으로 다음을 지정합니다. **`0 20 ? * MON-FRI *`**.(예: 5분)
6. Select event bus(이벤트 버스 선택)에 대해 AWS default event bus(AWS 기본 이벤트 버스)를 선택하십시오. 계정의 AWS 서비스가 이벤트를 출력하면 항상 계정의 기본 이벤트 버스로 이동합니다.
7. 대상에서 CodeBuild project(CodeBuild 프로젝트)를 선택합니다.
8. Project ARN(프로젝트 ARN)에 빌드 프로젝트의 ARN을 입력합니다.

9. 이 자습서에서는 CodeBuild에 기본값을 재정의하는 파라미터를 전달하는 선택적 단계를 추가합니다. CodeBuild를 대상으로 설정하는 경우에는 이 단계가 필요하지 않습니다. 파라미터를 전달하기 위해 입력 구성, 상수(JSON 텍스트)를 선택합니다.

   상수(JSON 텍스트) 아래 상자에, { "timeoutInMinutesOverride": 30 }을 입력하여 이렇게 예약된 빌드의 제한 시간이 30분으로 재정의되도록 설정합니다.

   전달할 수 있는 파라미터에 대한 자세한 내용은 AWS CodeBuild API 레퍼런스의 StartBuild 섹션을 참조하십시오. 이 필드의 `projectName` 파라미터는 전달할 수 없습니다. 대신, Project ARN(프로젝트 ARN)의 ARN을 사용하여 프로젝트를 지정할 수 있습니다.
10. EventBridge는 빌드 프로젝트 실행에 필요한 IAM 역할을 생성할 수 있습니다.

    - IAM 역할을 자동으로 생성하려면 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
    - 이전에 생성한 IAM 역할을 사용하려면 기존 역할 사용을 선택합니다. 이 역할은 빌드를 호출할 수 있는 충분한 권한을 이미 가지고 있는 역할이어야 합니다. EventBridge는 선택한 역할에 추가 권한을 부여하지 않습니다.
11. Create를 선택합니다.

자습서: Amazon EC2 인스턴스의 로그 상태 변화

이 자습서의 예에서는 Amazon EC2의 상태 변경 알림이 Amazon CloudWatch Logs에 기록되도록 하는 규칙을 생성합니다.

Amazon EC2 상태 변경 알림이 CloudWatch Logs에 기록되도록 하는 규칙을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 [Rules]를 선택합니다.
3. [Create rule]을 선택합니다.
4. 규칙에 대해 이름과 설명을 입력하십시오.
5. Define pattern(패턴 정의)에 대해 다음을 수행하십시오.

   a. [Event pattern]을 선택합니다.
   b. Pre-defined pattern by service(서비스별 사전 정의된 패턴)을 선택하십시오.
   c. Service provider(서비스 제공업체)에 대해 AWS를 선택하십시오.
   d. 서비스 이름에서 EC2를 선택합니다.
   e. 이벤트 유형에서 EC2 인스턴스 상태 변경 알림을 선택합니다.
   f. 모든 상태와 모든 인스턴스를 선택합니다.
6. 대상에서 CloudWatch 로그 그룹을 선택하십시오.
7. 로그 그룹에서 상태 변경 알림을 수신할 로그 그룹의 이름을 입력합니다.
8. Create를 선택합니다.

자습서: EventBridge 스키마 레지스트리를 사용하여 이벤트용 코드 바인딩 다운로드

이벤트 스키마에 대한 코드 바인딩을 생성하여 Java, Python 및 TypeScript 개발 속도를 높일 수 있습니다. 기존 AWS 서비스, 생성한 스키마 및 이벤트 버스의 이벤트를 기반으로 생성하는 스키마에 대한 코드 바인딩을 가져올 수 있습니다. EventBridge 콘솔, EventBridge 스키마 레지스트리 API를 사용하여 스키마에 대한 코드 바인딩을 생성하고 AWS Toolkit를 사용하여 IDE에서 직접 생성할 수 있습니다.

이 자습서에서는 AWS 서비스 이벤트에 대한 EventBridge 스키마에서 코드 바인딩을 생성하고 다운로드합니다.

EventBridge 스키마에서 코드 바인딩을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 Schemas(스키마)를 선택합니다.
3. AWS event schema registry(AWS 이벤트 스키마 레지스트리) 탭을 선택합니다.
4. 스키마 레지스트리를 탐색하거나 스키마를 검색하여 코드 바인딩을 원하는 AWS 서비스에 대한 스키마를 찾습니다.
5. 스키마 이름을 선택하여 Schema details(스키마 세부 정보) 페이지를 표시합니다.
6. Version(버전) 섹션에서 Download code bindings(코드 바인딩 다운로드)를 선택합니다.
7. Download code bindings(코드 바인딩 다운로드) 페이지에서 다운로드할 코드 바인딩의 언어를 선택합니다.
8. Download(다운로드)를 선택합니다.

   다운로드가 시작되는 데 몇 초가 걸릴 수 있습니다. 다운로드 파일은 선택한 언어에 대한 코드 바인딩의 zip 파일입니다.

자체 코드에서 이러한 코드 바인딩을 사용하면 이 EventBridge 이벤트를 사용하여 애플리케이션을 빠르게 빌드할 수 있습니다.

규칙에 대한 예약 표현식

Cron 또는 Rate 식을 사용하여 EventBridge에서 자동 일정에 따라 자체 트리거되는 규칙을 생성할 수 있습니다. 예약된 모든 이벤트는 UTC 시간대를 사용하며 예약의 최소 단위는 1분입니다.

EventBridge는 cron 표현식과 rate 표현식을 지원합니다. rate 표현식은 정의하기가 더 간단하지만 cron 표현식이 지원하는 세분화된 일정 관리는 제공하지 않습니다. 예를 들어, cron 표현식을 사용하여 매주 또는 매월 특정 요일의 지정된 시간에 트리거되는 규칙을 정의할 수 있습니다. 반대로 rate 표현식은 매 시간 한 번 또는 매일 한 번과 같이 정기적으로 규칙을 트리거합니다.

> **Note**
>
> EventBridge는 일정 표현식에 초 단위의 정밀성을 제공하지 않습니다. Cron 표현식을 사용해 가장 정밀하게 설정할 수 있는 단위가 1분입니다. EventBridge와 대상 서비스가 분산되어 있기 때문에 예약된 규칙이 트리거 되는 시간과 대상 서비스가 대상 리소스 실행을 인식하는 시간 간에는 몇 초의 지연이 있을 수 있습니다. 예약된 규칙은 지정한 시간(분) 이내에 트리거되지만 정확한 초 단위로 트리거되지는 않습니다.

형식

- cron 표현식 (p. 32)
- rate 표현식 (p. 34)

cron 표현식

cron 표현식에는 각각 공백으로 구분되는 필수 필드 6개가 있습니다.

구문

```
cron(fields)
```

필드	값	와일드카드
분	0~59	, - * /
시간	0~23	, - * /
일	1~31	, - * ? / L W
월	1-12 또는 JAN-DEC	, - * /
요일	1-7 또는 SUN-SAT	, - * ? L #
연도	1970~2199	, - * /

와일드카드

- ,(쉼표) 와일드카드에는 추가 값이 포함되어 있습니다. 예를 들어, '월' 필드에서 JAN, FEB, MAR은 1월, 2월, 3월을 포함한다는 의미입니다.

- -(대시) 와일드카드로 범위를 지정할 수 있습니다. 예를 들어, '일' 필드에서 1-15는 지정된 달의 1일에서 15일까지 포함한다는 의미입니다.
- *(별표) 와일드카드로 필드에 모든 값을 포함할 수 있습니다. '시간' 필드에서 *는 모든 시간을 포함한다는 의미입니다. '일' 및 '요일' 필드 모두에서 *를 사용할 수 없습니다. 필드 중 하나에 사용할 경우 다른 하나에는 반드시 ?를 사용해야 합니다.
- /(슬래시) 와일드카드로 증분을 지정할 수 있습니다. 예를 들어, '분' 필드에 1/10을 입력하면 지정한 시간의 1분부터 시작해서 매 10분 간격을 지정할 수 있습니다(즉, 11분, 21분, 31분 등).
- ?(물음표) 와일드카드로 어떤 한 가지나 다른 것을 지정할 수 있습니다. '일' 필드에 7을 입력하고 '요일' 필드에는 ?을 입력하면 매월 7일이 무슨 요일이든 상관없이 7번째 되는 날을 지정한다는 의미입니다.
- '일' 또는 '요일' 필드에서 L 와일드카드로 해당 월 또는 주의 마지막 날을 지정할 수 있습니다.
- '일' 필드에서는 **W** 와일드카드로 어떤 한 평일을 지정할 수 있습니다. 예를 들어 '일' 필드에 **3W**를 입력하면 해당 월의 세 번째 평일에 가장 가까운 날을 지정할 수 있습니다.
- '요일' 필드의 # 와일드카드는 그 달에 속한 정해진 요일의 특정 인스턴스를 지정합니다. 예를 들어, 3#2는 그 달의 두 번째 화요일입니다. 3은 각 주의 셋째 날이므로 화요일을 나타내고 2는 그 달의 두 번째 해당 요일입니다.

제한 사항

- 같은 cron 표현식에서 '일' 및 '요일' 필드를 지정할 수 없습니다. 이들 필드 중 하나에 값(또는 *)을 지정하는 경우에는 다른 필드에서 반드시 ?(물음표)를 사용해야 합니다.
- 1분보다 빠른 속도로 이어지는 cron 표현식은 지원되지 않습니다.

예제

예약에 따라 규칙을 생성할 때는 다음과 같이 동일한 cron 문자열을 사용할 수 있습니다.

분	시간	일	월	요일	연도	의미
0	10	*	*	?	*	매일 오전 10시(UTC)에 실행
15	12	*	*	?	*	매일 오후 12시 15분(UTC)에 실행
0	18	?	*	월-금	*	매주 월요일부터 금요일까지 오후 6시(UTC)에 실행
0	8	1	*	?	*	매월 1일 오전 8시(UTC)에 실행
0/15	*	*	*	?	*	15분마다 실행
0/10	*	?	*	월-금	*	월요일부터 금요일까지 10분마다 실행

분	시간	일	월	요일	연도	의미
0/5	8~17	?	*	월-금	*	월요일부터 금요일까지 오전 8시부터 오후 5시 55분(UTC) 사이에 5분마다 실행

다음 예제는 AWS CLI의 put-rule 명령에서 cron 표현식을 사용하는 방법을 보여줍니다. 첫 번째 예제는 매일 12:00pm UTC에 트리거되는 규칙을 생성합니다.

```
aws events put-rule --schedule-expression "cron(0 12 * * ? *)" --name MyRule1
```

다음 예제는 매일 2:00pm UTC 이후 5분 및 35분에 트리거되는 규칙을 생성합니다.

```
aws events put-rule --schedule-expression "cron(5,35 14 * * ? *)" --name MyRule2
```

두 번째 예제는 2019년부터 2022년까지 매달 마지막 금요일 10:15am UTC에 트리거되는 규칙을 생성합니다.

```
aws events put-rule --schedule-expression "cron(15 10 ? * 6L 2019-2022)" --name MyRule3
```

rate 표현식

rate 표현식은 예약된 이벤트 규칙을 생성할 때 시작되며, 정의된 예약 일정에 따라 실행됩니다.

rate 표현식에는 필수 필드가 2개 있습니다. 각 필드는 공백으로 구분됩니다.

구문

```
rate(value unit)
```

USD 상당

양수.

unit

시간 단위. 1의 값(예: `minute`)과 1을 초과하는 값(예: `minutes`)은 서로 다른 단위가 필요합니다.

유효값: 분 | 분 | 시간 | 시간 | 일 | 일

제한 사항

값이 1과 같을 경우에는 단위가 단수여야 합니다. 마찬가지로, 1보다 큰 값에 대해서는 단위가 복수여야 합니다. 예를 들어, rate(1 hours)와 rate(5 hour)는 잘못된 식이며, rate(1 hour)와 rate(5 hours)가 유효한 식입니다.

예제

다음 예제는 AWS CLI의 put-rule 명령에서 rate 표현식을 사용하는 방법을 보여줍니다. 첫 번째 예제는 1분마다 규칙을 트리거하고, 두 번째 예제는 5분마다 규칙을 트리거하며, 다음은 한 시간에 한 번 트리거하고, 세 번째 예제는 하루에 한 번 트리거합니다.

```
aws events put-rule --schedule-expression "rate(1 minute)" --name MyRule2
```

```
aws events put-rule --schedule-expression "rate(5 minutes)" --name MyRule3
```

```
aws events put-rule --schedule-expression "rate(1 hour)" --name MyRule4
```

```
aws events put-rule --schedule-expression "rate(1 day)" --name MyRule5
```

EventBridge의 이벤트 및 이벤트 패턴

Amazon EventBridge의 이벤트는 JSON 객체로 표현됩니다. 모든 이벤트는 구조가 비슷하며 최상위 필드가 동일합니다.

EventBridge 규칙은 이벤트 패턴을 사용하여 이벤트 버스의 AWS 이벤트와 일치시킵니다. 패턴이 일치하면 규칙은 해당 이벤트를 대상으로 라우팅합니다.

주제

AWS 이벤트

다음은 Amazon EventBridge의 이벤트 예제입니다.

```
{
  "version": "0",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "EC2 Instance State-change Notification",
  "source": "aws.ec2",
  "account": "111122223333",
  "time": "2017-12-22T18:43:48Z",
  "region": "us-west-1",
  "resources": [
    "arn:aws:ec2:us-west-1:123456789012:instance/ i-1234567890abcdef0"
  ],
  "detail": {
    "instance-id": " i-1234567890abcdef0",
    "state": "terminated"
  }
}
```

다음과 같이 이벤트에 대한 세부 정보를 기억하는 것이 중요합니다.

- 이들은 모두 동일한 최상위 수준 필드를 가지고 있으며, 위의 예제에서도 표시되었듯이 반드시 있어야 하는 필드들입니다.
- detail 최상위 필드의 콘텐츠는 이벤트를 생성한 서비스와 이벤트에 따라 달라집니다. source 필드와 detail-type 필드를 조합하여 detail 필드에 나타나는 필드와 값을 식별할 수 있습니다. AWS 서비스에서 생성되는 이벤트의 예제는 EventBridge 지원되는 AWS 서비스의 이벤트 예제 (p. 57) 단원을 참조하십시오.

다음은 각 이벤트 필드에 대한 설명입니다.

version

기본적으로 이 필드는 모든 이벤트에서 0으로 설정되어 있습니다.

id

모든 이벤트에 대해 고유 값이 생성됩니다. 규칙을 통해 대상으로 이동되어 처리되는 이벤트를 추적하는 데 도움이 될 수 있습니다.

detail-type

source 필드를 함께 사용하여 detail 필드에 나타나는 필드와 값을 식별합니다.

CloudTrail을 통해 전달되는 모든 이벤트는 `detail-type`의 값으로 `AWS API Call via CloudTrail`을 보유합니다. 자세한 내용은 CloudTrail을 통해 전달된 이벤트 (p. 102) 단원을 참조하십시오.

소스

이벤트 소스인 서비스를 식별합니다. AWS에서 발생한 모든 이벤트는 "aws"로 시작됩니다. 고객이 생성한 이벤트는 "aws"로 시작되지 않는 한 어떤 값이라도 가질 수 있습니다. Java 패키지 이름 스타일을 사용하여 도메인 이름 문자열을 역순으로 만드는 것이 좋습니다.

AWS 서비스의 `source`에 대한 올바른 값을 찾으려면 AWS 서비스 네임스페이스의 표를 참조하십시오. 예를 들어 Amazon CloudFront에 대한 `source` 값은 `aws.cloudfront`입니다.

계정

AWS 계정을 식별하는 12자리 숫자입니다.

시간

이벤트를 호출한 서비스에서 지정할 수 있는 이벤트 타임스탬프입니다. 이벤트가 시간 간격 내에 있으면 서비스는 시작 시간을 보고해서 이 값이 이벤트가 실제 수신되는 시간보다 훨씬 앞에 있도록 할 수 있습니다.

region

이벤트를 호출한 AWS 리전을 식별합니다.

리소스

이 JSON 어레이에는 이벤트에서 호출된 리소스를 식별하는 ARN이 포함되어 있습니다. ARN 포함 여부는 서비스의 단독 재량에 따라 결정됩니다. 예를 들어 Amazon EC2 인스턴스 상태 변경에는 Amazon EC2 인스턴스 ARN이 포함되어 있고, Auto Scaling 이벤트에는 인스턴스 및 Auto Scaling 그룹 모두에 대한 ARN이 포함되어 있지만, AWS CloudTrail에서의 API 호출에는 리소스 ARN이 포함되지 않습니다.

detail

이벤트를 호출한 서비스의 단독 재량에 따라 콘텐츠가 결정되는 JSON 개체입니다. 위 예제의 세부 콘텐츠는 필드가 단 두 개로 매우 간단합니다. AWS API 호출 이벤트는 약 50개의 필드가 몇 가지 수준으로 중첩된 세부 개체를 가지고 있습니다.

AWS 서비스에서 사용할 수 있는 모든 이벤트 유형 목록을 보려면 EventBridge 지원되는 AWS 서비스의 이벤트 예제 (p. 57) 단원을 참조하십시오.

이벤트 패턴

이벤트 패턴은 일치하는 이벤트와 동일한 구조를 갖습니다. 그들은 필터링하는 이벤트와 거의 비슷하게 보입니다. 규칙은 이벤트 패턴을 사용하여 이벤트를 선택하고 대상으로 이를 라우팅합니다. 패턴은 이벤트와 일치할 수도 있고 아닐 수도 있습니다. 다음은 EventBridge에서 발생할 수 있는 간단한 AWS 이벤트의 예입니다.

```
{
  "version": "0",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "EC2 Instance State-change Notification",
  "source": "aws.ec2",
  "account": "111122223333",
  "time": "2017-12-22T18:43:48Z",
  "region": "us-west-1",
```

```
  "resources": [
    "arn:aws:ec2:us-west-1:123456789012:instance/ i-1234567890abcdef0"
],
"detail": {
  "instance-id": " i-1234567890abcdef0",
  "state": "terminated"
  }
}
```

이벤트 패턴은 일치하는 이벤트와 동일한 구조를 갖습니다. 예를 들어 다음 이벤트 패턴을 사용하면 Amazon EC2의 이벤트만 구독할 수 있습니다.

```
{
  "source": [ "aws.ec2" ]
}
```

패턴은 단순히 일치시키려는 필드를 인용하고 찾고 있는 값을 제공합니다.

위의 샘플 이벤트는 대부분의 이벤트와 마찬가지로 중첩된 구조를 가지고 있습니다. 모든 `instance-termination` 이벤트를 처리한다고 가정합니다. 다음과 같은 이벤트 패턴을 만듭니다.

```
{
  "source": [ "aws.ec2" ],
  "detail-type": [ "EC2 Instance State-change Notification" ],
  "detail": {
    "state": [ "terminated" ]
  }
}
```

일치시킬 필드 지정

원하는 필드만 지정합니다. 앞의 예에서는 세 개의 필드, 즉 "`source`" 및 "`detail-type`" 최상위 필드, "`detail`" 객체 필드 내 "`state`" 필드에 대한 값만 제공합니다. EventBridge는 필터를 적용하는 동안 이벤트의 다른 모든 필드를 무시합니다.

일치 값

일치 값은 항상 어레이에 있습니다. 일치시킬 값은 "[" and "]"로 둘러싸인 JSON 어레이에 있습니다. 이는 여러 값을 제공할 수 있도록 하기 위한 것입니다. 예를 들어 Amazon EC2 또는 Fargate의 이벤트에 관심이 있는 경우 다음을 지정할 수 있습니다.

```
{
    "source": [ "aws.ec2", "aws.fargate" ]
}
```

`"source"` 필드의 값이 `"aws.ec2"` 또는 `"aws.fargate"`인 이벤트와 일치합니다.

모든 JSON 데이터 유형 일치

모든 JSON 데이터 유형을 일치시킬 수 있습니다. 다음 예제 Amazon EC2 Auto Scaling 이벤트를 살펴보십시오.

```
{
  "version": "0",
  "id": "3e3c153a-8339-4e30-8c35-687ebef853fe",
  "detail-type": "EC2 Instance Launch Successful",
```

```
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "2015-11-11T21:31:47Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "eventVersion": "",
    "responseElements": null
  }
}
```

위 예제의 경우 다음과 같이 "responseElements" 필드에 일치시킬 수 있습니다.

```
{
  "source": [ "aws.autoscaling" ],
  "detail-type": [ "EC2 Instance Launch Successful" ],
  "detail": {
   "responseElements": [ null ]
  }
}
```

이 작업은 숫자에서도 작동합니다. 다음 Amazon Macie 이벤트를 살펴보십시오(간결함을 위해 잘림).

```
{
  "version": "0",
  "id": "3e355723-fca9-4de3-9fd7-154c289d6b59",
  "detail-type": "Macie Alert",
  "source": "aws.macie",
  "account": "123456789012",
  "time": "2017-04-24T22:28:49Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id/alert/alert_id",
    "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id"
  ],
  "detail": {
  "notification-type": "ALERT_CREATED",
  "name": "Scanning bucket policies",
  "tags": [
  "Custom_Alert",
  "Insider"
  ],
  "url": "https://lb00.us-east-1.macie.aws.amazon.com/111122223333/posts/alert_id",
  "alert-arn": "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id/alert/alert_
  "risk-score": 80,
  "trigger": {
    "rule-arn": "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id",
    "alert-type": "basic",
    "created-at": "2017-01-02 19:54:00.644000",
    "description": "Alerting on failed enumeration of large number of bucket policie
    "risk": 8
},
"created-at": "2017-04-18T00:21:12.059000",
.
.
.
```

위험 점수가 80이고 트리거 위험이 8인 항목을 일치시키려면 다음을 수행하십시오.

```
{
  "source": ["aws.macie"],
  "detail-type": ["Macie Alert"],
```

```
  "detail": {
    "risk-score": [80],
    "trigger": {
      "risk": [8]
    }
  }
}
```

이벤트 패턴과의 단순 일치

다음 예제 이벤트는 후속 이벤트 패턴이 이 이벤트 JSON과 일치하는 방법을 표시하는 데 사용됩니다.

```
{
  "version": "0",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "EC2 Instance State-change Notification",
  "source": "aws.ec2",
  "account": "111122223333",
  "time": "2017-12-22T18:43:48Z",
  "region": "us-west-1",
  "resources": [
    "arn:aws:ec2:us-west-1:123456789012:instance/ i-1234567890abcdef0"
  ],
  "detail": {
    "instance-id": " i-1234567890abcdef0",
    "state": "terminated"
  }
}
```

이벤트 패턴은 이벤트와 비슷한 구조를 가진 JSON 객체로 표현이 됩니다.

```
{
  "source": [ "aws.ec2" ],
  "detail-type": [ "EC2 Instance State-change Notification" ],
  "detail": {
    "state": [ "running" ]
  }
}
```

이 이벤트 패턴은 `"state"` 필드의 값이 `"running"`에서 일치하므로 예제 이벤트에서 일치하지 않지만 예제 이벤트의 값은 `"terminated"`입니다.

이벤트 패턴 일치에 대해 다음 사항을 기억해야 합니다.

- 이벤트와 패턴을 일치시키려면 이벤트에 패턴에 나열된 모든 필드 이름이 포함되어 있어야 합니다. 필드 이름은 같은 중첩 구조를 가진 이벤트에 나타나야 합니다.
- 패턴에서 언급되지 않은 이벤트의 다른 필드들은 무시가 되는데, 언급되지 않은 필드를 위한 "*": "*" 와일드카드가 있어서 효과적입니다.
- 대문자 변환이나 기타 문자열 정규화가 없을 경우에는 정확하게 일치합니다(문자별로).
- 일치되는 값들은 JSON 규칙을 따릅니다. 즉, 따옴표로 묶인 문자열, 숫자 및 따옴표로 묶지 않은 키워드(`true`, `false` 및 `null`)가 값이 될 수 있습니다.
- 일치하는 숫자는 문자열 표현 수준에 있습니다. 예를 들어 300, 300.0 및 3.0e2는 동일한 것으로 간주되지 않습니다.

이벤트에 일치하는 패턴을 쓸 때 `TestEventPattern` API 또는 `test-event-pattern` CLI 명령을 사용하여 패턴이 원하는 이벤트와 일치하게 할 수 있습니다. 자세한 내용은 TestEventPattern 또는 test-event-pattern을 참조하십시오.

다음 이벤트 패턴은 이전 예제 이벤트와 일치합니다. 패턴에 지정된 인스턴스 값들 중 하나가 이벤트와 일치하고 패턴이 이벤트에 포함되지 않은 어떤 추가 필드도 지정하지 않는다는 점에서 첫 번째 패턴이 일치합니다. "terminated" 상태가 이벤트에 포함되어 있다는 점에서 두 번째 패턴도 일치합니다.

```
{
  "resources": [
    "arn:aws:ec2:us-east-1:123456789012:instance/i-12345678",
    "arn:aws:ec2:us-east-1:123456789012:instance/i-abcdefgh"
  ]
}
```

```
{
  "detail": {
    "state": [ "terminated" ]
  }
}
```

이들 이벤트 패턴은 이 페이지 맨 위에 있는 이벤트와 일치하지 않습니다. 패턴이 상태에 대해 "pending" 값을 지정하고 이 값이 이벤트에 나타나지 않는다는 점에서 첫 번째 패턴은 일치하지 않습니다. 패턴에 지정된 리소스 값이 이벤트에 나타나지 않는다는 점에서 두 번째 패턴도 일치하지 않습니다.

```
{
  "source": [ "aws.ec2" ],
  "detail-type": [ "EC2 Instance State-change Notification" ],
  "detail": {
    "state": [ "pending" ]
  }
}
```

```
{
  "source": [ "aws.ec2" ],
  "detail-type": [ "EC2 Instance State-change Notification" ],
  "resources": [ "arn:aws:ec2:us-east-1::image/ami-12345678" ]
}
```

EventBridge 이벤트 패턴에서 Null 값과 빈 문자열 일치

null 값 또는 빈 문자열을 갖는 이벤트 필드와 일치하는 패턴을 생성할 수 있습니다. 이것이 어떻게 작동하는지 보려면 다음 예제 이벤트를 살펴보십시오.

```
{
  "version": "0",
  "id": "3e3c153a-8339-4e30-8c35-687ebef853fe",
  "detail-type": "EC2 Instance Launch Successful",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "2015-11-11T21:31:47Z",
  "region": "us-east-1",
  "resources": [
   ],
  "detail": {
    "eventVersion": "",
    "responseElements": null
   }
```

```
}
```

`eventVersion`의 값이 빈 문자열인 이벤트들을 일치시키려면 이 이벤트 예제와 일치할 다음 패턴을 사용합니다.

```
{
  "detail": {
     "eventVersion": [""]
  }
}
```

`responseElements`의 값이 null인 이벤트들을 일치시키려면 이 이벤트 예제와 일치할 다음 패턴을 사용합니다.

```
{
  "detail": {
     "responseElements": [null]
  }
}
```

Null 값과 빈 문자열은 패턴 일치 시 서로 바꾸어 사용할 수 없습니다. 빈 문자열을 감지하기 위해 쓴 패턴은 `null` 값을 포착하지 못합니다.

EventBridge 이벤트 패턴 내 어레이

패턴의 각 필드의 값은 하나 이상의 값을 포함하는 어레이로서, 어레이 값 중 어떤 것이라도 이벤트의 값과 일치하면 패턴이 일치됩니다. 이벤트의 값이 어레이일 경우에는 패턴 어레이와 이벤트 어레이에서 교차되는 부분이 없는 경우에만 패턴이 일치합니다.

예를 들어 다음을 포함하는 이벤트 패턴을 살펴보십시오.

```
"resources": [
   "arn:aws:ec2:us-east-1:123456789012:instance/i-b188560f",
   "arn:aws:ec2:us-east-1:111122223333:instance/i-b188560f",
   "arn:aws:ec2:us-east-1:444455556666:instance/i-b188560f",
]
```

패턴 어레이의 첫 번째 항목이 이벤트 어레이의 두 번째 항목과 일치하기 때문에 이 예제 패턴은 다음 텍스트가 포함된 이벤트와 일치합니다.

```
"resources": [
   "arn:aws:autoscaling:us-east-1:123456789012:autoScalingGroup:eb56d16b-bbf0-401d-b893-
d5978ed4a025:autoScalingGroupName/ASGTerminate",
   "arn:aws:ec2:us-east-1:123456789012:instance/i-b188560f"
]
```

이벤트 패턴을 사용한 콘텐츠 기반 필터링

Amazon EventBridge는 이벤트 패턴을 사용하여 선언적 필터링을 지원합니다. 이벤트 패턴 콘텐츠 필터링을 사용하면 매우 특정한 조건에서만 트리거되는 복잡한 규칙을 작성할 수 있습니다. 예를 들어 이벤트 필드가 특정 숫자 범위 내에 있거나 이벤트가 특정 IP 주소에서 가져온 경우 또는 이벤트 JSON에 특정 필드가 없는

경우에만 트리거되는 규칙을 원할 수 있습니다. 콘텐츠 필터링을 사용하면 필터링 조건이 충족되는 경우에만 규칙이 대상을 호출하도록 이벤트 패턴에 복잡한 규칙을 만들 수 있습니다.

주제

접두사 일치

이벤트 소스에 있는 값의 접두사를 일치시킬 수 있습니다. 예를 들어 다음 이벤트 패턴은 `"time"` 필드가 `"2017-10-02"`로 시작된 모든 이벤트에서 일치합니다.

```
{
  "time": [ { "prefix": "2017-10-02" } ],
}
```

위의 이벤트 패턴은 `"time": "2017-10-02T18:43:48Z"`을(를) 포함하여 시간 필드에 해당 날짜가 있는 모든 이벤트와 일치합니다.

> Note
>
> 접두사 일치는 문자열 값 필드에서만 작동합니다.

접두사 일치 예제

유럽 리전의 모든 AWS Auto Scaling 이벤트를 처리한다고 가정합니다. 다음 이벤트 패턴은 일치시키는 방법을 보여줍니다.

```
{
  "source": [ "aws.autoscaling" ],
  "region": [ { "prefix": "eu-" } ]
}
```

Anything-but 일치

Anything-but 일치는 규칙에 제공된 것을 제외한 모든 것과 일치합니다.

문자열만 포함하거나 숫자만 포함하는 목록을 포함하여 문자열 및 숫자 값과 함께 `anything-but`을 사용할 수 있습니다.

다음은 문자열을 먼저 사용한 다음 숫자를 사용한 단일 `anything-but` 일치를 보여줍니다.

```
{
  "detail": {
    "state": [ { "anything-but": "initializing" } ]
  }
}

{
```

```
  "detail": {
    "x-limit": [ { "anything-but": 123 } ]
  }
}
```

다음은 문자열 목록과의 anything-but 일치를 보여줍니다.

```
{
  "detail": {
    "state": [ { "anything-but": [ "stopped", "overloaded" ] } ]
  }
}
```

다음은 숫자 목록과의 anything-but 일치를 보여줍니다.

```
{
  "detail": {
    "x-limit": [ { "anything-but": [ 100, 200, 300 ] } ]
  }
}
```

다음은 접두사와 일치하는 anything-but 이벤트 패턴을 보여 줍니다. "state" 필드의 init 접두사가 있는 이벤트를 제외한 모든 이벤트에서 일치합니다.

```
{
  "detail": {
    "state": [ { "anything-but": { "prefix": "init" } } ]
  }
}
```

Anything-but 일치 예제

특정 필드 값을 포함하기보다는 제외하려는 경우가 있습니다. API 호출의 AWS CloudTrail 보고서인 이벤트를 제외한 모든 이벤트를 처리한다고 가정합니다.

```
{
  "detail-type": [ { "anything-but": "AWS API Call via CloudTrail" } ]
}
```

anything-but 일치 표현식은 리터럴 문자열 또는 값 목록을 블랙리스트로 지정할 수 있습니다. 목록에는 모든 문자열 또는 모든 숫자가 포함되어야 합니다. Amazon EC2 또는 Amazon S3에서 가져온 이벤트를 제외한 모든 이벤트를 보려면 다음을 수행합니다.

```
{
  "source": [ { "anything-but": [ "aws.ec2", "aws.s3" ] } ]
}
```

anything-but 일치 표현식은 중첩된 일치 표현식을 사용하여 접두사를 제외할 수도 있습니다. 예를 들어 EventBridge 주요 이벤트 버스에는 모든 AWS 서비스에서 오는 많은 이벤트가 있습니다. 그러나 또한 PutEvents API를 사용하여 자체 이벤트를 주입 할 수 있습니다. 모든 AWS 이벤트의 “source” 필드는 "aws." 문자열로 시작하므로 AWS 이벤트를 구별하고 사용자 고유의 이벤트만 처리할 수 있습니다.

```
{
  "source": [ { "anything-but": { "prefix": "aws." } } ]
}
```

숫자 일치

다음은 이벤트 패턴에 대한 숫자 일치를 보여줍니다.

```
{
  "detail": {
    "c-count": [ { "numeric": [ ">", 0, "<=", 5 ] } ],
    "d-count": [ { "numeric": [ "<", 10 ] } ],
    "x-limit": [ { "numeric": [ "=", 3.018e2 ] } ]
  }
}
```

이 패턴은 모든 필드에 해당하는 평가와 일치합니다. 숫자 일치는 JSON 숫자 값에서만 작동하며 -1.0e9와 +1.0e9 사이의 값으로 제한되며 정밀도는 15자리(소수점 오른쪽의 6자리)입니다.

IP 주소 일치

IP 주소 일치는 IPv4 및 IPv6 주소 모두에 사용할 수 있습니다.

```
{
  "detail": {
    "source-ip": [ { "cidr": "10.0.0.0/24" } ]
  }
}
```

Exists 일치

Exists 일치는 이벤트의 JSON에서 필드의 유무에 따라 작동합니다.

다음 이벤트 패턴은 `detail.c-count` 필드가 없는 모든 이벤트와 일치합니다.

```
{
  "detail": {
    "c-count": [ { "exists": false  } ]
  }
}
```

> **Note**
>
> Exists 일치는 리프 노드에서만 작동합니다. 중간 노드에서는 작동하지 않습니다.

예를 들어 위의 패턴은 다음 이벤트와 일치합니다.

```
{
  "detail-type": [ "EC2 Instance State-change Notification" ],
  "resources": [ "arn:aws:ec2:us-east-1:123456789012:instance/i-02ebd4584a2ebd341" ],
  "detail": {
    "state": [ "initializing", "running" ]
  }
}
```

그러나 `c-count`는 리프 노드가 아니기 때문에 다음 이벤트와도 일치합니다.

```
{
  "detail-type": [ "EC2 Instance State-change Notification" ],
  "resources": [ "arn:aws:ec2:us-east-1:123456789012:instance/i-02ebd4584a2ebd341" ],
```

```
  "detail": {
    "state": [ "initializing", "running" ]
    "c-count" : {
       "c1" : 100
    }
  }
}
```

존재 일치 예제

일련의 이벤트에 대한 Elasticsearch 전체 텍스트 인덱스를 만들려 한다고 가정합니다. 이렇게 하려면 다음과 같이 설명 필드가 있는 모든 이벤트를 선택 합니다.

```
{
  "detail": {
  "description": [ { "exists": true } ],
  }
}
```

`{ "exists": false }`을(를) 사용하여 특정 필드를 포함하지 않는 이벤트를 선택할 수도 있습니다.

여러 일치가 있는 복잡한 예제

여러 일치 규칙을 보다 복잡한 이벤트 패턴으로 결합할 수 있습니다. 예를 들어 다음은 `anything-but`과 `numeric`을(를) 단일 이벤트 패턴으로 결합합니다.

```
{
  "time": [ { "prefix": "2017-10-02" } ],
  "detail": {
    "state": [ { "anything-but": "initializing" } ],
    "c-count": [ { "numeric": [ ">", 0, "<=", 5 ] } ],
    "d-count": [ { "numeric": [ "<", 10 ] } ],
    "x-limit": [ { "anything-but": [ 100, 200, 300 ] } ]
  }
}
```

대상 입력 변환

EventBridge의 입력 변환기 기능은 이벤트의 텍스트를 규칙의 대상으로 전달하기 전에 사용자 지정합니다. JSON 경로를 사용하여 원래 이벤트 소스의 값을 참조하는 변수를 정의할 수 있습니다. 입력에서 각 값을 할당하여 여러 변수를 정의할 수 있습니다. 그런 다음 <*variable-name*>으로 입력 템플릿 내에서 이러한 변수를 사용할 수 있습니다. "<" and ">" 문자는 이스케이프되지 않습니다.

JSON 경로를 정의하지 않고 사용할 수 있는 미리 정의된 세 가지 변수가 있습니다. 이러한 변수는 예약되어 있으며 이러한 이름으로 변수를 만들 수 없습니다.

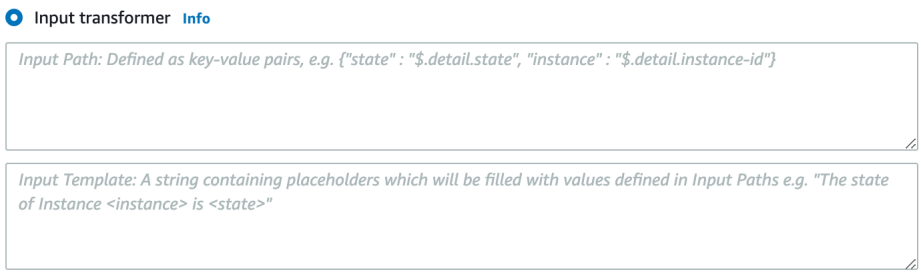
- `aws.events.rule-arn` — EventBridge 규칙의 Amazon 리소스 이름(ARN)입니다.
- `aws.events.rule-name` — EventBridge 규칙의 이름입니다.
- `aws.events.event` — 원래 이벤트의 복사본입니다.

입력 변환 예제

다음은 Amazon EC2 이벤트의 예입니다.

```
{
  "version": "0",
  "id": "7bf73129-1428-4cd3-a780-95db273d1602",
  "detail-type": "EC2 Instance State-change Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "2015-11-11T21:29:54Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2:us-east-1:123456789012:instance/i-abcd1111"
  ],
  "detail": {
    "instance-id": "i-0123456789",
    "state": "RUNNING"
  }
}
```

콘솔에서 규칙을 정의할 때 입력 구성 아래에서 입력 변환기 옵션을 선택합니다. 이 옵션은 텍스트 상자 두 개를 표시합니다. 하나는 입력 경로, 하나는 입력 템플릿에 대한 텍스트 상자입니다.



입력 경로는 변수를 정의하는 데 사용됩니다. JSON 경로를 사용하여 이벤트의 항목을 참조하고 해당 값을 변수에 저장합니다. 예를 들어 첫 번째 텍스트 상자에 다음을 입력하여 이벤트 예제의 값을 참조하는 입력 경로를 만들 수 있습니다.

```
{
```

```
  "instance" : "$.detail.instance",
  "state" : "$.detail.state"
}
```

이는 <instance> 및 <state>라는 두 가지 변수를 정의합니다. 입력 템플릿을 만들 때 이러한 변수를 참조할 수 있습니다.

입력 템플릿은 대상에 전달하려는 정보에 대한 템플릿입니다. 문자열이나 JSON을 대상에 전달하는 템플릿을 만들 수 있습니다. 다음 입력 템플릿 예제는 이전 이벤트 및 입력 경로를 사용하여 이벤트를 대상으로 라우팅하기 전에 해당 이벤트를 출력 예제로 변환합니다.

설명	템플릿	출력
단순 문자열	`"instance <instance> is in <state>"`	`"instance i-0123456789 is in RUNNING"`
이스케이프된 따옴표가 있는 문자열	`"instance \"<instance>\" is in <state>"`	`"instance \"i-0123456789\" is in RUNNING"` 이는 EventBridge 콘솔의 동작입니다. AWS CLI는 슬래시 문자를 이스케이프하고 결과는 `"instance "i-0123456789" is in RUNNING"`입니다.
단순 JSON	`{ "instance" : <instance>, "state": <state> }`	`{ "instance" : "i-0123456789", "state": "RUNNING" }`
변수와 정적 정보가 혼합된 JSON	`{ "instance" : <instance>, "state": [ 9, <state>, true ], "Transformed" : "Yes" }`	`{ "instance" : "i-0123456789", "state": [ 9, "RUNNING", true ], "Transformed" : "Yes" }`
JSON에 예약된 변수 포함	`{ "instance" : <instance>, "state": <state>, "ruleArn" : <aws.events.rule-arn>, "ruleName" : <aws.events.rule-name>, "originalEvent" : <aws.events.event> }`	`{ "instance" : "i-0123456789", "state": "RUNNING", "ruleArn" : "arn:aws:events:us-east-2:123456789012:rule/example", "ruleName" : "example", "originalEvent" : { ... // commented for brevity } }`

설명	템플릿	출력
문자열에 예약된 변수 포함	`"<aws.events.rule-name> triggered"`	`"example triggered"`

EventBridge API를 사용하여 입력 변환

EventBridge API에서 입력 변환을 사용하는 방법에 대한 자세한 내용과 예제는 입력 변환기를 사용하여 이벤트에서 데이터를 추출하고 대상에 해당 데이터를 입력 단원을 참조하십시오.

입력 변환과 관련된 일반적인 문제

다음은 EventBridge에서 입력을 변환할 때 나타나는 몇 가지 일반적인 문제입니다.

- 문자열의 경우 따옴표가 필요합니다.
- 템플릿에 대한 JSON 경로를 만들 때 검증이 수행되지 않습니다.
- 변수를 지정하여 이벤트에 존재하지 않는 JSON 경로를 일치시키는 경우 변수가 생성되지 않기 때문에 출력에 나타나지 않습니다.
- 대상에 전달되는 JSON은 축소되고 이스케이프됩니다.
- EventBridge는 대상에 대한 입력 템플릿을 채울 때 입력 경로에서 추출한 값을 이스케이프하지 않습니다.
- EventBridge는 JSON 내에서 따옴표 안의 변수를 지원하지 않습니다(예: {"value" : "instance <instance> is in <state>"}).

Amazon EventBridge 스키마 레지스트리

EventBridge 스키마 레지스트리를 사용하면 EventBridge의 이벤트에 대한 OpenAPI 스키마를 검색, 생성 및 관리할 수 있습니다. 기존 AWS 서비스에 대한 스키마를 찾고, 사용자 지정 스키마를 생성 및 업로드하거나, 이벤트 버스의 이벤트를 기반으로 스키마를 생성할 수 있습니다. EventBridge의 모든 스키마에 대해 코드 바인딩을 생성 및 다운로드하여 이러한 이벤트를 사용하는 애플리케이션을 신속하게 빌드할 수 있습니다.

스키마는 Amazon EventBridge의 모든 AWS 서비스 이벤트에 사용할 수 있습니다. 스키마를 생성 또는 업로드하거나 이벤트 버스의 이벤트에서 직접 스키마를 자동으로 유추할 수도 있습니다. 이벤트에 대한 스키마를 찾거나 만든 후에는 인기 있는 프로그래밍 언어에 대한 코드 바인딩을 다운로드할 수 있습니다. 스키마에 대한 코드 바인딩을 탐색, 검색, 생성, 업로드할 수 있습니다. API를 사용하여 Amazon EventBridge 콘솔에서 스키마를 관리하거나 AWS Toolkit를 사용하여 IDE에서 직접 스키마를 관리할 수 있습니다. AWS Serverless Application Model을 사용하여 이벤트를 사용하는 서버리스 앱을 빠르게 빌드할 수 있습니다.

API 또는 Amazon CloudFront을(를) 통해 EventBridge 스키마 레지스트리를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- Amazon EventBridge 스키마 레지스트리 API 참조
- AWS CloudFormation의 EventSchemas 리소스 유형 참조

주제

기존 AWS 이벤트 스키마 검색

Amazon EventBridge에는 EventBridge의 모든 AWS 서비스에 대한 스키마가 포함되어 있습니다. Amazon EventBridge 콘솔에서 또는 API 작업을 사용하여 이러한 스키마를 검색하거나 찾아볼 수 있습니다 (`SearchSchemas` 참조).

AWS 서비스용 스키마를 찾으려면

<result>

Schema details(스키마 세부 정보) 페이지가 표시됩니다.
</result>

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 Schemas(스키마)를 선택합니다.
3. Schemas(스키마) 페이지에서 AWS event schema registry(AWS 이벤트 스키마 레지스트리)를 선택합니다.
   <result>

   사용 가능한 스키마의 첫 페이지가 표시됩니다.

</result>

4. 스키마를 검색하려면 Search AWS event schemas(AWS 이벤트 스키마)에 검색어를 입력합니다.

   검색 시 사용 가능한 스키마의 이름과 내용 모두에 대해 일치하는 항목이 반환되고 해당 스키마의 버전이 표시됩니다.
5. 스키마 이름을 선택하여 이벤트 스키마를 엽니다.

스키마 레지스트리

스키마 레지스트리는 스키마의 컨테이너입니다. 레지스트리는 스키마가 논리적 그룹에 있도록 스키마를 수집하고 구성합니다. 모든 스키마 또는 기본 제공 스키마, AWS 이벤트 스키마 레지스트리 및 검색된 스키마 레지스트리를 볼 수 있습니다. 사용자 지정 레지스트리를 생성하여 생성하거나 업로드하는 스키마를 수집하고 구성할 수도 있습니다.

사용자 지정 레지스트리를 생성하려면

<result>

사용자 지정 레지스트리가 만들어집니다. 이제 기본 Schemas(스키마) 페이지에서 볼 수 있습니다.
</result>

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 Schemas(스키마)를 선택한 다음 Create registry(레지스트리 생성)를 선택합니다.
3. Registry details(레지스트리 세부 정보) 페이지에서 이름을 입력합니다.
4. 필요에 따라 새 레지스트리에 대한 설명을 입력합니다.
5. 생성을 선택합니다.

새 레지스트리에서 Create custom schema(사용자 지정 스키마 생성)를 선택하거나 새 스키마를 만들 때 해당 레지스트리를 선택할 수 있습니다.

Amazon EventBridge 스키마 레지스트리 API를 사용하여 레지스트리를 만들려면 `CreateRegistry` API 작업을 사용합니다.

스키마 업로드 또는 생성

스키마는 OpenAPI 사양에서 JSON 파일을 사용하여 정의됩니다. 이 사양을 사용하여 EventBridge에서 고유한 이벤트 스키마를 만들거나 업로드할 수 있습니다. 템플릿을 다운로드하거나 EventBridge 콘솔에서 직접 템플릿을 편집할 수 있습니다.

다운로드한 템플릿에서 스키마를 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 Schema registry(스키마 레지스트리)를 선택합니다.
3. 스키마 템플릿 아래의 Getting started(시작하기) 섹션에서 Download(다운로드)를 선택합니다.
4. 또는 다음 코드 예제에서 JSON을 다운로드할 수 있습니다.

```
{
    "openapi": "3.0.0",
    "info": {
      "version": "1.0.0",
      "title": "Event"
    },
```

```
  "paths": {},
  "components": {
    "schemas": {
      "Event": {
        "type": "object",
        "properties": {
          "ordinal": {
            "type": "number",
            "format": "int64"
          },
          "name": {
            "type": "string"
          },
          "price": {
            "type": "number",
            "format": "double"
          },
          "address": {
            "type": "string"
          },
          "comments": {
            "type": "array",
            "items": {
              "type": "string"
            }
          },
          "created_at": {
            "type": "string",
            "format": "date-time"
          }
        }
      }
    }
  }
}
```

5. 스키마가 이벤트와 일치하도록 템플릿을 편집합니다. 이벤트에 대한 자세한 내용은 EventBridge의 이벤트 및 이벤트 패턴 (p. 36) 단원을 참조하십시오.

스키마 파일을 업로드하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 Schemas(스키마), Create schema(스키마 생성) 순으로 선택합니다.
3. 필요에 따라 스키마 레지스트리를 선택하거나 생성합니다.
4. Schema details(스키마 세부 정보)에서 스키마의 이름을 입력합니다.
5. 필요에 따라 생성한 스키마에 대한 설명을 입력합니다.
6. Create(생성) 탭을 선택한 상태에서 스키마 파일을 텍스트 상자로 끌어오거나 스키마 소스를 붙여넣습니다.
7. 생성을 선택합니다.

콘솔에서 직접 템플릿을 편집하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 Schemas(스키마), Create schema(스키마 생성) 순으로 선택합니다.
3. 필요에 따라 스키마 레지스트리를 선택하거나 생성합니다.
4. Schema details(스키마 세부 정보)에서 스키마의 이름을 입력합니다.
5. 필요에 따라 생성한 스키마에 대한 설명을 입력합니다.
6. Create(생성) 탭을 선택한 상태에서 Load template(템플릿 로드)을 선택합니다.

7. 스키마가 이벤트와 일치하도록 템플릿을 편집합니다. 이벤트에 대한 자세한 내용은 EventBridge의 이벤트 및 이벤트 패턴 (p. 36) 단원을 참조하십시오.
8. 생성을 선택합니다.

스키마 레지스트리 API를 사용하여 EventBridge 스키마를 생성하려면 `CreateSchema` API 작업을 사용합니다.

이벤트 JSON에서 스키마 생성

이벤트의 JSON을 사용하면 이러한 유형의 이벤트에 대한 스키마를 자동으로 생성할 수 있습니다. 기존 이벤트 코드가 주어지면 사용자 지정 스키마를 빠르게 생성할 수 있습니다. 스키마가 생성되면 코드 바인딩을 다운로드하여 이러한 유형의 이벤트에 대한 애플리케이션을 만들 수 있습니다.

이벤트 JSON을 기반으로 EventBridge 스키마를 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 Schemas(스키마), Create schema(스키마 생성) 순으로 선택합니다.
3. 필요에 따라 스키마 레지스트리를 선택하거나 생성합니다.
4. Schema details(스키마 세부 정보)에서 스키마의 이름을 입력합니다.
5. 필요에 따라 생성한 스키마에 대한 설명을 입력합니다.
6. Discover from JSON(JSON에서 검색)을 선택합니다.
7. JSON 아래의 텍스트 상자에서 이벤트의 JSON 소스를 붙여넣거나 끌어옵니다.

   예를 들어 실패한 실행에 대해 이 AWS Step Functions 이벤트의 소스를 붙여넣을 수 있습니다.

```
{
    "version": "0",
    "id": "315c1398-40ff-a850-213b-158f73e60175",
    "detail-type": "Step Functions Execution Status Change",
    "source": "aws.states",
    "account": "012345678912",
    "time": "2019-02-26T19:42:21Z",
    "region": "us-east-1",
    "resources": [
      "arn:aws:states:us-east-1:012345678912:execution:state-machine-name:execution-
name"
    ],
    "detail": {
        "executionArn": "arn:aws:states:us-east-1:012345678912:execution:state-machine-
name:execution-name",
        "stateMachineArn": "arn:aws:states:us-east-1:012345678912:stateMachine:state-
machine",
        "name": "execution-name",
        "status": "FAILED",
        "startDate": 1551225146847,
        "stopDate": 1551225151881,
        "input": "{}",
        "output": null
    }
}
```

8. Discover schema(스키마 검색)를 선택합니다.
9. EventBridge는 이벤트에 대한 OpenAPI 스키마를 생성합니다. 예를 들어 붙여넣은 이벤트에 대해 생성된 스키마는 다음과 같습니다.

```
{
```

```
 "openapi": "3.0.0",
 "info": {
   "version": "1.0.0",
   "title": "StepFunctionsExecutionStatusChange"
 },
 "paths": {},
 "components": {
   "schemas": {
     "AWSEvent": {
       "type": "object",
       "required": ["detail-type", "resources", "detail", "id", "source", "time",
"region", "version", "account"],
       "x-amazon-events-detail-type": "Step Functions Execution Status Change",
       "x-amazon-events-source": "aws.states",
       "properties": {
         "detail": {
           "$ref": "#/components/schemas/StepFunctionsExecutionStatusChange"
         },
         "account": {
           "type": "string"
         },
         "detail-type": {
           "type": "string"
         },
         "id": {
           "type": "string"
         },
         "region": {
           "type": "string"
         },
         "resources": {
           "type": "array",
           "items": {
             "type": "string"
           }
         },
         "source": {
           "type": "string"
         },
         "time": {
           "type": "string",
           "format": "date-time"
         },
         "version": {
           "type": "string"
         }
       }
     },
     "StepFunctionsExecutionStatusChange": {
       "type": "object",
       "required": ["output", "input", "executionArn", "name", "stateMachineArn",
"startDate", "stopDate", "status"],
       "properties": {
         "executionArn": {
           "type": "string"
         },
         "input": {
           "type": "string"
         },
         "name": {
           "type": "string"
         },
         "output": {},
         "startDate": {
           "type": "integer",
           "format": "int64"
```

```
          },
          "stateMachineArn": {
            "type": "string"
          },
          "status": {
            "type": "string"
          },
          "stopDate": {
            "type": "integer",
            "format": "int64"
          }
        }
      }
    }
  }
}
```

10. 스키마가 생성되면 Create(생성)를 선택합니다.

이벤트 버스의 이벤트를 기반으로 스키마 생성

Amazon EventBridge는 이벤트 버스의 이벤트를 기반으로 스키마를 유추할 수 있습니다. 이벤트 버스에서 이벤트 검색을 사용하도록 설정하면 해당 버스의 이벤트에 대한 스키마가 생성됩니다.

> Note
>
> 이벤트 버스에서 이벤트 검색을 사용하도록 설정하면 비용이 발생할 수 있습니다. 매월 처음 5백만 건의 수집 이벤트는 무료입니다.

이벤트 버스에서 스키마 검색을 사용하도록 설정하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 이벤트 버스를 선택합니다.
3. Default event bus(기본 이벤트 버스)에서 검색을 사용하도록 설정하려면 Start discovery(검색 시작)를 선택합니다.
4. Custom event bus(사용자 지정 이벤트 버스)에서 검색을 활성화하려면 사용자 지정 이벤트 버스의 라디오 단추를 선택하고 Start Discovery(검색 시작)를 선택합니다.

검색된 스키마는 Schemas(스키마) 페이지의 Discovered schemas registry(검색된 스키마 레지스트리)에 표시됩니다. 해당 버스의 이벤트 내용을 변경하면 Discovered schemas registry(검색된 스키마 레지스트리)에 EventBridge 관련 스키마의 새 버전이 만들어집니다.

EventBridge 스키마에 대한 코드 바인딩 생성

이벤트 스키마에 대한 코드 바인딩을 생성하여 Java, Python 및 TypeScript 개발 속도를 높일 수 있습니다. 기존 AWS 서비스, 생성한 스키마 및 이벤트 버스의 이벤트를 기반으로 생성하는 스키마에 대한 코드 바인딩을 가져올 수 있습니다. EventBridge 콘솔, EventBridge 스키마 레지스트리 API를 사용하여 스키마에 대한 코드 바인딩을 생성하고 AWS Toolkit를 사용하여 IDE에서 직접 생성할 수 있습니다.

코드 바인딩을 생성하려면 이벤트 버스에서 검색을 사용하도록 설정해야 합니다.

EventBridge 스키마에서 코드 바인딩을 생성하려면

1. https://console.aws.amazon.com/events/에서 Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 Schemas(스키마)를 선택합니다.

3. 스키마 레지스트리를 살펴보거나 스키마를 검색하여 코드 바인딩을 원하는 스키마를 찾습니다.
4. 스키마 이름을 선택하여 Schema details(스키마 세부 정보) 페이지를 표시합니다.
5. Version(버전) 섹션에서 Download code bindings(코드 바인딩 다운로드)를 선택합니다.
6. Download code bindings(코드 바인딩 다운로드) 페이지에서 다운로드할 코드 바인딩의 언어를 선택합니다.
7. Download(다운로드)를 선택합니다.

   다운로드가 시작되는 데 몇 초가 걸릴 수 있습니다. 다운로드 파일은 선택한 언어에 대한 코드 바인딩의 zip 파일입니다.

EventBridge 스키마 레지스트리의 AWS Toolkit 통합

EventBridge 스키마 레지스트리는 일부 AWS Toolkit와 통합되므로 스키마를 찾아보거나 검색하고 IDE에서 직접 스키마에 대한 코드 바인딩을 다운로드할 수 있습니다.

자세한 내용은 다음 AWS Toolkit 설명서 링크를 참조하십시오.

- AWS Toolkit for JetBrains
- VS 코드용 AWS 도구 키트

EventBridge 지원되는 AWS 서비스의 이벤트 예제

다음 목록의 AWS 서비스는 EventBridge에서 감지할 수 있는 이벤트를 출력합니다.

또한 이벤트를 출력하지 않고 CloudTrail을 통해 전달되는 이벤트를 감시하면서 이 페이지에 나와 있지 않은 서비스에도 EventBridge를 사용할 수 있습니다. 자세한 내용은 CloudTrail을 통해 전달된 이벤트 (p. 102) 단원을 참조하십시오.

이벤트 유형

Amazon Augmented AI 이벤트

Amazon Augmented AI에서 생성된 이벤트의 예제는 Amazon Augmented AI에서 이벤트 사용을 참조하십시오.

Application Auto Scaling 이벤트

Application Auto Scaling에서 생성된 이벤트의 예는 애플리케이션 Auto Scaling 이벤트 및 EventBridge를 참조하십시오.

AWS Batch 이벤트

AWS Batch에서 생성되는 이벤트의 예제는 AWS Batch 이벤트를 참조하십시오.

Amazon EventBridge 예약된 이벤트

다음은 예약된 이벤트의 예제입니다.

```
{
  "id": "53dc4d37-cffa-4f76-80c9-8b7d4a4d2eaa",
  "detail-type": "Scheduled Event",
  "source": "aws.events",
  "account": "123456789012",
  "time": "2015-10-08T16:53:06Z",
  "region": "us-east-1",
  "resources": [ "arn:aws:events:us-east-1:123456789012:rule/MyScheduledRule" ],
  "detail": {}
}
```

Amazon Chime 이벤트

Amazon Chime에서 생성된 이벤트의 예제는 EventBridge를 사용하여 Amazon Chime 자동화를 참조하십시오.

CloudWatch의 이벤트

CloudWatch의 샘플 이벤트의 경우 AWS CodeBuild 사용 설명서의 경보 이벤트 및 EventBridge를 참조하십시오.

CodeBuild 이벤트

CodeBuild 샘플 이벤트의 경우 AWS CodeBuild 사용 설명서의 빌드 알림 입력 형식 참조를 참조하십시오.

CodeCommit 이벤트

CodeCommit 샘플 이벤트는 AWS CodeCommit 사용 설명서의 EventBridge 및 CloudWatch 이벤트에서 CodeCommit 이벤트 모니터링을 참조하십시오.

AWS CodeDeploy 이벤트

다음은 CodeDeploy 이벤트의 예제입니다. 자세한 내용은 AWS CodeDeploy User Guide의 CloudWatch 이벤트을 이용한 모니터링 배포를 참조하십시오.

CodeDeploy 배포 상태 변경 알림

배포 상태가 변경되었습니다.

```
{
  "account": "123456789012",
  "region": "us-east-1",
  "detail-type": "CodeDeploy Deployment State-change Notification",
  "source": "aws.codedeploy",
  "version": "0",
  "time": "2016-06-30T22:06:31Z",
  "id": "c071bfbf-83c4-49ca-a6ff-3df053957145",
  "resources": [
    "arn:aws:codedeploy:us-east-1:123456789012:application:myApplication",
    "arn:aws:codedeploy:us-east-1:123456789012:deploymentgroup:myApplication/
myDeploymentGroup"
  ],
  "detail": {
    "instanceGroupId": "9fd2fbef-2157-40d8-91e7-6845af69e2d2",
    "region": "us-east-1",
    "application": "myApplication",
    "deploymentId": "d-123456789",
    "state": "SUCCESS",
    "deploymentGroup": "myDeploymentGroup"
  }
}
```

CodeDeploy 인스턴스 상태 변경 알림

배포 그룹에 소속된 인스턴스의 상태가 변경되었습니다.

```
{
  "account": "123456789012",
  "region": "us-east-1",
  "detail-type": "CodeDeploy Instance State-change Notification",
```

```
  "source": "aws.codedeploy",
  "version": "0",
  "time": "2016-06-30T23:18:50Z",
  "id": "fb1d3015-c091-4bf9-95e2-d98521ab2ecb",
  "resources": [
    "arn:aws:ec2:us-east-1:123456789012:instance/i-0000000aaaaaaaaaa",
    "arn:aws:codedeploy:us-east-1:123456789012:deploymentgroup:myApplication/
myDeploymentGroup",
    "arn:aws:codedeploy:us-east-1:123456789012:application:myApplication"
  ],
  "detail": {
    "instanceId": "i-0000000aaaaaaaaaa",
    "region": "us-east-1",
    "state": "SUCCESS",
    "application": "myApplication",
    "deploymentId": "d-123456789",
    "instanceGroupId": "8cd3bfa8-9e72-4cbe-a1e5-da4efc7efd49",
    "deploymentGroup": "myDeploymentGroup"
  }
}
```

CodePipeline 이벤트

다음은 CodePipeline 이벤트의 예제입니다.

파이프라인 실행 상태 변경

```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "CodePipeline Pipeline Execution State Change",
  "source": "aws.codepipeline",
  "account": "123456789012",
  "time": "2017-04-22T03:31:47Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:codepipeline:us-east-1:123456789012:pipeline:myPipeline"
  ],
  "detail": {
    "pipeline": "myPipeline",
    "version": "1",
    "state": "STARTED",
    "execution-id": "01234567-0123-0123-0123-012345678901"
  }
}
```

단계 실행 상태 변경

```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "CodePipeline Stage Execution State Change",
  "source": "aws.codepipeline",
  "account": "123456789012",
  "time": "2017-04-22T03:31:47Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:codepipeline:us-east-1:123456789012:pipeline:myPipeline"
  ],
  "detail": {
    "pipeline": "myPipeline",
```

```
    "version": "1",
    "execution-id": "01234567-0123-0123-0123-012345678901",
    "stage": "Prod",
    "state": "STARTED"
  }
}
```

작업 실행 상태 변경

이 예제에는 두 개의 `region` 필드가 있습니다. 상단의 필드는 대상 파이프라인의 작업이 실행된 AWS 리전의 이름입니다. 이 예에서는 `us-east-1`입니다. `detail` 섹션의 `region`은 이벤트가 생성된 AWS 리전입니다. 이 리전은 파이프라인이 생성된 리전과 동일합니다. 이 예에서는 `us-west-2`입니다.

```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "CodePipeline Action Execution State Change",
  "source": "aws.codepipeline",
  "account": "123456789012",
  "time": "2017-04-22T03:31:47Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:codepipeline:us-east-1:123456789012:pipeline:myPipeline"
  ],
  "detail": {
    "pipeline": "myPipeline",
    "version": 1,
    "execution-id": "01234567-0123-0123-0123-012345678901",
    "stage": "Prod",
    "action": "myAction",
    "state": "STARTED",
    "region":"us-west-2",
    "type": {
      "owner": "AWS",
      "category": "Deploy",
      "provider": "CodeDeploy",
      "version": 1
    }
  }
}
```

AWS Config 이벤트

AWS Config 이벤트에 대한 자세한 내용은 AWS Config Developer Guide의 Amazon CloudWatch Events로 AWS Config 모니터링을 참조하십시오.

Amazon EBS 이벤트

Amazon EBS 이벤트에 대한내용은 Amazon CloudWatch Events의Amazon EBS Linux 인스턴스용 Amazon EC2 사용 설명서에 대해 를 참조하십시오.

Amazon EC2 Auto Scaling 이벤트

Auto Scaling 이벤트에 대한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Auto Scaling 그룹 조정 시 CloudWatch 이벤트 수신을 참조하십시오.

Amazon EC2 스팟 인스턴스 중단 이벤트

스팟 인스턴스 중단에 대한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 스팟 인스턴스 중단 공지를 참조하십시오.

Amazon EC2 상태 변경 이벤트

다음은 인스턴스 상태가 변경할 때 Amazon EC2 인스턴스 이벤트의 예입니다.

EC2 인스턴스 상태 변경 알림

이 예제는 `pending` 상태 인스턴스에 대한 것입니다. `state`에 가능한 기타 값은 `running`, `shutting-down`, `stopped`, `stopping`, `terminated`입니다.

```
{
   "id":"7bf73129-1428-4cd3-a780-95db273d1602",
   "detail-type":"EC2 Instance State-change Notification",
   "source":"aws.ec2",
   "account":"123456789012",
   "time":"2015-11-11T21:29:54Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ec2:us-east-1:123456789012:instance/i-abcd1111"
   ],
   "detail":{
      "instance-id":"i-abcd1111",
      "state":"pending"
   }
}
```

Amazon Elastic Container Registry(Amazon ECR) 이벤트

Amazon ECR은 이미지 작업 이벤트를 EventBridge로 전송합니다. 이미지가 푸시, 스캔 또는 삭제될 때 이벤트가 전송됩니다.

Amazon ECS 샘플 이벤트는 Amazon Elastic Container Registry 사용 설명서의 Amazon ECR 이벤트를 참조하십시오.

Amazon Elastic Container Service(Amazon ECS) 이벤트

Amazon ECS는 두 유형의 이벤트, 즉 컨테이너 인스턴스 이벤트와 작업 이벤트를 EventBridge로 전송합니다. 해당 작업에 EC2 시작 유형을 사용 중이라면 컨테이너 인스턴스 이벤트만 전송됩니다. Fargate 시작 유형을 사용하는 작업의 경우 작업 상태 이벤트만 수신하게 됩니다. Amazon ECS는 컨테이너 인스턴스 및 작업의 상태를 추적합니다. 리소스 중 하나가 변경되면 이벤트가 트리거됩니다. 이러한 이벤트는 컨테이너 인스턴스 상태 변경 이벤트 또는 작업 상태 변경 이벤트로 분류됩니다.

Amazon ECS 샘플 이벤트는 Amazon Elastic Container Service Developer Guide의 Amazon ECS 이벤트를 참조하십시오.

AWS Elemental MediaConvert 이벤트

MediaConvert 샘플 이벤트에 대한 자세한 내용은 AWS Elemental MediaConvert 사용 설명서의 CloudWatch 이벤트를 사용하여 AWS Elemental MediaConvert 작업 모니터링을 참조하십시오.

AWS Elemental MediaPackage 이벤트

MediaPackage 샘플 이벤트에 대한 자세한 내용은 AWS Elemental MediaPackage 사용 설명서의 Amazon CloudWatch Events로 AWS Elemental MediaPackage 모니터링을 참조하십시오.

AWS Elemental MediaStore 이벤트

MediaStore 샘플 이벤트에 대한 자세한 내용은 AWS Elemental MediaStore 사용 설명서.의 CloudWatch 이벤트로 AWS Elemental MediaStore 자동화를 참조하십시오.

Amazon EMR 이벤트

Amazon EMR이 보고하는 이벤트에는 `Source` 값으로 `aws.emr`이 있는 반면, CloudTrail을 통해 보고되는 Amazon EMR API 이벤트에는 `Source` 값으로 `aws.elasticmapreduce`가 있습니다.

다음은 Amazon EMR이 보고하는 이벤트의 예제입니다.

Amazon EMR Auto Scaling 정책 상태 변경

```
{
   "version":"0",
   "id":"2f8147ab-8c48-47c6-b0b6-3ee23ec8d300",
   "detail-type":"EMR Auto Scaling Policy State Change",
   "source":"aws.emr",
   "account":"123456789012",
   "time":"2016-12-16T20:42:44Z",
   "region":"us-east-1",
   "resources":[],
   "detail":{
      "resourceId":"ig-X2LBMHTGPCBU",
      "clusterId":"j-1YONHTCP3YZKC",
      "state":"PENDING",
      "message":"AutoScaling policy modified by user request",
      "scalingResourceType":"INSTANCE_GROUP"
   }
}
```

Amazon EMR 클러스터 상태 변경 – 시작

```
{
  "version": "0",
  "id": "999cccaa-eaaa-0000-1111-123456789012",
  "detail-type": "EMR Cluster State Change",
  "source": "aws.emr",
  "account": "123456789012",
  "time": "2016-12-16T20:43:05Z",
  "region": "us-east-1",
```

```
  "resources": [],
  "detail": {
    "severity": "INFO",
    "stateChangeReason": "{\"code\":\"\"}",
    "name": "Development Cluster",
    "clusterId": "j-123456789ABCD",
    "state": "STARTING",
    "message": "Amazon EMR cluster j-123456789ABCD (Development Cluster) was requested at
 2016-12-16 20:42 UTC and  is being created."
  }
}
```

Amazon EMR 클러스터 상태 변경 – 종료

```
{
  "version": "0",
  "id": "1234abb0-f87e-1234-b7b6-000000123456",
  "detail-type": "EMR Cluster State Change",
  "source": "aws.emr",
  "account": "123456789012",
  "time": "2016-12-16T21:00:23Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "severity": "INFO",
    "stateChangeReason": "{\"code\":\"USER_REQUEST\",\"message\":\"Terminated by user
 request\"}",
    "name": "Development Cluster",
    "clusterId": "j-123456789ABCD",
    "state": "TERMINATED",
    "message": "Amazon EMR Cluster jj-123456789ABCD (Development Cluster) has terminated at
 2016-12-16 21:00 UTC with a reason of USER_REQUEST."
  }
}
```

Amazon EMR 인스턴스 그룹 상태 변경

```
{
  "version": "0",
  "id": "999cccaa-eaaa-0000-1111-123456789012",
  "detail-type": "EMR Instance Group State Change",
  "source": "aws.emr",
  "account": "123456789012",
  "time": "2016-12-16T20:57:47Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "market": "ON_DEMAND",
    "severity": "INFO",
    "requestedInstanceCount": "2",
    "instanceType": "m3.xlarge",
    "instanceGroupType": "CORE",
    "instanceGroupId": "ig-ABCDEFGHIJKL",
    "clusterId": "j-123456789ABCD",
    "runningInstanceCount": "2",
    "state": "RUNNING",
    "message": "The resizing operation for instance group ig-ABCDEFGHIJKL in Amazon EMR
 cluster j-123456789ABCD (Development Cluster) is complete. It now has an instance count of
 2. The resize started at 2016-12-16 20:57 UTC and took 0 minutes to complete."
  }
}
```

Amazon EMR 단계 상태 변경

```
{
  "version": "0",
  "id": "999cccaa-eaaa-0000-1111-123456789012",
  "detail-type": "EMR Step Status Change",
  "source": "aws.emr",
  "account": "123456789012",
  "time": "2016-12-16T20:53:09Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "severity": "ERROR",
    "actionOnFailure": "CONTINUE",
    "stepId": "s-ZYXWVUTSRQPON",
    "name": "CustomJAR",
    "clusterId": "j-123456789ABCD",
    "state": "FAILED",
    "message": "Step s-ZYXWVUTSRQPON (CustomJAR) in Amazon EMR cluster j-123456789ABCD
 (Development Cluster) failed at 2016-12-16 20:53 UTC."
  }
}
```

Amazon GameLift 이벤트

다음은 Amazon GameLift 이벤트의 예제입니다. 자세한 내용은 Amazon GameLift 개발자 안내서의 FlexMatch 이벤트 참조를 참조하십시오.

매치메이킹 검색

```
{
  "version": "0",
  "id": "cc3d3ebe-1d90-48f8-b268-c96655b8f013",
  "detail-type": "GameLift Matchmaking Event",
  "source": "aws.gamelift",
  "account": "123456789012",
  "time": "2017-08-08T21:15:36.421Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:gamelift:us-west-2:123456789012:matchmakingconfiguration/SampleConfiguration"
  ],
  "detail": {
    "tickets": [
      {
        "ticketId": "ticket-1",
        "startTime": "2017-08-08T21:15:35.676Z",
        "players": [
          {
            "playerId": "player-1"
          }
        ]
      }
    ],
    "estimatedWaitMillis": "NOT_AVAILABLE",
    "type": "MatchmakingSearching",
    "gameSessionInfo": {
      "players": [
        {
          "playerId": "player-1"
        }
      ]
    }
  }
```

```
}
```

잠재적 매치 생성됨

```
{
  "version": "0",
  "id": "fce8633f-aea3-45bc-aeba-99d639cad2d4",
  "detail-type": "GameLift Matchmaking Event",
  "source": "aws.gamelift",
  "account": "123456789012",
  "time": "2017-08-08T21:17:41.178Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:gamelift:us-west-2:123456789012:matchmakingconfiguration/SampleConfiguration"
  ],
  "detail": {
    "tickets": [
      {
        "ticketId": "ticket-1",
        "startTime": "2017-08-08T21:15:35.676Z",
        "players": [
          {
            "playerId": "player-1",
            "team": "red"
          }
        ]
      },
      {
        "ticketId": "ticket-2",
        "startTime": "2017-08-08T21:17:40.657Z",
        "players": [
          {
            "playerId": "player-2",
            "team": "blue"
          }
        ]
      }
    ],
    "acceptanceTimeout": 600,
    "ruleEvaluationMetrics": [
      {
        "ruleName": "EvenSkill",
        "passedCount": 3,
        "failedCount": 0
      },
      {
        "ruleName": "EvenTeams",
        "passedCount": 3,
        "failedCount": 0
      },
      {
        "ruleName": "FastConnection",
        "passedCount": 3,
        "failedCount": 0
      },
      {
        "ruleName": "NoobSegregation",
        "passedCount": 3,
        "failedCount": 0
      }
    ],
    "acceptanceRequired": true,
    "type": "PotentialMatchCreated",
    "gameSessionInfo": {
      "players": [
```

```
        {
          "playerId": "player-1",
          "team": "red"
        },
        {
          "playerId": "player-2",
          "team": "blue"
        }
      ]
    },
    "matchId": "3faf26ac-f06e-43e5-8d86-08feff26f692"
  }
}
```

매치 수락

```
{
  "version": "0",
  "id": "b3f76d66-c8e5-416a-aa4c-aa1278153edc",
  "detail-type": "GameLift Matchmaking Event",
  "source": "aws.gamelift",
  "account": "123456789012",
  "time": "2017-08-09T20:04:42.660Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:gamelift:us-west-2:123456789012:matchmakingconfiguration/SampleConfiguration"
  ],
  "detail": {
    "tickets": [
      {
        "ticketId": "ticket-1",
        "startTime": "2017-08-09T20:01:35.305Z",
        "players": [
          {
            "playerId": "player-1",
            "team": "red"
          }
        ]
      },
      {
        "ticketId": "ticket-2",
        "startTime": "2017-08-09T20:04:16.637Z",
        "players": [
          {
            "playerId": "player-2",
            "team": "blue",
            "accepted": false
          }
        ]
      }
    ],
    "type": "AcceptMatch",
    "gameSessionInfo": {
      "players": [
        {
          "playerId": "player-1",
          "team": "red"
        },
        {
          "playerId": "player-2",
          "team": "blue",
          "accepted": false
        }
      ]
    },
```

```
    "matchId": "848b5f1f-0460-488e-8631-2960934d13e5"
  }
}
```

매치 수락 완료됨

```
{
  "version": "0",
  "id": "b1990d3d-f737-4d6c-b150-af5ace8c35d3",
  "detail-type": "GameLift Matchmaking Event",
  "source": "aws.gamelift",
  "account": "123456789012",
  "time": "2017-08-08T20:43:14.621Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:gamelift:us-west-2:123456789012:matchmakingconfiguration/SampleConfiguration"
  ],
  "detail": {
    "tickets": [
      {
        "ticketId": "ticket-1",
        "startTime": "2017-08-08T20:30:40.972Z",
        "players": [
          {
            "playerId": "player-1",
            "team": "red"
          }
        ]
      },
      {
        "ticketId": "ticket-2",
        "startTime": "2017-08-08T20:33:14.111Z",
        "players": [
          {
            "playerId": "player-2",
            "team": "blue"
          }
        ]
      }
    ],
    "acceptance": "TimedOut",
    "type": "AcceptMatchCompleted",
    "gameSessionInfo": {
      "players": [
        {
          "playerId": "player-1",
          "team": "red"
        },
        {
          "playerId": "player-2",
          "team": "blue"
        }
      ]
    },
    "matchId": "a0d9bd24-4695-4f12-876f-ea6386dd6dce"
  }
}
```

매치메이킹 성공함

```
{
  "version": "0",
  "id": "5ccb6523-0566-412d-b63c-1569e00d023d",
```

```
  "detail-type": "GameLift Matchmaking Event",
  "source": "aws.gamelift",
  "account": "123456789012",
  "time": "2017-08-09T19:59:09.159Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:gamelift:us-west-2:123456789012:matchmakingconfiguration/SampleConfiguration"
  ],
  "detail": {
    "tickets": [
      {
        "ticketId": "ticket-1",
        "startTime": "2017-08-09T19:58:59.277Z",
        "players": [
          {
            "playerId": "player-1",
            "playerSessionId": "psess-6e7c13cf-10d6-4756-a53f-db7de782ed67",
            "team": "red"
          }
        ]
      },
      {
        "ticketId": "ticket-2",
        "startTime": "2017-08-09T19:59:08.663Z",
        "players": [
          {
            "playerId": "player-2",
            "playerSessionId": "psess-786b342f-9c94-44eb-bb9e-c1de46c472ce",
            "team": "blue"
          }
        ]
      }
    ],
    "type": "MatchmakingSucceeded",
    "gameSessionInfo": {
      "gameSessionArn": "arn:aws:gamelift:us-west-2:123456789012:gamesession/836cf48d-
bcb0-4a2c-bec1-9c456541352a",
      "ipAddress": "192.168.1.1",
      "port": 10777,
      "players": [
        {
          "playerId": "player-1",
          "playerSessionId": "psess-6e7c13cf-10d6-4756-a53f-db7de782ed67",
          "team": "red"
        },
        {
          "playerId": "player-2",
          "playerSessionId": "psess-786b342f-9c94-44eb-bb9e-c1de46c472ce",
          "team": "blue"
        }
      ]
    },
    "matchId": "c0ec1a54-7fec-4b55-8583-76d67adb7754"
  }
}
```

매치메이킹 시간 초과됨

```
{
  "version": "0",
  "id": "fe528a7d-46ad-4bdc-96cb-b094b5f6bf56",
  "detail-type": "GameLift Matchmaking Event",
  "source": "aws.gamelift",
  "account": "123456789012",
  "time": "2017-08-09T20:11:35.598Z",
```

```
  "region": "us-west-2",
  "resources": [
    "arn:aws:gamelift:us-west-2:123456789012:matchmakingconfiguration/SampleConfiguration"
  ],
  "detail": {
    "reason": "TimedOut",
    "tickets": [
      {
        "ticketId": "ticket-1",
        "startTime": "2017-08-09T20:01:35.305Z",
        "players": [
          {
            "playerId": "player-1",
            "team": "red"
          }
        ]
      }
    ],
    "ruleEvaluationMetrics": [
      {
        "ruleName": "EvenSkill",
        "passedCount": 3,
        "failedCount": 0
      },
      {
        "ruleName": "EvenTeams",
        "passedCount": 3,
        "failedCount": 0
      },
      {
        "ruleName": "FastConnection",
        "passedCount": 3,
        "failedCount": 0
      },
      {
        "ruleName": "NoobSegregation",
        "passedCount": 3,
        "failedCount": 0
      }
    ],
    "type": "MatchmakingTimedOut",
    "message": "Removed from matchmaking due to timing out.",
    "gameSessionInfo": {
      "players": [
        {
          "playerId": "player-1",
          "team": "red"
        }
      ]
    }
  }
}
```

매치메이킹 취소됨

```
{
  "version": "0",
  "id": "8d6f84da-5e15-4741-8d5c-5ac99091c27f",
  "detail-type": "GameLift Matchmaking Event",
  "source": "aws.gamelift",
  "account": "123456789012",
  "time": "2017-08-09T20:00:07.843Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:gamelift:us-west-2:123456789012:matchmakingconfiguration/SampleConfiguration"
```

```
  ],
  "detail": {
    "reason": "Cancelled",
    "tickets": [
      {
        "ticketId": "ticket-1",
        "startTime": "2017-08-09T19:59:26.118Z",
        "players": [
          {
            "playerId": "player-1"
          }
        ]
      }
    ],
    "ruleEvaluationMetrics": [
      {
        "ruleName": "EvenSkill",
        "passedCount": 0,
        "failedCount": 0
      },
      {
        "ruleName": "EvenTeams",
        "passedCount": 0,
        "failedCount": 0
      },
      {
        "ruleName": "FastConnection",
        "passedCount": 0,
        "failedCount": 0
      },
      {
        "ruleName": "NoobSegregation",
        "passedCount": 0,
        "failedCount": 0
      }
    ],
    "type": "MatchmakingCancelled",
    "message": "Cancelled by request.",
    "gameSessionInfo": {
      "players": [
        {
          "playerId": "player-1"
        }
      ]
    }
  }
}
```

매치메이킹 실패함

```
{
  "version": "0",
  "id": "025b55a4-41ac-4cf4-89d1-f2b3c6fd8f9d",
  "detail-type": "GameLift Matchmaking Event",
  "source": "aws.gamelift",
  "account": "123456789012",
  "time": "2017-08-16T18:41:09.970Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:gamelift:us-west-2:123456789012:matchmakingconfiguration/SampleConfiguration"
  ],
  "detail": {
    "tickets": [
      {
        "ticketId": "ticket-1",
```

```
        "startTime": "2017-08-16T18:41:02.631Z",
        "players": [
          {
            "playerId": "player-1",
            "team": "red"
          }
        ]
      }
    ],
    "customEventData": "foo",
    "type": "MatchmakingFailed",
    "reason": "UNEXPECTED_ERROR",
    "message": "An unexpected error was encountered during match placing.",
    "gameSessionInfo": {
      "players": [
        {
          "playerId": "player-1",
          "team": "red"
        }
      ]
    },
    "matchId": "3ea83c13-218b-43a3-936e-135cc570cba7"
  }
}
```


AWS Glue 이벤트

다음은 AWS Glue 이벤트의 형식입니다.

작업 실행 성공


```
{
    "version":"0",
    "id":"abcdef00-1234-5678-9abc-def012345678",
    "detail-type":"Glue Job State Change",
    "source":"aws.glue",
    "account":"123456789012",
    "time":"2017-09-07T18:57:21Z",
    "region":"us-west-2",
    "resources":[],
    "detail":{
        "jobName":"MyJob",
        "severity":"INFO",
        "state":"SUCCEEDED",
        "jobRunId":"jr_abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789",
        "message":"Job run succeeded"
    }
}
```


작업 실행 실패


```
{
    "version":"0",
    "id":"abcdef01-1234-5678-9abc-def012345678",
    "detail-type":"Glue Job State Change",
    "source":"aws.glue",
    "account":"123456789012",
    "time":"2017-09-07T06:02:03Z",
    "region":"us-west-2",
    "resources":[],
    "detail":{
```

```
        "jobName":"MyJob",
        "severity":"ERROR",
        "state":"FAILED",
        "jobRunId":"jr_0123456789abcdef0123456789abcdef0123456789abcdef0123456789abcdef",
        "message":"JobName:MyJob and
 JobRunId:jr_0123456789abcdef0123456789abcdef0123456789abcdef0123456789abcdef failed to
 execute with exception Role arn:aws:iam::123456789012:role/Glue_Role should be given
 assume role permissions for Glue Service."
    }
}
```

Timeout

```
{
    "version":"0",
    "id":"abcdef00-1234-5678-9abc-def012345678",
    "detail-type":"Glue Job State Change",
    "source":"aws.glue",
    "account":"123456789012",
    "time":"2017-11-20T20:22:06Z",
    "region":"us-east-1",
    "resources":[],
    "detail":{
        "jobName":"MyJob",
        "severity":"WARN",
        "state":"TIMEOUT",
        "jobRunId":"jr_abc0123456789abcdef0123456789abcdef0123456789abcdef0123456789def",
        "message":"Job run timed out"
    }
}
```

중지된 작업 실행

```
{
   "version":"0",
   "id":"abcdef00-1234-5678-9abc-def012345678",
   "detail-type":"Glue Job State Change",
   "source":"aws.glue",
   "account":"123456789012",
   "time":"2017-11-20T20:22:06Z",
   "region":"us-east-1",
   "resources":[],
   "detail":{
       "jobName":"MyJob",
       "severity":"INFO",
       "state":"STOPPED",
       "jobRunId":"jr_abc0123456789abcdef0123456789abcdef0123456789abcdef0123456789def",
       "message":"Job run stopped"
    }
}
```

크롤러 시작됨

```
{
   "version":"0",
   "id":"05efe8a2-c309-6884-a41b-3508bcdc9695",
   "detail-type":"Glue Crawler State Change",
   "source":"aws.glue",
   "account":"561226563745",
   "time":"2017-11-11T01:09:46Z",
   "region":"us-east-1",
   "resources":[
```

```
   ],
   "detail":{
      "accountId":"561226563745",
      "crawlerName":"S3toS3AcceptanceTestCrawlera470bd94-9e00-4518-8942-e80c8431c322",
      "startTime":"2017-11-11T01:09:46Z",
      "state":"Started",
      "message":"Crawler Started"
   }
}
```

크롤러 성공

```
{
   "version":"0",
   "id":"3d675db5-59b9-6388-b8e8-e0a9b6d567a9",
   "detail-type":"Glue Crawler State Change",
   "source":"aws.glue",
   "account":"561226563745",
   "time":"2017-11-11T01:25:00Z",
   "region":"us-east-1",
   "resources":[

   ],
   "detail":{
      "tablesCreated":"0",
      "warningMessage":"N/A",
      "partitionsUpdated":"0",
      "tablesUpdated":"0",
      "message":"Crawler Succeeded",
      "partitionsDeleted":"0",
      "accountId":"561226563745",
      "runningTime (sec)":"7",
      "tablesDeleted":"0",
      "crawlerName":"SchedulerTestCrawler51fb3a8b-1015-49f0-a969-ca126680b94b",
      "completionDate":"2017-11-11T01:25:00Z",
      "state":"Succeeded",
      "partitionsCreated":"0",
      "cloudWatchLogLink":"https://console.aws.amazon.com/
cloudwatch/home?region=us-east-1#logEventViewer:group=/aws-glue/
crawlers;stream=SchedulerTestCrawler51fb3a8b-1015-49f0-a969-ca126680b94b"
   }
}
```

크롤러 실패

```
{
   "version":"0",
   "id":"f7965b59-470f-2e06-bb89-a8cebaabefac",
   "detail-type":"Glue Crawler State Change",
   "source":"aws.glue",
   "account":"782104008917",
   "time":"2017-10-20T05:10:08Z",
   "region":"us-east-1",
   "resources":[

   ],
   "detail":{
      "crawlerName":"test-crawler-notification",
      "errorMessage":"Internal Service Exception",
      "accountId":"1234",
      "cloudWatchLogLink":"https://console.aws.amazon.com/cloudwatch/home?region=us-
east-1#logEventViewer:group=/aws-glue/crawlers;stream=test-crawler-notification",
```

```
      "state":"Failed",
      "message":"Crawler Failed"
   }
}
```

작업 실행이 시작 중 상태입니다.

```
{
   "version":"0",
   "id":"66fbc5e1-aac3-5e85-63d0-856ec669a050",
   "detail-type":"Glue Job Run Status",
   "source":"aws.glue",
   "account":"123456789012",
   "time":"2018-04-24T20:57:34Z",
   "region":"us-east-1",
   "resources":[],
   "detail":{
      "jobName":"MyJob",
      "severity":"INFO",
      "notificationCondition":{
         "NotifyDelayAfter":1.0
      },
      "state":"STARTING",
          "jobRunId":"jr_6aa58e7a3aa44e2e4c7db2c50e2f7396cb57901729e4b702dcb2cfbbeb3f7a86",
      "message":"Job is in STARTING state",
      "startedOn":"2018-04-24T20:55:47.941Z"
   }
}
```

작업 실행이 실행 중 상태입니다.

```
{
   "version":"0",
   "id":"66fbc5e1-aac3-5e85-63d0-856ec669a050",
   "detail-type":"Glue Job Run Status",
   "source":"aws.glue",
   "account":"123456789012",
   "time":"2018-04-24T20:57:34Z",
   "region":"us-east-1",
   "resources":[],
   "detail":{
      "jobName":"MyJob",
      "severity":"INFO",
      "notificationCondition":{
         "NotifyDelayAfter":1.0
      },
      "state":"RUNNING",
    "jobRunId":"jr_6aa58e7a3aa44e2e4c7db2c50e2f7396cb57901729e4b702dcb2cfbbeb3f7a86",
      "message":"Job is in RUNNING state",
      "startedOn":"2018-04-24T20:55:47.941Z"
   }
}
```

작업 실행이 중단 상태입니다.

```
{
   "version":"0",
   "id":"66fbc5e1-aac3-5e85-63d0-856ec669a050",
   "detail-type":"Glue Job Run Status",
   "source":"aws.glue",
   "account":"123456789012",
   "time":"2018-04-24T20:57:34Z",
```

```
   "region":"us-east-1",
   "resources":[],
   "detail":{
      "jobName":"MyJob",
      "severity":"INFO",
      "notificationCondition":{
         "NotifyDelayAfter":1.0
      },
      "state":"STOPPING",
    "jobRunId":"jr_6aa58e7a3aa44e2e4c7db2c50e2f7396cb57901729e4b702dcb2cfbbeb3f7a86",
      "message":"Job is in STOPPING state",
      "startedOn":"2018-04-24T20:55:47.941Z"
   }
}
```

AWS Glue 데이터 카탈로그 테이블 상태 변경

```
{
    "version": "0",
    "id": "2617428d-715f-edef-70b8-d210da0317a0",
    "detail-type": "Glue Data Catalog Table State Change",
    "source": "aws.glue",
    "account": "123456789012",
    "time": "2019-01-16T18:16:01Z",
    "region": "eu-west-1",
    "resources": [
        "arn:aws:glue:eu-west-1:123456789012:table/d1/t1"
    ],
    "detail": {
        "databaseName": "d1",
        "changedPartitions": [
            "[C.pdf, dir3]",
            "[D.doc, dir4]"
        ],
        "typeOfChange": "BatchCreatePartition",
        "tableName": "t1"
    }
}
```

AWS Glue 데이터 카탈로그 데이터베이스 상태 변경

다음 예제에서 `typeofChange`은 `CreateTable`입니다. 이 필드에 사용할 수 있는 다른 값은 `CreateDatabase`와 `UpdateTable`입니다.

```
{
    "version": "0",
    "id": "60e7ddc2-a588-5328-220a-21c060f6c3f4",
    "detail-type": "Glue Data Catalog Database State Change",
    "source": "aws.glue",
    "account": "123456789012",
    "time": "2019-01-16T18:08:48Z",
    "region": "eu-west-1",
    "resources": [
        "arn:aws:glue:eu-west-1:123456789012:table/d1/t1"
    ],
    "detail": {
        "databaseName": "d1",
        "typeOfChange": "CreateTable",
        "changedTables": [
            "t1"
        ]
    }
}
```

AWS IoT Greengrass 이벤트

AWS IoT Greengrass 이벤트에 대한 자세한 내용은 AWS IoT Greengrass 개발자 안내서의 배포 알림 가져오기를 참조하십시오.

AWS Ground Station 이벤트

예제 AWS Ground Station 이벤트에 대한 자세한 내용은 AWS Ground Station 사용 설명서의 CloudWatch 이벤트를 사용한 AWS Ground Station 자동화를 참조하십시오.

Amazon GuardDuty 이벤트

예제 Amazon GuardDuty 이벤트에 관한 내용은 Amazon GuardDuty 사용 설명서의 Amazon CloudWatch Events를 사용하여 Amazon GuardDuty 모니터링을 참조하십시오.

AWS 상태 이벤트

다음은 AWS Personal Health Dashboard(AWS Health) 이벤트의 형식입니다. 자세한 내용은 AWS Health 사용 설명서의 Amazon CloudWatch Events를 사용하여 AWS 상태 이벤트 관리를 참조하십시오.

AWS 상태 이벤트 형식

```
{
  "version": "0",
  "id": "7bf73129-1428-4cd3-a780-95db273d1602",
  "detail-type": "AWS Health Event",
  "source": "aws.health",
  "account": "123456789012",
  "time": "2016-06-05T06:27:57Z",
  "region": "region",
  "resources": [],
  "detail": {
    "eventArn": "arn:aws:health:region::event/id",
    "service": "service",
    "eventTypeCode": "AWS_service_code",
    "eventTypeCategory": "category",
    "startTime": "Sun, 05 Jun 2016 05:01:10 GMT",
    "endTime": "Sun, 05 Jun 2016 05:30:57 GMT",
    "eventDescription": [{
      "language": "lang-code",
      "latestDescription": "description"
    }]
    ...
  }
}
```

eventTypeCategory

이벤트의 범주 코드입니다. 가능한 값은 `issue`, `accountNotification` 및 `scheduledChange` 입니다.

eventTypeCode

이벤트 유형의 고유 식별자입니다. 예를 들면 `AWS_EC2_INSTANCE_NETWORK_MAINTENANCE_SCHEDULED`나

`AWS_EC2_INSTANCE_REBOOT_MAINTENANCE_SCHEDULED`와 같습니다. `MAINTENANCE_SCHEDULED`를 포함하는 이벤트는 보통 `startTime`을 기준으로 약 2주 전에 푸시됩니다.

id

이벤트의 고유 식별자입니다.

service

이벤트에 영향을 받는 AWS 서비스입니다. 예: `EC2`, `S3`, `REDSHIFT`, `RDS` 등.

Elastic Load Balancing API 문제

```
{
  "version": "0",
  "id": "121345678-1234-1234-1234-123456789012",
  "detail-type": "AWS Health Event",
  "source": "aws.health",
  "account": "123456789012",
  "time": "2016-06-05T06:27:57Z",
  "region": "ap-southeast-2",
  "resources": [],
  "detail": {
    "eventArn": "arn:aws:health:ap-southeast-2::event/
AWS_ELASTICLOADBALANCING_API_ISSUE_90353408594353980",
    "service": "ELASTICLOADBALANCING",
    "eventTypeCode": "AWS_ELASTICLOADBALANCING_API_ISSUE",
    "eventTypeCategory": "issue",
    "startTime": "Sat, 11 Jun 2016 05:01:10 GMT",
    "endTime": "Sat, 11 Jun 2016 05:30:57 GMT",
    "eventDescription": [{
      "language": "en_US",
      "latestDescription": "A description of the event will be provided here"
  }
}
```

Amazon EC2 인스턴스 스토어의 드라이브 성능 저하

```
{
  "version": "0",
  "id": "121345678-1234-1234-1234-123456789012",
  "detail-type": "AWS Health Event",
  "source": "aws.health",
  "account": "123456789012",
  "time": "2016-06-05T06:27:57Z",
  "region": "us-west-2",
  "resources": [
    "i-abcd1111"
  ],
  "detail": {
    "eventArn": "arn:aws:health:us-west-2::event/
AWS_EC2_INSTANCE_STORE_DRIVE_PERFORMANCE_DEGRADED_90353408594353980",
    "service": "EC2",
    "eventTypeCode": "AWS_EC2_INSTANCE_STORE_DRIVE_PERFORMANCE_DEGRADED",
    "eventTypeCategory": "issue",
    "startTime": "Sat, 05 Jun 2016 15:10:09 GMT",
    "eventDescription": [{
      "language": "en_US",
      "latestDescription": "A description of the event will be provided here"
    }],
    "affectedEntities": [{
      "entityValue": "i-abcd1111",
```

```
      "tags": {
        "stage": "prod",
        "app": "my-app"
  }
}
```

AWS KMS 이벤트

다음은 AWS Key Management Service(AWS KMS) 이벤트의 예제입니다. 자세한 내용은 AWS Key Management Service Developer Guide의 AWS KMS 이벤트를 참조하십시오.

KMS CMK 로테이션

AWS KMS에서 CMK의 키 자료가 자동으로 로테이션됩니다.

```
{
  "version": "0",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "KMS CMK Rotation",
  "source": "aws.kms",
  "account": "111122223333",
  "time": "2016-08-25T21:05:33Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
  ],
  "detail": {
    "key-id": "1234abcd-12ab-34cd-56ef-1234567890ab"
  }
}
```

KMS가 가져온 키 자료 만료

AWS KMS에서 기간이 만료된 CMK의 키 자료가 삭제됩니다.

```
{
  "version": "0",
  "id": "9da9af57-9253-4406-87cb-7cc400e43465",
  "detail-type": "KMS Imported Key Material Expiration",
  "source": "aws.kms",
  "account": "111122223333",
  "time": "2016-08-22T20:12:19Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
  ],
  "detail": {
    "key-id": "1234abcd-12ab-34cd-56ef-1234567890ab"
  }
}
```

KMS CMK 삭제

AWS KMS가 예약된 CMK 삭제를 완료했습니다.

```
{
  "version": "0",
  "id": "e9ce3425-7d22-412a-a699-e7a5fc3fbc9a",
  "detail-type": "KMS CMK Deletion",
```

```
  "source": "aws.kms",
  "account": "111122223333",
  "time": "2016-08-19T03:23:45Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
  ],
  "detail": {
    "key-id": "1234abcd-12ab-34cd-56ef-1234567890ab"
  }
}
```

Amazon Macie 이벤트

다음은 Amazon Macie 이벤트의 예제입니다.

알림 생성됨

```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "Macie Alert",
  "source": "aws.macie",
  "account": "123456789012",
  "time": "2017-04-24T22:28:49Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id/alert/alert_id",
    "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id"
  ],
  "detail": {
    "notification-type": "ALERT_CREATED",
    "name": "Scanning bucket policies",
    "tags": [
      "Custom_Alert",
      "Insider"
    ],
    "url": "https://lb00.us-east-1.macie.aws.amazon.com/111122223333/posts/alert_id",
    "alert-arn": "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id/alert/alert_id",
    "risk-score": 80,
    "trigger": {
      "rule-arn": "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id",
      "alert-type": "basic",
      "created-at": "2017-01-02 19:54:00.644000",
      "description": "Alerting on failed enumeration of large number of bucket policies",
      "risk": 8
    },
    "created-at": "2017-04-18T00:21:12.059000",
    "actor": "555566667777:assumed-role:superawesome:aroaidpldc7nsesfnheji",
    "summary": {
      "Description": "Alerting on failed enumeration of large number of bucket policies",
      "IP": {
        "34.199.185.34": 121,
        "34.205.153.2": 2,
        "72.21.196.70": 2
      },
      "Time Range": [
        {
          "count": 125,
          "start": "2017-04-24T20:23:49Z",
          "end": "2017-04-24T20:25:54Z"
        }
```

```
      ],
      "Source ARN": "arn:aws:sts::123456789012:assumed-role/RoleName",
      "Record Count": 1,
      "Location": {
        "us-east-1": 125
      },
      "Event Count": 125,
      "Events": {
        "GetBucketLocation": {
          "count": 48,
          "ISP": {
            "Amazon": 48
          }
        },
        "ListRoles": {
          "count": 2,
          "ISP": {
            "Amazon": 2
          }
        },
        "GetBucketPolicy": {
          "count": 37,
          "ISP": {
            "Amazon": 37
          },
          "Error Code": {
            "NoSuchBucketPolicy": 22
          }
        },
        "GetBucketAcl": {
          "count": 37,
          "ISP": {
            "Amazon": 37
          }
        },
        "ListBuckets": {
          "count": 1,
          "ISP": {
            "Amazon": 1
          }
        }
      },
      "recipientAccountId": {
        "123456789012": 125
      }
    }
  }
}
```

```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "Macie Alert",
  "source": "aws.macie",
  "account": "123456789012",
  "time": "2017-04-18T18:15:41Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id/alert/alert_id",
    "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id"
  ],
  "detail": {
    "notification-type": "ALERT_CREATED",
    "name": "Bucket is writable by all authenticated users",
```

```
    "tags": [
      "Custom_Alert",
      "Audit"
    ],
    "url": "https://lb00.us-east-1.macie.aws.amazon.com/111122223333/posts/alert_id",
    "alert-arn": "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id/alert/alert_id",
    "risk-score": 70,
    "trigger": {
      "rule-arn": "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id",
      "alert-type": "basic",
      "created-at": "2017-04-08 00:21:30.749000",
      "description": "Bucket is writable by all authenticated users",
      "risk": 7
    },
    "created-at": "2017-04-18T18:16:17.046454",
    "actor": "444455556666",
    "summary": {
      "Description": "Bucket is writable by all authenticated users",
      "Bucket": {
        "secret-bucket-name": 1
      },
      "Record Count": 1,
      "ACL": {
        "secret-bucket-name": [
          {
            "Owner": {
              "DisplayName": "bucket_owner",
              "ID": "089d2842f4b392f5c5c61f073bd2e4a37b3bb2e62659318c6960e8981648a17e"
            },
            "Grants": [
              {
                "Grantee": {
                  "Type": "Group",
                  "URI": "http://acs.amazonaws.com/groups/global/AuthenticatedUsers"
                },
                "Permission": "WRITE"
              }
            ]
          }
        ]
      },
      "Event Count": 1,
      "Timestamps": {
        "2017-01-10T22:48:06.784937": 1
      }
    }
  }
}
```


알림 업데이트됨


```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "Macie Alert",
  "source": "aws.macie",
  "account": "123456789012",
  "time": "2017-04-18T17:47:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id/alert/alert_id",
    "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id"
  ],
  "detail": {
    "notification-type": "ALERT_UPDATED",
```

```
    "name": "Public bucket contains high risk object",
    "tags": [
      "Custom_Alert",
      "Audit"
    ],
    "url": "https://lb00.us-east-1.macie.aws.amazon.com/111122223333/posts/alert_id",
    "alert-arn": "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id/alert/alert_id",
    "risk-score": 100,
    "trigger": {
     "rule-arn": "arn:aws:macie:us-east-1:123456789012:trigger/trigger_id",
      "alert-type": "basic",
      "created-at": "2017-04-08 00:23:39.138000",
      "description": "Public bucket contains high risk object",
      "risk": 10
    },
    "created-at": "2017-04-08T00:36:26.270000",
    "actor": "public_bucket",
    "summary": {
      "Description": "Public bucket contains high risk object",
      "Object": {
        "public_bucket/secret_key.txt": 1,
        "public_bucket/financial_summary.txt": 1
      },
      "Record Count": 2,
      "Themes": {
        "Secret Markings": 1,
        "Corporate Proposals": 1,
        "Confidential Markings": 1
      },
      "Event Count": 2,
      "DLP risk": {
        "7": 2
      },
      "Owner": {
        "bucket_owner": 2
      },
      "Timestamps": {
        "2017-04-03T16:12:53+00:00": 2
      }
    }
  }
}
```



```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "Macie Alert",
  "source": "aws.macie",
  "account": "123456789012",
  "time": "2017-04-22T03:31:47Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:macie:us-east-1:123456789012:trigger/macie/alert/alert_id",
    "arn:aws:macie:us-east-1:123456789012:trigger/macie"
  ],
  "detail": {
    "notification-type": "ALERT_UPDATED",
    "name": "Lists the instance profiles that have the specified associated IAM role, Lists
 the names of the inline policies that are embedded in the specified IAM role",
    "tags": [
      "Predictive",
      "Behavioral_Anomaly"
    ],
    "url": "https://lb00.us-east-1.macie.aws.amazon.com/111122223333/posts/alert_id",
```

```
    "alert-arn": "arn:aws:macie:us-east-1:123456789012:trigger/macie/alert/alert_id",
    "risk-score": 20,
    "created-at": "2017-04-22T03:08:35.256000",
    "actor": "123456789012:assumed-role:rolename",
    "trigger": {
      "alert-type": "predictive",
      "features": {
        "distinctEventName": {
          "name": "distinctEventName",
          "description": "Event Names executed during a user session",
          "narrative": "A sudden increase in event names utilized by a user can be an
 indicator of a change in user behavior or account risk",
          "risk": 3
        },
        "ListInstanceProfilesForRole": {
          "name": "ListInstanceProfilesForRole",
          "description": "Lists the instance profiles that have the specified associated
 IAM role",
          "narrative": "Information collection activity suggesting the start of a
 reconnaissance or exfiltration campaign",
          "anomalous": true,
          "multiplier": 8.420560747663552,
          "excession_times": [
            "2017-04-21T18:00:00Z"
          ],
          "risk": 1
        },
        "ListRolePolicies": {
          "name": "ListRolePolicies",
          "description": "Lists the names of the inline policies that are embedded in the
 specified IAM role",
          "narrative": "Information collection activity suggesting the start of a
 reconnaisance or exfiltration campaign",
          "anomalous": true,
          "multiplier": 12.017441860465116,
          "excession_times": [
            "2017-04-21T18:00:00Z"
          ],
          "risk": 2
        }
      }
    }
  }
}
```


AWS Management 콘솔 로그인 이벤트

AWS Management 콘솔 로그인 이벤트는 CloudWatch 이벤트에서 감지할 수 있습니다. 리전 로그인 이벤트에 대한 자세한 내용은 리전 로그인 이벤트 로깅을 참조하십시오.

다음은 콘솔 로그인 이벤트의 예제입니다.


```
{
  "id": "6f87d04b-9f74-4f04-a780-7acf4b0a9b38",
  "detail-type": "AWS Console Sign In via CloudTrail",
  "source": "aws.signin",
  "account": "123456789012",
  "time": "2016-01-05T18:21:27Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
      "eventVersion": "1.02",
```

```
      "userIdentity": {
            "type": "Root",
            "principalId": "123456789012",
            "arn": "arn:aws:iam::123456789012:root",
            "accountId": "123456789012"
            },
      "eventTime": "2016-01-05T18:21:27Z",
      "eventSource": "signin.amazonaws.com",
      "eventName": "ConsoleLogin",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "0.0.0.0",
      "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/47.0.2526.106 Safari/537.36",
      "requestParameters": null,
      "responseElements": {
            "ConsoleLogin": "Success"
            },
      "additionalEventData": {
            "LoginTo": "https://console.aws.amazon.com/console/home?state=hashArgs
%23&isauthcode=true",
            "MobileVersion": "No",
            "MFAUsed": "No" },
      "eventID": "324731c0-64b3-4421-b552-dfc3c27df4f6",
      "eventType": "AwsConsoleSignIn"
      }
}
```

AWS OpsWorks 스택 이벤트

다음은 AWS OpsWorks 스택 이벤트의 예제입니다.

AWS OpsWorks 스택 인스턴스 상태 변경

AWS OpsWorks 스택 인스턴스 상태의 변화를 표시합니다. 다음은 인스턴스 상태입니다.

- booting
- connection_lost
- online
- pending
- rebooting
- requested
- running_setup
- setup_failed
- shutting_down
- start_failed
- stopping
- stop_failed
- stopped
- terminating
- terminated

```
{
  "version": "0",
```

```
  "id": "dc5fa8df-48f1-2108-b1b9-1fe5ebcf2296",
  "detail-type": "OpsWorks Instance State Change",
  "source": "aws.opsworks",
  "account": "123456789012",
  "time": "2018-01-25T11:12:23Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:opsworks:us-east-1:123456789012:instance/a648d98f-fdd8-4323-952a-a50z3e4z500z"
  ],
  "detail": {
    "initiated_by": "user",
    "hostname": "testing1",
    "stack-id": "acd3df16-e859-4598-8414-377b12a902da",
    "layer-ids": [
      "d1a0cb7f-c7e9-4a63-811c-976f0267b2c8"
    ],
    "instance-id": "a648d98f-fdd8-4323-952a-a50z3e4z500z",
    "ec2-instance-id": "i-08b1c2b67aa292276",
    "status": "requested"
  }
}
```

`initiated_by` 필드는 인스턴스가 `requested`, `terminating` 또는 `stopping` 상태에 있을 때만 채워집니다. `initiated_by` 필드에는 다음 값 중 하나가 포함될 수 있습니다.

- `user` - 사용자가 API 또는 AWS Management 콘솔을 사용하여 인스턴스 상태 변화를 요청했습니다.
- `auto-scaling` - AWS OpsWorks 스택의 자동 확장 기능에 의해 인스턴스 상태 변화가 시작되었습니다.
- `auto-healing` - AWS OpsWorks 스택의 자동 복구 기능에 의해 인스턴스 상태 변화가 시작되었습니다.

AWS OpsWorks 스택 명령 상태 변경

AWS OpsWorks 스택 명령의 상태에 변화가 생겼습니다. 다음은 명령 상태입니다.

- `expired` - 명령 시간이 초과되었습니다.
- `failed` - 일반적인 명령 오류가 발생했습니다.
- `skipped` - 인스턴스가 AWS OpsWorks 스택에서 Amazon EC2과 다른 상태를 갖기 때문에 명령을 건너뛰었습니다.
- `successful` - 명령이 성공했습니다.
- `superseded` - 이미 적용된 구성 변경이 적용되었기 때문에 명령을 건너뛰었습니다.

```
{
  "version": "0",
  "id": "96c778b6-a40e-c8c1-aafc-c9852a3a7b52",
  "detail-type": "OpsWorks Command State Change",
  "source": "aws.opsworks",
  "account": "123456789012",
  "time": "2018-01-26T08:54:40Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:opsworks:us-east-1:123456789012:instance/a648d98f-fdd8-4323-952a-a50a3e4e500f"
  ],
  "detail": {
    "command-id": "acc9f4f3-a3ec-4fab-b70f-c7d04e71e3ec",
    "instance-id": "a648d98f-fdd8-4323-952a-a50a3e4e500f",
    "type": "setup",
    "status": "successful"
  }
}
```

AWS OpsWorks 스택 배포 상태 변경

AWS OpsWorks 스택 배포의 상태에 변화가 생겼습니다. 다음은 배포 상태입니다.

- `running`
- `successful`
- `failed`

```
{
  "version": "0",
  "id": "b8230afa-60c7-f43f-b632-841c1cfb22ff",
  "detail-type": "OpsWorks Deployment State Change",
  "source": "aws.opsworks",
  "account": "123456789012",
  "time": "2018-01-25T11:15:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:opsworks:us-east-1:123456789012:instance/a648d98f-fdd8-4323-952a-a50a3e4e500f"
  ],
  "detail": {
    "duration": 16,
    "stack-id": "acd3df16-e859-4598-8414-377b12a902da",
    "instance-ids": [
      "a648d98f-fdd8-4323-952a-a50a3e4e500f"
    ],
    "deployment-id": "606419dc-418e-489c-8531-bff9770fc346",
    "command": "configure",
    "status": "successful"
  }
}
```

`duration` 필드는 배포가 완료될 때만 채워지며 시간을 초 단위로 표시합니다.

AWS OpsWorks 스택 경보

AWS OpsWorks 스택 서비스 오류가 발생했습니다.

```
{
  "version": "0",
  "id": "f99faa6f-0e27-e398-95bb-8f190806d275",
  "detail-type": "OpsWorks Alert",
  "source": "aws.opsworks",
  "account": "123456789012",
  "time": "2018-01-20T16:51:29Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "stack-id": "2f48f2be-ac7d-4dd5-80bb-88375f94db7b",
    "instance-id": "986efb74-69e8-4c6d-878e-5b77c054cbb0",
    "type": "InstanceStop",
    "message": "The shutdown of the instance timed out. Please try stopping it again."
  }
}
```

SageMaker 이벤트

다음은 SageMaker 이벤트의 예제입니다.

SageMaker 학습 작업 상태 변경

SageMaker 학습 작업의 상태 변화를 표시합니다.

`TrainingJobStatus`의 값이 `Failed`인 경우 이벤트에 `FailureReason` 필드가 포함되며, 이 필드는 학습 작업이 실패한 이유에 대한 설명을 제공합니다.

```
{
    "version": "0",
    "id": "844e2571-85d4-695f-b930-0153b71dcb42",
    "detail-type": "SageMaker Training Job State Change",
    "source": "aws.sagemaker",
    "account": "123456789012",
    "time": "2018-10-06T12:26:13Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:sagemaker:us-east-1:123456789012:training-job/kmeans-1"
    ],
    "detail": {
        "TrainingJobName": "89c96cc8-dded-4739-afcc-6f1dc936701d",
        "TrainingJobArn": "arn:aws:sagemaker:us-east-1:123456789012:training-job/kmeans-1",
        "TrainingJobStatus": "Completed",
        "SecondaryStatus": "Completed",
        "HyperParameters": {
            "Hyper": "Parameters"
        },
        "AlgorithmSpecification": {
            "TrainingImage": "TrainingImage",
            "TrainingInputMode": "TrainingInputMode"
        },
        "RoleArn": "a little teapot, some little teapot",
        "InputDataConfig": [
            {
                "ChannelName": "Train",
                "DataSource": {
                    "S3DataSource": {
                        "S3DataType": "S3DataType",
                        "S3Uri": "S3Uri",
                        "S3DataDistributionType": "S3DataDistributionType"
                    }
                },
                "ContentType": "ContentType",
                "CompressionType": "CompressionType",
                "RecordWrapperType": "RecordWrapperType"
            }
        ],
        "OutputDataConfig": {
            "KmsKeyId": "KmsKeyId",
            "S3OutputPath": "S3OutputPath"
        },
        "ResourceConfig": {
            "InstanceType": "InstanceType",
            "InstanceCount": 3,
            "VolumeSizeInGB": 20,
            "VolumeKmsKeyId": "VolumeKmsKeyId"
        },
        "VpcConfig": {

        },
        "StoppingCondition": {
            "MaxRuntimeInSeconds": 60
        },
        "CreationTime": "2018-10-06T12:26:13Z",
        "TrainingStartTime": "2018-10-06T12:26:13Z",
        "TrainingEndTime": "2018-10-06T12:26:13Z",
        "LastModifiedTime": "2018-10-06T12:26:13Z",
        "SecondaryStatusTransitions": [
```

```
        ],
        "Tags": {

        }
    }
}
```

SageMaker HyperParameter 튜닝 작업 상태 변경

SageMaker 하이퍼파라미터 튜닝 작업의 상태 변화를 표시합니다.

```
{
  "version": "0",
  "id": "844e2571-85d4-695f-b930-0153b71dcb42",
  "detail-type": "SageMaker HyperParameter Tuning Job State Change",
  "source": "aws.sagemaker",
  "account": "123456789012",
  "time": "2018-10-06T12:26:13Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:sagemaker:us-east-1:123456789012:tuningJob/x"
  ],
  "detail": {
    "HyperParameterTuningJobName": "016bffd3-6d71-4d3a-9710-0a332b2759fc",
    "HyperParameterTuningJobArn": "arn:aws:sagemaker:us-east-1:123456789012:tuningJob/x",
    "TrainingJobDefinition": {
      "StaticHyperParameters": {},
      "AlgorithmSpecification": {
        "TrainingImage": "trainingImageName",
        "TrainingInputMode": "inputModeFile",
        "MetricDefinitions": [
          {
            "Name": "metricName",
            "Regex": "regex"
          }
        ]
      },
      "RoleArn": "roleArn",
      "InputDataConfig": [
        {
          "ChannelName": "channelName",
          "DataSource": {
            "S3DataSource": {
              "S3DataType": "s3DataType",
              "S3Uri": "s3Uri",
              "S3DataDistributionType": "s3DistributionType"
            }
          },
          "ContentType": "contentType",
          "CompressionType": "gz",
          "RecordWrapperType": "RecordWrapper"
        }
      ],
      "VpcConfig": {
        "SecurityGroupIds": [
          "securityGroupIds"
        ],
        "Subnets": [
          "subnets"
        ]
      },
      "OutputDataConfig": {
        "KmsKeyId": "kmsKeyId",
        "S3OutputPath": "s3OutputPath"
```

```
      },
      "ResourceConfig": {
        "InstanceType": "instanceType",
        "InstanceCount": 10,
        "VolumeSizeInGB": 500,
        "VolumeKmsKeyId": "volumeKeyId"
      },
      "StoppingCondition": {
        "MaxRuntimeInSeconds": 3600
      }
    },
    "HyperParameterTuningJobStatus": "status",
    "CreationTime": "2018-10-06T12:26:13Z",
    "LastModifiedTime": "2018-10-06T12:26:13Z",
    "TrainingJobStatusCounters": {
      "Completed": 1,
      "InProgress": 0,
      "RetryableError": 0,
      "NonRetryableError": 0,
      "Stopped": 0
    },
    "ObjectiveStatusCounters": {
      "Succeeded": 1,
      "Pending": 0,
      "Failed": 0
    },
    "Tags": {}
  }
}
```

SageMaker 변환 작업 상태 변경

SageMaker 배치(batch) 변환 작업의 상태 변화를 표시합니다.

`TransformJobStatus`의 값이 `Failed`인 경우 이벤트에 `FailureReason` 필드가 포함되며, 이 필드는 학습 작업이 실패한 이유에 대한 설명을 제공합니다.

```
{
    "version": "0",
    "id": "844e2571-85d4-695f-b930-0153b71dcb42",
    "detail-type": "SageMaker Transform Job State Change",
    "source": "aws.sagemaker",
    "account": "123456789012",
    "time": "2018-10-06T12:26:13Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:sagemaker:us-east-1:123456789012:transform-job/myjob"
    ],
    "detail": {
        "TransformJobName": "4b52bd8f-e034-4345-818d-884bdd7c9724",
        "TransformJobArn": "arn:aws:sagemaker:us-east-1:123456789012:transform-job/myjob",
        "TransformJobStatus": "Completed",
        "ModelName": "ModelName",
        "MaxConcurrentTransforms": 5,
        "MaxPayloadInMB": 10,
        "BatchStrategy": "Strategy",
        "Environment": {
            "environment1": "environment2"
        },
        "TransformInput": {
            "DataSource": {
                "S3DataSource": {
                    "S3DataType": "s3DataType",
                    "S3Uri": "s3Uri"
```

```
                }
            },
            "ContentType": "content type",
            "CompressionType": "compression type",
            "SplitType": "split type"
        },
        "TransformOutput": {
            "S3OutputPath": "s3Uri",
            "Accept": "accept",
            "AssembleWith": "assemblyType",
            "KmsKeyId": "kmsKeyId"
        },
        "TransformResources": {
            "InstanceType": "instanceType",
            "InstanceCount": 3
        },
        "CreationTime": "2018-10-06T12:26:13Z",
        "TransformStartTime": "2018-10-06T12:26:13Z",
        "TransformEndTime": "2018-10-06T12:26:13Z",
        "Tags": {

        }
    }
}
```

AWS Security Hub 이벤트

예제 Security Hub 이벤트에 대한 내용은 AWS Security Hub 사용 설명서의 Amazon CloudWatch Events를 이용한 AWS Security Hub 모니터링을 참조하십시오.

AWS Server Migration Service 이벤트

다음은 AWS Server Migration Service 이벤트의 예제입니다.

복제 작업 삭제 알림

```
{
    "version": "0",
    "id": "5630992d-92cd-439f-f2a8-92c8212aee24",
    "detail-type": "Server Migration Job State Change",
    "source": "aws.sms",
    "account": "123456789012",
    "time": "2018-02-07T22:30:11Z",
    "region": "us-west-1",
    "resources": [
        "arn:aws:sms:us-west-1:123456789012:sms-job-21a64348"
    ],
    "detail": {
        "state": "Deleted",
        "replication-run-id": "N/A",
        "replication-job-id": "sms-job-21a64348",
        "version": "1.0"
    }
}
```

복제 작업 완료 알림

```
{
    "version": "0",
```

```
    "id": "3f9c59cc-f941-522a-be6d-f08e44ff1715",
    "detail-type": "Server Migration Job State Change",
    "source": "aws.sms",
    "account": "123456789012",
    "time": "2018-02-07T22:54:00Z",
    "region": "us-west-1",
    "resources": [
        "arn:aws:sms:us-west-1:123456789012:sms-job-2ea64347",
        "arn:aws:sms:us-west-1:123456789012:sms-job-2ea64347/sms-run-e1a64388"
    ],
    "detail": {
        "state": "Completed",
        "replication-run-id": "sms-run-e1a64388",
        "replication-job-id": "sms-job-2ea64347",
        "ami-id": "ami-746d6314",
        "version": "1.0"
    }
}
```

AWS 시스템 관리자 이벤트

다음은 AWS 시스템 관리자 이벤트의 예제입니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 Run 명령의 명령 실행 상태 변경 기록을 참조하십시오.

Run Command 상태 변경 알림

```
{
    "version": "0",
    "id": "51c0891d-0e34-45b1-83d6-95db273d1602",
    "detail-type": "EC2 Command Status-change Notification",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2016-07-10T21:51:32Z",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1:123456789012:instance/i-abcd1111"],
    "detail": {
        "command-id": "e8d3c0e4-71f7-4491-898f-c9b35bee5f3b",
        "document-name": "AWS-RunPowerShellScript",
        "expire-after": "2016-07-14T22:01:30.049Z",
        "parameters": {
            "executionTimeout": ["3600"],
            "commands": ["date"]
        },
        "requested-date-time": "2016-07-10T21:51:30.049Z",
        "status": "Success"
    }
}
```

Run Command 호출 상태 변경 알림

```
{
    "version": "0",
    "id": "4780e1b8-f56b-4de5-95f2-95db273d1602",
    "detail-type": "EC2 Command Invocation Status-change Notification",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2016-07-10T21:51:32Z",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1:123456789012:instance/i-abcd1111"],
    "detail": {
        "command-id": "e8d3c0e4-71f7-4491-898f-c9b35bee5f3b",
```

```
        "document-name": "AWS-RunPowerShellScript",
        "instance-id": "i-9bb89e2b",
        "requested-date-time": "2016-07-10T21:51:30.049Z",
        "status": "Success"
    }
}
```

자동화 단계 상태 변경 알림

```
{
  "version": "0",
  "id": "eeca120b-a321-433e-9635-dab369006a6b",
  "detail-type": "EC2 Automation Step Status-change Notification",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2016-11-29T19:43:35Z",
  "region": "us-east-1",
  "resources": ["arn:aws:ssm:us-east-1:123456789012:automation-
execution/333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "arn:aws:ssm:us-east-1:123456789012:automation-definition/runcommand1:1"],
  "detail": {
    "ExecutionId": "333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "Definition": "runcommand1",
    "DefinitionVersion": 1.0,
    "Status": "Success",
    "EndTime": "Nov 29, 2016 7:43:25 PM",
    "StartTime": "Nov 29, 2016 7:43:23 PM",
    "Time": 2630.0,
    "StepName": "runFixedCmds",
    "Action": "aws:runCommand"
  }
}
```

자동화 실행 상태 변경 알림

```
{
  "version": "0",
  "id": "d290ece9-1088-4383-9df6-cd5b4ac42b99",
  "detail-type": "EC2 Automation Execution Status-change Notification",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2016-11-29T19:43:35Z",
  "region": "us-east-1",
  "resources": ["arn:aws:ssm:us-east-1:123456789012:automation-
execution/333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "arn:aws:ssm:us-east-1:123456789012:automation-definition/runcommand1:1"],
  "detail": {
    "ExecutionId": "333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "Definition": "runcommand1",
    "DefinitionVersion": 1.0,
    "Status": "Success",
    "StartTime": "Nov 29, 2016 7:43:20 PM",
    "EndTime": "Nov 29, 2016 7:43:26 PM",
    "Time": 5753.0,
    "ExecutedBy": "arn:aws:iam::123456789012:user/userName"
  }
}
```

State Manager 연결 상태 변경

```
{
   "version":"0",
```

```
   "id":"db839caf-6f6c-40af-9a48-25b2ae2b7774",
   "detail-type":"EC2 State Manager Association State Change",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2017-05-16T23:01:10Z",
   "region":"us-west-1",
   "resources":[
      "arn:aws:ssm:us-west-1::document/AWS-RunPowerShellScript"
   ],
   "detail":{
      "association-id":"6e37940a-23ba-4ab0-9b96-5d0a1a05464f",
      "document-name":"AWS-RunPowerShellScript",
      "association-version":"1",
      "document-version":"Optional.empty",
      "targets":"[{\"key\":\"InstanceIds\",\"values\":[\"i-12345678\"]}]",
      "creation-date":"2017-02-13T17:22:54.458Z",
      "last-successful-execution-date":"2017-05-16T23:00:01Z",
      "last-execution-date":"2017-05-16T23:00:01Z",
      "last-updated-date":"2017-02-13T17:22:54.458Z",
      "status":"Success",
      "association-status-aggregated-count":"{\"Success\":1}",
      "schedule-expression":"cron(0 */30 * * * ? *)",
      "association-cwe-version":"1.0"
   }
}
```


State Manager 인스턴스 연결 상태 변경


```
{
   "version":"0",
   "id":"6a7e8feb-b491-4cf7-a9f1-bf3703467718",
   "detail-type":"EC2 State Manager Instance Association State Change",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2017-02-23T15:23:48Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ec2:us-east-1:123456789012:instance/i-12345678",
      "arn:aws:ssm:us-east-1:123456789012:document/my-custom-document"
   ],
   "detail":{
      "association-id":"34fcb7e0-9a14-4984-9989-0e04e3f60bd8",
      "instance-id":"i-12345678",
      "document-name":"my-custom-document",
      "document-version":"1",
      "targets":"[{\"key\":\"instanceids\",\"values\":[\"i-12345678\"]}]",
      "creation-date":"2017-02-23T15:23:48Z",
      "last-successful-execution-date":"2017-02-23T16:23:48Z",
      "last-execution-date":"2017-02-23T16:23:48Z",
      "status":"Success",
      "detailed-status":"",
      "error-code":"testErrorCode",
      "execution-summary":"testExecutionSummary",
      "output-url":"sampleurl",
      "instance-association-cwe-version":"1"
   }
}
```


AWS 시스템 관리자 구성 규정 준수 이벤트

다음은 Amazon EC2 Systems Manager(SSM) 구성 규정 준수 이벤트의 예제입니다.

연결 규정 준수

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:03:26Z",
  "region": "us-west-1",
  "resources": [
    "arn:aws:ssm:us-west-1:461348341421:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "last-runtime": "2017-01-01T10:10:10Z",
    "compliance-status": "compliant",
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-type": "Association"
  }
}
```

연결 규정 미준수

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:02:31Z",
  "region": "us-west-1",
  "resources": [
    "arn:aws:ssm:us-west-1:461348341421:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "last-runtime": "2017-01-01T10:10:10Z",
    "compliance-status": "non_compliant",
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-type": "Association"
  }
}
```

패치 규정 준수

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:03:26Z",
  "region": "us-west-1",
  "resources": [
    "arn:aws:ssm:us-west-1:461348341421:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-status": "compliant",
    "compliance-type": "Patch",
    "patch-baseline-id": "PB789",
```

```
    "severity": "critical"
  }
}
```

패치 규정 미준수

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:02:31Z",
  "region": "us-west-1",
  "resources": [
    "arn:aws:ssm:us-west-1:461348341421:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-status": "non_compliant",
    "compliance-type": "Patch",
    "patch-baseline-id": "PB789",
    "severity": "critical"
  }
}
```

AWS 시스템 관리자 유지 관리 Windows가 설치된 이벤트

다음은 시스템 관리자 유지 관리 Windows가 설치된 이벤트의 예제입니다.

대상 등록

상태는 `DEREGISTERED`가 될 수도 있습니다.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Target Registration Notification",
   "source":"aws.ssm",
   "account":"012345678901",
   "time":"2016-11-16T00:58:37Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ssm:us-west-2:001312665065:maintenancewindow/mw-0ed7251d3fcf6e0c2",
      "arn:aws:ssm:us-west-2:001312665065:windowtarget/
e7265f13-3cc5-4f2f-97a9-7d3ca86c32a6"
   ],
   "detail":{
      "window-target-id":"e7265f13-3cc5-4f2f-97a9-7d3ca86c32a6",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "status":"REGISTERED"
   }
}
```

Window 실행 유형

기타 가능한 상태는 PENDING, IN_PROGRESS, SUCCESS, FAILED, TIMED_OUT, SKIPPED_OVERLAPPING입니다.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Execution State-change Notification",
   "source":"aws.ssm",
   "account":"012345678901",
   "time":"2016-11-16T01:00:57Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ssm:us-west-2:0123456789ab:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "start-time":"2016-11-16T01:00:56.427Z",
      "end-time":"2016-11-16T01:00:57.070Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"b60fb56e-776c-4e5c-84ee-123456789012",
      "status":"TIMED_OUT"
   }
}
```

작업 실행 유형

기타 가능한 상태는 IN_PROGRESS, SUCCESS, FAILED, TIMED_OUT입니다.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Task Execution State-change Notification",
   "source":"aws.ssm",
   "account":"012345678901",
   "time":"2016-11-16T01:00:56Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ssm:us-west-2:0123456789ab:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "start-time":"2016-11-16T01:00:56.759Z",
      "task-execution-id":"6417e808-7f35-4d1a-843f-123456789012",
      "end-time":"2016-11-16T01:00:56.847Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"b60fb56e-776c-4e5c-84ee-123456789012",
      "status":"TIMED_OUT"
   }
}
```

처리된 작업 대상

기타 가능한 상태는 IN_PROGRESS, SUCCESS, FAILED, TIMED_OUT입니다.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Task Target Invocation State-change Notification",
   "source":"aws.ssm",
   "account":"012345678901",
   "time":"2016-11-16T01:00:57Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ssm:us-west-2:0123456789ab:maintenancewindow/mw-123456789012345678"
```

```
    ],
    "detail":{
       "start-time":"2016-11-16T01:00:56.427Z",
       "end-time":"2016-11-16T01:00:57.070Z",
       "window-id":"mw-0ed7251d3fcf6e0c2",
       "window-execution-id":"b60fb56e-776c-4e5c-84ee-123456789012",
       "task-execution-id":"6417e808-7f35-4d1a-843f-123456789012",
       "window-target-id":"e7265f13-3cc5-4f2f-97a9-123456789012",
       "status":"TIMED_OUT",
       "owner-information":"Owner"
    }
}
```

기간 상태 변경

가능한 상태는 `ENABLED`와 `DISABLED`입니다.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window State-change Notification",
   "source":"aws.ssm",
   "account":"012345678901",
   "time":"2016-11-16T00:58:37Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ssm:us-west-2:0123456789ab:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "window-id":"mw-123456789012",
      "status":"DISABLED"
   }
}
```

AWS 시스템 관리자 Parameter Store 이벤트

다음은 Amazon EC2 Systems Manager(SSM) Parameter Store 이벤트의 예제입니다.

파라미터 생성

```
{
  "version": "0",
  "id": "6a7e4feb-b491-4cf7-a9f1-bf3703497718",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-05-22T16:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-east-1:123456789012:parameter/foo"
  ],
  "detail": {
    "operation": "Create",
    "name": "foo",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```

파라미터 업데이트

```
{
  "version": "0",
  "id": "9547ef2d-3b7e-4057-b6cb-5fdf09ee7c8f",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-05-22T16:44:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-east-1:123456789012:parameter/foo"
  ],
  "detail": {
    "operation": "Update",
    "name": "foo",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```

파라미터 삭제

```
{
  "version": "0",
  "id": "80e9b391-6a9b-413c-839a-453b528053af",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-05-22T16:45:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-east-1:123456789012:parameter/foo"
  ],
  "detail": {
    "operation": "Delete",
    "name": "foo",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```

AWS Step Functions 이벤트

Step Functions 샘플 이벤트는 AWS Step Functions 개발자 안내서의 Step Functions 이벤트 예제를 참조하십시오.

AWS 리소스의 태그 변경 이벤트

다음은 태그 이벤트의 예입니다.

```
{
    "version": "0",
    "id": "ffd8a6fe-32f8-ef66-c85c-111111111111",
    "detail-type": "Tag Change on Resource",
    "source": "aws.tag",
    "account": "123456789012",
```

```
    "time": "2018-09-18T20:41:06Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:instance/i-0000000aaaaaaaaaa"
    ],
    "detail": {
        "changed-tag-keys": [
            "key2",
            "key3"
        ],
        "service": "ec2",
        "resource-type": "instance",
        "version": 5,
        "tags": {
            "key4": "value4",
            "key1": "value1",
            "key2": "value2"
        }
    }
}
```

AWS Trusted Advisor 이벤트

다음은 AWS Trusted Advisor 이벤트의 예제입니다. 자세한 내용은 AWS Support User Guide의 Amazon CloudWatch Events를 사용하여 Trusted Advisor 검사 결과 모니터링을 참조하십시오.

낮은 사용률의 Amazon EC2 인스턴스

```
{
  "version": "0",
  "id": "1234abcd-ab12-123a-123a-1234567890ab",
  "detail-type": "Trusted Advisor Check Item Refresh Notification",
  "source": "aws.trustedadvisor",
  "account": "123456789012",
  "time": "2018-01-12T20:07:49Z",
  "region": "us-east-2",
  "resources": [],
  "detail": {
    "check-name": "Low Utilization Amazon EC2 Instances",
    "check-item-detail": {
      "Day 1": "0.1%  0.00MB",
      "Day 2": "0.1%  0.00MB",
      "Day 3": "0.1%  0.00MB",
      "Region/AZ": "ca-central-1a",
      "Estimated Monthly Savings": "$9.22",
      "14-Day Average CPU Utilization": "0.1%",
      "Day 14": "0.1%  0.00MB",
      "Day 13": "0.1%  0.00MB",
      "Day 12": "0.1%  0.00MB",
      "Day 11": "0.1%  0.00MB",
      "Day 10": "0.1%  0.00MB",
      "14-Day Average Network I/O": "0.00MB",
      "Number of Days Low Utilization": "14 days",
      "Instance Type": "t2.micro",
      "Instance ID": "i-01234567890abcdef",
      "Day 8": "0.1%  0.00MB",
      "Instance Name": null,
      "Day 9": "0.1%  0.00MB",
      "Day 4": "0.1%  0.00MB",
      "Day 5": "0.1%  0.00MB",
      "Day 6": "0.1%  0.00MB",
      "Day 7": "0.1%  0.00MB"
```

```
    },
    "status": "WARN",
    "resource_id": "arn:aws:ec2:ca-central-1:123456789012:instance/i-01234567890abcdef",
    "uuid": "aa12345f-55c7-498e-b7ac-123456789012"
  }
}
```

Load Balancer 최적화

```
{
  "version": "0",
  "id": "1234abcd-ab12-123a-123a-1234567890ab",
  "detail-type": "Trusted Advisor Check Item Refresh Notification",
  "source": "aws.trustedadvisor",
  "account": "123456789012",
  "time": "2018-01-12T20:07:03Z",
  "region": "us-east-2",
  "resources": [],
  "detail": {
    "check-name": "Load Balancer Optimization ",
    "check-item-detail": {
      "Instances in Zone a": "1",
      "Status": "Yellow",
      "Instances in Zone b": "0",
      "# of Zones": "2",
      "Region": "eu-central-1",
      "Load Balancer Name": "my-load-balance",
      "Instances in Zone e": null,
      "Instances in Zone c": null,
      "Reason": "Single AZ",
      "Instances in Zone d": null
    },
    "status": "WARN",
    "resource_id": "arn:aws:elasticloadbalancing:eu-central-1:123456789012:loadbalancer/my-
load-balancer",
    "uuid": "aa12345f-55c7-498e-b7ac-123456789012"
  }
}
```

노출된 액세스 키

```
{
  "version": "0",
  "id": "1234abcd-ab12-123a-123a-1234567890ab",
  "detail-type": "Trusted Advisor Check Item Refresh Notification",
  "source": "aws.trustedadvisor",
  "account": "123456789012",
  "time": "2018-01-12T19:38:24Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "check-name": "Exposed Access Keys",
    "check-item-detail": {
      "Case ID": "12345678-1234-1234-abcd-1234567890ab",
      "Usage (USD per Day)": "0",
      "User Name (IAM or Root)": "my-username",
      "Deadline": "1440453299248",
      "Access Key ID": "AKIAIOSFODNN7EXAMPLE",
      "Time Updated": "1440021299248",
      "Fraud Type": "Exposed",
      "Location": "www.example.com"
    },
    "status": "ERROR",
```

```
    "resource_id": "",
    "uuid": "aa12345f-55c7-498e-b7ac-123456789012"
  }
}
```

Amazon WorkSpaces 이벤트

Amazon WorkSpaces 이벤트에 대한 내용은 Amazon WorkSpaces Administration Guide의 CloudWatch 이벤트를 사용하여 WorkSpaces 모니터링을 참조하십시오.

CloudTrail을 통해 전달된 이벤트

이벤트를 출력하지 않고 이 페이지에 열거되지 않은 서비스에도 EventBridge를 사용할 수 있습니다. AWS CloudTrail은 AWS API 호출과 같은 이벤트를 자동으로 기록하는 서비스입니다. CloudTrail에서 캡처한 정보에서 트리거되는 EventBridge 규칙을 생성할 수 있습니다. CloudTrail에 대한 자세한 내용은 AWS CloudTrail이란 무엇입니까? 단원을 참조하십시오. CloudTrail을 사용하는 EventBridge 규칙 생성에 대한 자세한 내용은 AWS CloudTrail을 사용하여 AWS API 호출에서 트리거되는 EventBridge 규칙 생성 (p. 6) 단원을 참조하십시오.

CloudTrail을 통해 전달되는 모든 이벤트는 `detail-type`의 값으로 `AWS API Call via CloudTrail`을 보유합니다.

AWS의 일부 발생은 EventBridge에 서비스 자체와 CloudTrail 둘 다로 보고될 수 있지만 방식은 서로 다릅니다. 예를 들어, 인스턴스를 시작하거나 종료하는 Amazon EC2 API 호출은 CloudTrail을 통해 EventBridge에 사용 가능한 이벤트를 생성합니다. 그러나 Amazon EC2 인스턴스 상태 변경(예: '실행 중'에서 '종료 중'으로)은 EventBridge 이벤트 자체가 변경되는 것입니다.

다음은 CloudTrail을 통해 전달된 이벤트의 예입니다. 이 이벤트는 Amazon S3에 대한 AWS API 호출에 의해 생성되어 버킷을 생성합니다.

```
{
    "version": "0",
    "id": "36eb8523-97d0-4518-b33d-ee3579ff19f0",
    "detail-type": "AWS API Call via CloudTrail",
    "source": "aws.s3",
    "account": "123456789012",
    "time": "2016-02-20T01:09:13Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
        "eventVersion": "1.03",
        "userIdentity": {
            "type": "Root",
            "principalId": "123456789012",
            "arn": "arn:aws:iam::123456789012:root",
            "accountId": "123456789012",
            "sessionContext": {
                "attributes": {
                    "mfaAuthenticated": "false",
                    "creationDate": "2016-02-20T01:05:59Z"
                }
            }
        },
        "eventTime": "2016-02-20T01:09:13Z",
        "eventSource": "s3.amazonaws.com",
        "eventName": "CreateBucket",
        "awsRegion": "us-east-1",
```

```
        "sourceIPAddress": "100.100.100.100",
        "userAgent": "[S3Console/0.4]",
        "requestParameters": {
            "bucketName": "bucket-test-iad"
        },
        "responseElements": null,
        "requestID": "9D767BCC3B4E7487",
        "eventID": "24ba271e-d595-4e66-a7fd-9c16cbf8abae",
        "eventType": "AwsApiCall"
    }
}
```


규모가 256KB보다 큰 AWS API 호출 이벤트는 지원되지 않습니다. 규칙의 트리거로 사용할 수 있는 API 호출에 대한 자세한 정보는 CloudTrail 이벤트 기록에서 지원하는 서비스를 참조하십시오.

AWS 계정 간 이벤트 전송 및 수신

AWS 계정이 다른 AWS 계정으로 이벤트를 전송하거나, 혹은 다른 계정의 이벤트를 수신하도록 설정할 수 있습니다. 이 설정은 계정이 동일한 조직에 속해 있거나 파트너 등의 관계를 갖는 조직에 속해 있는 경우 유용할 수 있습니다.

계정에서 이벤트를 전송 또는 수신하도록 설정하는 경우 이벤트를 전송하거나 수신할 수 있는 AWS 계정을 각각 지정합니다. AWS Organizations 기능을 사용하면 조직을 지정하여 해당 조직에 속한 모든 계정에 액세스하는 권한을 부여할 수 있습니다. 자세한 내용은 AWS Organizations 사용 설명서의 AWS Organizations란 무엇입니까? 단원을 참조하십시오.

전체 프로세스는 다음과 같습니다.

- 수신자 계정에서 지정된 AWS 계정, 조직 또는 모든 AWS 계정이 이벤트를 수신자 계정으로 전송할 수 있도록 기본 이벤트 버스에 대한 권한을 편집합니다.
- 발신자 계정에서 수신자 계정의 이벤트 버스를 대상으로 갖는 하나 이상의 규칙을 설정합니다.

  발신자 계정이 권한을 가지고 있는 AWS 조직에 속하여 이벤트를 전송할 수 있는 권한이 있다면 발신자 계정 역시 이벤트를 수신자 계정으로 전송할 수 있는 정책이 적용되는 IAM 역할을 가지고 있어야 합니다. AWS Management 콘솔을 사용하여 수신자 계정을 대상으로 하는 규칙을 생성할 경우 이러한 역할이 자동으로 생성됩니다. AWS CLI를 사용하는 경우에는 역할을 수동으로 생성해야 합니다.
- 수신자 계정에서 발신자 계정이 전송하는 이벤트를 일치시키는 하나 이상의 규칙을 설정합니다.

수신자 계정이 이벤트 버스에 권한을 추가하는 AWS 리전은 발신자 계정이 수신자 계정으로 이벤트를 전송하기 위해 규칙을 생성하는 리전과 동일해야 합니다.

한 계정에서 다른 계정으로 전송되는 이벤트에 대해서는 전송 계정에서 사용자 지정 이벤트로 요금이 부과됩니다. 수신 계정에는 요금이 부과되지 않습니다. 자세한 내용은 Amazon EventBridge 요금을 참조하십시오.

수신자 계정은 발신자 계정에서 수신되는 이벤트를 제3의 계정으로 전송하는 규칙을 설정하는 경우 이러한 이벤트는 제3의 계정으로 전송되지 않습니다.

AWS 계정이 다른 AWS 계정에서 이벤트를 수신하도록 설정

다른 계정 또는 조직으로부터 이벤트를 수신하려면 먼저 계정의 기본 이벤트 버스에서 권한을 편집합니다. 기본 이벤트 버스는 AWS 서비스, 권한이 부여된 다른 AWS 계정 및 `PutEvents` 호출이 전송하는 이벤트를 수락합니다.

기본 이벤트 버스에서 다른 AWS 계정에 권한을 부여하는 권한을 편집할 때 계정 ID 또는 조직 ID를 기준으로 계정을 지정할 수 있습니다. 또는 모든 AWS 계정으로부터 이벤트를 수신하도록 선택할 수도 있습니다.

> Warning
>
> 모든 AWS 계정으로부터 이벤트를 수신하도록 선택할 경우 다른 계정으로부터 수신하는 이벤트만 일치시키는 규칙을 생성하도록 주의하십시오. 더욱 안전한 규칙을 생성하려면 이벤트를 수신할 계정 하나 이상의 계정 ID가 입력되는 `Account` 필드가 각 규칙의 모든 패턴에 포함되어야 합니다. 이벤트 패턴에 계정 필드가 포함되는 규칙은 `Account` 필드에 나열되어 있지 않은 계정에서 전송된 이벤트와 일치하지 않습니다. 자세한 내용은 EventBridge의 이벤트 및 이벤트 패턴 (p. 36) 단원을 참조하십시오.

콘솔을 사용하여 계정이 다른 AWS 계정으로부터 이벤트를 수신하도록 설정하려면

1. https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 이벤트 버스, 권한 추가를 선택합니다.
3. AWS 계정 또는 조직을 선택합니다.

   AWS 계정을 선택할 경우에는 이벤트를 수신할 계정의 12자리 AWS 계정 ID를 입력합니다. 다른 모든 AWS 계정에서 이벤트를 수신하려면 Everybody(*)를 선택합니다.

   조직을 선택할 경우에는 내 조직을 선택하여 현재 계정이 속한 조직의 모든 계정에게 권한을 부여합니다. 혹은 다른 조직을 선택한 후 해당 조직의 조직 ID를 입력합니다. 조직 ID를 입력할 때는 접두사로 `o-`를 추가해야 합니다.
4. 추가를 선택합니다.
5. 이러한 단계를 반복하여 다른 계정이나 조직을 추가할 수 있습니다.

AWS CLI를 사용하여 계정이 다른 AWS 계정으로부터 이벤트를 수신하도록 설정하려면

1. 특정 AWS 계정이 이벤트를 전송하도록 설정하려면 다음 명령을 실행합니다.

```
aws events put-permission --action events:PutEvents --statement-id MySid --
principal SenderAccountID
```

   AWS 조직이 이벤트를 전송하도록 설정하려면 다음 명령을 실행합니다.

```
aws events put-permission --action events:PutEvents --statement-id MySid
                  --principal \* --condition '{"Type" : "StringEquals", "Key":
 "aws:PrincipalOrgID", "Value": "SenderOrganizationID"}'
```

   다른 모든 AWS 계정이 이벤트를 전송하도록 설정하려면 다음 명령을 실행합니다.

```
aws events put-permission --action events:PutEvents --statement-id MySid --principal \*
```

   `aws events put-permission`을 여러 차례 실행하여 권한을 AWS 계정 및 조직에게 개별적으로 부여할 수는 있지만 각 계정과 조직을 모두 한 번의 명령으로 지정하지는 못합니다.
2. 기본 이벤트 버스의 권한을 설정한 후 선택적으로 `describe-event-bus` 명령을 사용하여 권한을 확인할 수 있습니다.

```
aws events describe-event-bus
```

다른 AWS 계정으로 이벤트 전송

이벤트를 다른 계정에게 전송하려면 다른 AWS 계정의 기본 이벤트 버스를 대상으로 하는 EventBridge 규칙을 구성하십시오. 해당 수신 계정의 이벤트 버스 역시 사용자의 계정으로부터 이벤트를 수신하도록 구성되어야 합니다.

콘솔을 사용하여 계정이 다른 AWS 계정으로 이벤트를 전송하도록 설정하려면

1. https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 이벤트와 Create Rule(규칙 만들기)을 선택합니다.
3. 이벤트 소스에서 이벤트 패턴을 선택한 후 다른 계정에 전송할 서비스 이름 및 이벤트 유형을 선택합니다.

4. 대상 추가를 선택합니다.
5. 대상에서 다른 AWS 계정의 이벤트 버스를 선택합니다. 계정 ID에서 이벤트를 전송할 AWS 계정의 12자리 계정 ID를 입력합니다.
6. 수신자 계정이 전체 조직에게 권한을 부여하여 발신자 계정에게 이벤트를 전송할 권한이 있을 때는 IAM 역할이 필요합니다.

   - IAM 역할을 자동으로 생성하려면 이 특정 리소스에 대해 새 역할 생성을 선택합니다.
   - 그렇지 않으면 기존 역할 사용을 선택합니다. 빌드를 호출할 수 있는 충분한 권한을 이미 가지고 있는 역할을 선택하십시오. EventBridge는 선택한 역할에 추가 권한을 부여하지 않습니다.
7. 페이지 하단에서 세부 정보 구성을 선택합니다.
8. 규칙의 이름과 설명을 입력하고 규칙 생성을 선택합니다.

AWS CLI를 사용하여 다른 AWS 계정으로 이벤트를 전송하려면

1. 발신자 계정이 수신자 계정이 권한을 부여한 AWS 조직에 속하여 이벤트를 전송할 수 있는 권한이 있다면 발신자 계정 역시 이벤트를 수신자 계정으로 전송할 수 있는 정책이 적용되는 역할을 가지고 있어야 합니다. 이번 단계에서는 역할을 생성하는 방법에 대해서 설명합니다.

   발신자 계정에게 조직이 아닌 AWS 계정 ID을 통한 이벤트 전송 권한이 부여되었다면 이번 단계는 선택 사항입니다. 2단계로 건너뛸 수 있습니다.

   a. 발신자 계정이 조직을 통해 권한을 부여 받았다면 필요한 IAM 역할을 생성해야 합니다. 먼저 다음과 같은 내용과 함께 `assume-role-policy-document.json`이라는 이름의 파일을 생성합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "events.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

   b. 역할을 생성하려면 다음과 같이 명령을 입력합니다.

```
$ aws iam create-role \
--profile sender \
--role-name event-delivery-role \
--assume-role-policy-document file://assume-role-policy-document.json
```

   c. 다음 콘텐츠가 포함된 `permission-policy.json`이라는 파일을 생성합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "events:PutEvents"
      ],
      "Resource": [
        "arn:aws:events:us-east-1:${receiver_account_id}:event-bus/default"
      ]
    }
  ]
```

```
}
```

d. 다음 명령을 입력하여 이 정책을 역할에 연결합니다.

```
$ aws iam put-role-policy \
--profile sender \
--role-name event-delivery-role \
--policy-name EventBusDeliveryRolePolicy
--policy-document file://permission-policy.json
```

2. `put-rule` 명령을 사용하여 다른 계정에 전송할 이벤트 유형과 일치하는 규칙을 생성합니다.
3. 다른 계정의 기본 이벤트 버스를 규칙의 대상으로 추가합니다.

   발신자 계정에게 계정 ID를 통해 이벤트를 전송할 수 있는 권한이 부여되었다면 역할을 지정할 필요 없습니다. 다음 명령을 실행합니다.

```
aws events put-targets --rule NameOfRuleMatchingEventsToSend --targets
 "Id"="MyId","Arn"="arn:aws:events:region:$ReceiverAccountID:event-bus/default"
```

   발신자 계정에게 조직을 통해 이벤트를 전송할 수 있는 권한이 부여되었다면 다음 예제와 같이 역할을 지정합니다.

```
aws events put-targets --rule NameOfRuleMatchingEventsToSend --targets
 "Id"="MyId","Arn"="arn:aws:events:region:$ReceiverAccountID:event-bus/
default","RoleArn"="arn:aws:iam:${sender_account_id}:role/event-delivery-role"
```

다른 AWS 계정의 이벤트를 일치시키는 규칙 작성

계정을 다른 AWS 계정으로부터 이벤트를 수신하도록 설정한 경우 해당 이벤트를 일치시키는 규칙을 작성합니다. 다른 계정에서 수신할 이벤트에 일치하도록 규칙의 이벤트 패턴을 설정합니다.

규칙의 이벤트 패턴에 `account`를 지정하지 않을 경우 다른 계정으로부터 수신하는 이벤트를 일치시키는 계정의 모든 규칙(신규 및 기존)이 해당 이벤트를 기준으로 트리거됩니다. 다른 계정에서 이벤트를 수신할 때 자체 계정에서 생성된 이벤트 패턴에서만 규칙이 트리거되도록 하려면 규칙의 이벤트 패턴에 `account`를 추가하고 자체 계정 ID를 지정해야 합니다.

모든 AWS 계정으로부터 이벤트를 수신하도록 AWS 계정을 설정하는 경우 계정의 모든 EventBridge 규칙에 `account`를 추가할 것을 강력히 권장합니다. 이는 알 수 없는 AWS 계정의 이벤트에서 계정의 규칙이 트리거되는 것을 방지합니다. 규칙에 `account` 필드를 지정할 때 AWS 계정 2개 이상의 계정 ID를 필드에 지정할 수 있습니다.

권한이 부여된 AWS 계정에서 일치하는 이벤트에서 규칙을 트리거하려면 이 규칙의 `account` 필드에 * 를 지정하지 마십시오. 이벤트의 `account` 파일에는 *가 절대 나타나지 않으므로 그렇게 하면 모든 이벤트가 일치하지 않을 것입니다. 대신 규칙에서 `account` 필드만 생략하십시오.

콘솔을 사용하여 다른 계정의 이벤트를 일치시키는 규칙을 작성하려면

1. https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 이벤트와 Create Rule(규칙 만들기)을 선택합니다.
3. 이벤트 소스에 대해 이벤트 패턴을 선택하고 규칙에 일치할 서비스 이름 및 이벤트 유형을 선택합니다.
4. 이벤트 패턴 미리 보기 옆에서 편집을 선택합니다.
5. 편집 창에서 이벤트를 전송하는 AWS 계정이 규칙에 따라 일치할 수 있도록 `Account` 라인을 추가합니다. 예를 들어, 편집 창이 원래 다음과 같이 표시됩니다.

```
{
```

```
  "source": [
    "aws.ec2"
  ],
  "detail-type": [
    "EBS Volume Notification"
  ]
}
```

다음을 추가하여 규칙이 AWS 계정 123456789012 및 111122223333가 전송하는 EBS 볼륨 알림을 일치시키도록 합니다.

```
{
  "account": [
    "123456789012","111122223333"
  ],
  "source": [
    "aws.ec2"
  ],
  "detail-type": [
    "EBS Volume Notification"
  ]
}
```

6. 이벤트 패턴을 편집한 후 저장을 선택합니다.
7. 계정에서 하나 이상의 대상을 설정하여 규칙 생성을 마칩니다.

AWS CLI를 사용하여 다른 AWS 계정의 이벤트를 일치시키는 규칙을 작성하려면

- `put-rule` 명령을 사용합니다. 규칙의 이벤트 패턴에 있는 `Account` 필드에서 규칙에 따라 일치해야 하는 나머지 AWS 계정을 지정합니다. 다음 규칙 예제는 AWS 계정 123456789012 및 111122223333의 Amazon EC2 인스턴스 상태 변경을 일치시킵니다.

```
aws events put-rule --name "EC2InstanceStateChanges" --event-pattern "{\"account\":
["123456789012", "111122223333"],\"source\":[\"aws.ec2\"],\"detail-type\":[\"EC2
 Instance State-change Notification\"]}"  --role-arn "arn:aws:iam::123456789012:role/
MyRoleForThisRule"
```

AWS Organizations를 사용하기 위해 발신자-수신자 관계 마이그레이션

계정 ID에 직접 부여된 권한이 있는 발신자 계정이 있는데, 이제 이러한 권한을 취소하고 조직에 권한을 부여하여 보내기 계정 액세스 권한을 부여하려는 경우 몇 가지 추가 단계를 수행해야 합니다. 다음 단계에서는 발신자 계정의 이벤트를 계속해서 수신자 계정으로 받을 수 있도록 합니다. 이는 조직을 통해 이벤트를 보내는 권한이 부여된 계정 역시 이벤트를 보내는 데 IAM 역할을 사용해야 하기 때문입니다.

발신자-수신자 관계를 마이그레이션하는 데 필요한 권한을 추가하려면

1. 발신자 계정에서 이벤트를 수신자 계정으로 보낼 수 있도록 하는 정책을 사용해 IAM 역할을 생성합니다.

   a. 다음 콘텐츠가 포함된 `assume-role-policy-document.json`이라는 파일을 생성합니다.

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

```
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "events.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

b. IAM 역할을 생성하려면 다음 명령을 입력합니다.

```
$ aws iam create-role \
--profile sender \
--role-name event-delivery-role \
--assume-role-policy-document file://assume-role-policy-document.json
```

c. 다음 콘텐츠가 포함된 `permission-policy.json`이라는 파일을 생성합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "events:PutEvents"
      ],
      "Resource": [
        "arn:aws:events:us-east-1:${receiver_account_id}:event-bus/default"
      ]
    }
  ]
}
```

d. 다음 명령을 입력하여 이 정책을 역할에 연결합니다.

```
$ aws iam put-role-policy \
--profile sender \
--role-name event-delivery-role \
--policy-name EventBusDeliveryRolePolicy
--policy-document file://permission-policy.json
```

2. 발신자 계정에서 수신자 계정의 기본 이벤트 버스를 대상으로 갖는 기존의 각 규칙을 편집합니다. 1단계에서 생성한 역할을 대상 정보에 추가하여 규칙을 편집합니다. 다음 명령을 사용합니다.

```
aws events put-targets --rule Rulename --targets
 "Id"="MyID","Arn"="arn:aws:events:region:$ReceiverAccountID:event-bus/
default","RoleArn"="arn:aws:iam:${sender_account_id}:role/event-delivery-role"
```

3. 수신자 계정에서 다음 명령을 실행하여 조직의 계정에 수신자 계정으로 이벤트를 보내는 권한을 부여합니다.

```
aws events put-permission --action events:PutEvents --statement-id Sid-For-Organization
 --principal \* --condition '{"Type" : "StringEquals", "Key": "aws:PrincipalOrgID",
 "Value": "SenderOrganizationID"}'
```

경우에 따라 발신자 계정에 원래 직접 부여된 권한을 취소할 수도 있습니다.

```
aws events remove-permission --statement-id Sid-for-SenderAccount
```

PutEvents를 통한 이벤트 추가

`PutEvents` 작업은 단일 요청으로 EventBridge에 여러 개의 이벤트를 전송합니다. 자세한 내용은 Amazon EventBridge API 참조의 PutEvents 및 AWS CLI Command Reference의 put-events를 참조하십시오.

각 `PutEvents` 요청은 제한된 수의 항목을 지원할 수 있습니다. 자세한 내용은 Amazon EventBridge 할당량 (p. 154) 단원을 참조하십시오. `PutEvents` 작업은 요청의 일반 순서에 따라 모든 항목들을 처리하고자 시도합니다. 각 이벤트는 `PutEvents` 호출 이후에 EventBridge가 할당한 고유 ID를 가지고 있습니다.

다음 예제에서는 Java 코드가 EventBridge에 동일한 이벤트를 두 개 전송합니다.

AWS SDK for Java Version 2.x

```
EventBridgeClient eventBridgeClient =
    EventBridgeClient.builder().build();

PutEventsRequestEntry requestEntry = PutEventsRequestEntry.builder()
    .resources("resource1", "resource2")
    .source("com.mycompany.myapp")
    .detailType("myDetailType")
    .detail("{ \"key1\": \"value1\", \"key2\": \"value2\" }")
    .build();

List & lt;
PutEventsRequestEntry > requestEntries = new ArrayList & lt;
PutEventsRequestEntry > ();
requestEntries.add(requestEntry);

PutEventsRequest eventsRequest = PutEventsRequest.builder()
    .entries(requestEntries)
    .build();

PutEventsResponse result = eventBridgeClient.putEvents(eventsRequest);

for (PutEventsResultEntry resultEntry: result.entries()) {
    if (resultEntry.eventId() != null) {
        System.out.println("Event Id: " + resultEntry.eventId());
    } else {
        System.out.println("PutEvents failed with Error Code: " +
 resultEntry.errorCode());
    }
}
```

AWS SDK for Java Version 1.0

```
PutEventsRequestEntry requestEntry = new PutEventsRequestEntry()
        .withTime(new Date())
        .withSource("com.mycompany.myapp")
        .withDetailType("myDetailType")
        .withResources("resource1", "resource2")
        .withDetail("{ \"key1\": \"value1\", \"key2\": \"value2\" }");

PutEventsRequest request = new PutEventsRequest()
        .withEntries(requestEntry, requestEntry);

PutEventsResult result = awsEventsClient.putEvents(request);

for (PutEventsResultEntry resultEntry : result.getEntries()) {
```

```
    if (resultEntry.getEventId() != null) {
        System.out.println("Event Id: " + resultEntry.getEventId());
    } else {
        System.out.println("Injection failed with Error Code: " +
 resultEntry.getErrorCode());
    }
}
```

`PutEvents` 결과에는 응답 항목 어레이가 포함되어 있습니다. 응답 어레이의 각 항목은 요청 및 응답의 일반 순서(위에서 아래로)를 이용해 요청 어레이의 항목과 직접 연관이 됩니다. 응답의 `Entries` 어레이에는 항상 요청 어레이와 같은 수의 항목이 포함됩니다.

PutEvents 사용 시 처리 실패

기본적으로 요청 내의 개별 항목이 실패해도 요청의 후속 항목들의 처리가 중단되지 않습니다. 따라서 응답 항목 어레이에는 처리에 성공한 항목과 실패한 항목이 모두 포함됩니다. 따라서 처리에 실패한 항목들을 찾아서 후속 호출에 이를 포함시켜야 합니다.

성공한 결과 항목에는 ID 값이 포함되고 실패한 결과 항목에는 `ErrorCode` 및 `ErrorMessage` 값이 포함됩니다. `ErrorCode` 파라미터는 오류의 유형을 반영합니다. `ErrorMessage`는 오류에 대한 세부 정보를 제공합니다. 아래 예제에서는 하나의 `PutEvents` 요청에 대해 3개의 결과 항목이 있습니다. 처리에 실패한 두 번째 항목이 응답에 반영됩니다.

예제: PutEvents 응답 구문

```
{
    "FailedEntryCount": 1,
    "Entries": [
        {
            "EventId": "11710aed-b79e-4468-a20b-bb3c0c3b4860"
        },
        {   "ErrorCode": "InternalFailure",
            "ErrorMessage": "Internal Service Failure"
        },
            "EventId": "d804d26a-88db-4b66-9eaf-9a11c708ae82"
        }
    ]
}
```

처리에 실패한 항목들을 후속 `PutEvents` 요청에 포함시킬 수 있습니다. 먼저, `PutEventsResult`에서 `FailedRecordCount` 파라미터를 확인해서 요청에 처리에 실패한 레코드가 있는지 확인합니다. 만약 있을 경우 null이 아닌 `ErrorCode`를 가진 각 `Entry`를 후속 요청에 추가해야 합니다. 이러한 유형의 핸들러 예제는 다음 코드를 참조하십시오.

예제: PutEvents 실패 핸들러

```
PutEventsRequestEntry requestEntry = new PutEventsRequestEntry()
        .withTime(new Date())
        .withSource("com.mycompany.myapp")
        .withDetailType("myDetailType")
        .withResources("resource1", "resource2")
        .withDetail("{ \"key1\": \"value1\", \"key2\": \"value2\" }");

List<PutEventsRequestEntry> putEventsRequestEntryList = new ArrayList<>();
for (int i = 0; i < 3; i++) {
    putEventsRequestEntryList.add(requestEntry);
}
```

```
PutEventsRequest putEventsRequest = new PutEventsRequest();
putEventsRequest.withEntries(putEventsRequestEntryList);
PutEventsResult putEventsResult = awsEventsClient.putEvents(putEventsRequest);

while (putEventsResult.getFailedEntryCount() > 0) {
    final List<PutEventsRequestEntry> failedEntriesList = new ArrayList<>();
    final List<PutEventsResultEntry> PutEventsResultEntryList =
 putEventsResult.getEntries();
    for (int i = 0; i < PutEventsResultEntryList.size(); i++) {
        final PutEventsRequestEntry putEventsRequestEntry =
 putEventsRequestEntryList.get(i);
        final PutEventsResultEntry putEventsResultEntry = PutEventsResultEntryList.get(i);
        if (putEventsResultEntry.getErrorCode() != null) {
            failedEntriesList.add(putEventsRequestEntry);
        }
    }
    putEventsRequestEntryList = failedEntriesList;
    putEventsRequest.setEntries(putEventsRequestEntryList);
    putEventsResult = awsEventsClient.putEvents(putEventsRequest);
    }
```

AWS CLI를 사용하여 이벤트 전송

AWS CLI를 사용하여 사용자 지정 이벤트를 전송할 수 있습니다. 다음 예제는 EventBridge로 하나의 사용자 지정 이벤트를 입력합니다.

```
aws events put-events \
--entries '[{"Time": "2016-01-14T01:02:03Z", "Source": "com.mycompany.myapp", "Resources":
 ["resource1", "resource2"], "DetailType": "myDetailType", "Detail": "{ \"key1\":
 \"value1\", \"key2\": \"value2\" }"}]'
```

예를 들어 entries.json이라는 파일을 다음과 같이 생성할 수도 있습니다.

```
[
  {
    "Time": "2016-01-14T01:02:03Z",
    "Source": "com.mycompany.myapp",
    "Resources": [
      "resource1",
      "resource2"
    ],
    "DetailType": "myDetailType",
    "Detail": "{ \"key1\": \"value1\", \"key2\": \"value2\" }"
  }
]
```

AWS CLI를 사용하여 이 파일에서 항목을 읽고 이벤트를 전송할 수 있습니다. 명령 프롬프트에서 다음과 같이 입력합니다.

```
aws events put-events --entries file://entries.json
```

PutEvents 이벤트 항목 크기 계산

`PutEvents` 작업을 사용하여 EventBridge로 사용자 지정 이벤트를 주입할 수 있습니다. 전체 항목 크기가 256KB를 넘지 않는 한 `PutEvents` 작업을 사용하여 여러 개의 이벤트를 주입할 수 있습니다. 아래 단계를

수행하여 이벤트 항목 크기를 미리 계산할 수 있습니다. 그런 다음 효율성을 위해 여러 개의 이벤트 항목을 하나의 요청으로 일괄 처리할 수 있습니다.

> **Note**
>
> 항목에는 크기 제한이 있습니다. 항목이 제한 크기보다 작다고 해서 EventBridge의 이벤트도 이 크기보다 작다는 의미는 아닙니다. JSON 형식의 문자와 키로 이벤트를 표현해야 하므로 이벤트 크기가 항상 항목 크기보다 큽니다. 자세한 내용은 EventBridge의 이벤트 및 이벤트 패턴 (p. 36) 단원을 참조하십시오.

`PutEventsRequestEntry` 크기는 다음과 같이 계산됩니다.

- `Time` 파라미터가 지정되어 있는 경우에는 14바이트로 측정됩니다.
- `Source` 및 `DetailType` 파라미터는 UTF-8 인코딩 형식에서 바이트 수로 측정됩니다.
- `Detail` 파라미터가 지정되어 있는 경우에는 각 항목이 UTF-8 인코딩 형식에서 바이트 수로 측정이 됩니다.
- `Resources` 파라미터가 지정되어 있는 경우에는 각 항목이 UTF-8 인코딩 형식에서 바이트 수로 측정이 됩니다.

다음의 Java 코드 예제는 지정된 `PutEventsRequestEntry` 개체의 크기를 계산합니다.

```
int getSize(PutEventsRequestEntry entry) {
    int size = 0;
    if (entry.getTime() != null) {
        size += 14;
    }
    size += entry.getSource().getBytes(StandardCharsets.UTF_8).length;
    size += entry.getDetailType().getBytes(StandardCharsets.UTF_8).length;
    if (entry.getDetail() != null) {
        size += entry.getDetail().getBytes(StandardCharsets.UTF_8).length;
    }
    if (entry.getResources() != null) {
        for (String resource : entry.getResources()) {
            if (resource != null) {
                size += resource.getBytes(StandardCharsets.UTF_8).length;
            }
        }
    }
    return size;
}
```

인터페이스 VPC 엔드포인트와 함께 EventBridge 사용

Amazon Virtual Private Cloud(Amazon VPC)를 사용하여 AWS 리소스를 호스팅하는 경우, VPC와 EventBridge 간에 프라이빗 연결을 설정할 수 있습니다. 이 연결을 사용하면 EventBridge가 퍼블릭 인터넷을 통하지 않고 VPC의 리소스와 통신하게 할 수 있습니다.

Amazon VPC란 사용자가 정의한 가상 네트워크에서 AWS 리소스를 시작할 때 사용할 수 있는 AWS 서비스입니다. VPC가 있으면 IP 주소 범위, 서브넷, 라우팅 테이블, 네트워크 게이트웨이 등 네트워크 설정을 제어할 수 있습니다. VPC를 EventBridge에 연결하려면 EventBridge에 대해 인터페이스 VPC 엔드포인트를 정의하십시오. 이 유형의 엔드포인트를 사용하여 VPC를 AWS 서비스에 연결할 수 있습니다. 이 엔드포인트를 이용하면 인터넷 게이트웨이나 NAT(네트워크 주소 변환) 인스턴스 또는 VPN 연결 없이도 EventBridge에 안정적이고 확장 가능하게 연결됩니다. 자세한 내용은 Amazon VPC 사용 설명서의 Amazon VPC란 무엇입니까? 단원을 참조하십시오.

인터페이스 VPC 엔드포인트는 프라이빗 IP 주소와 함께 탄력적 네트워크 인터페이스를 사용하여 AWS 서비스 간 프라이빗 통신을 사용할 수 있는 AWS 기술인 AWS PrivateLink에 의해 구동됩니다. 자세한 내용은 새 기능 - AWS 서비스를 위한 AWS PrivateLink 단원을 참조하십시오.

다음은 Amazon VPC 사용자를 위한 단계들입니다. 자세한 내용은 시작하기(출처: Amazon VPC 사용 설명서)를 참조하십시오.

가용성

현재 EventBridge가 VPC 엔드포인트를 지원하는 리전은 다음과 같습니다.

- 미국 동부(오하이오)
- 미국 동부(버지니아 북부)
- 미국 서부(캘리포니아 북부 지역)
- 미국 서부(오레곤)
- 아시아 태평양(뭄바이)
- 아시아 태평양(서울)
- 아시아 태평양(싱가포르)
- 아시아 태평양(시드니)
- 아시아 태평양(도쿄)
- 캐나다(중부)
- 유럽(프랑크푸르트)
- 유럽(아일랜드)
- 유럽(런던)
- 유럽(파리)
- 남아메리카(상파울루)

EventBridge에 대한 VPC 엔드포인트를 생성합니다.

VPC에서 EventBridge를 사용하기 시작하려면 EventBridge에 대한 인터페이스 VPC 엔드포인트를 생성합니다. 선택할 서비스 이름은 com.amazonaws.*Region*.events입니다. 자세한 내용은 Amazon VPC 사용 설명서의 인터페이스 엔드포인트 생성을 참조하십시오.

EventBridge에 대해 설정을 변경할 필요가 없습니다. EventBridge는 퍼블릭 엔드포인트 또는 프라이빗 인터페이스 VPC 엔드포인트 중 사용 중인 엔드포인트를 사용하여 다른 AWS 서비스를 호출합니다. 예를 들어, EventBridge용 인터페이스 VPC 엔드포인트를 생성할 때, 규칙이 작동하면 Amazon SNS에 알림을 전송하는 EventBridge 규칙이 이미 있는 경우에는 이 알림이 인터페이스 VPC 엔드포인트를 통해 흐르기 시작합니다.

CloudWatch 지표를 통한 사용량 모니터링

EventBridge는 1분마다 Amazon CloudWatch로 지표를 전송합니다.

EventBridge 지표

`AWS/Events` 네임스페이스에는 다음 지표가 포함되어 있습니다.

이 모든 측정치는 Count를 단위로 사용하므로 Sum과 SampleCount는 가장 유용한 통계가 아닙니다.

지표	설명
`DeadLetterInvocations`	이벤트에 대한 응답으로 규칙 대상이 호출되지 않은 횟수를 측정합니다. 여기에는 동일한 규칙을 다시 트리거하여 무한 루프를 초래하는 호출이 포함됩니다. 유효한 차원: RuleName 단위: 개수
`Invocations`	이벤트에 대한 응답으로 규칙 대상이 호출된 횟수를 측정합니다. 여기에는 성공한 호출과 실패한 호출이 모두 포함되지만, 병목 현상에 걸리거나 재시도하여 계속해서 실패한 호출은 포함되지 않습니다. DeadLetterInvocations는 포함되지 않습니다. Note EventBridge는 0이 아닌 값일 때만 이 지표를 CloudWatch에게 전송합니다. 유효한 차원: RuleName 단위: 개수
`FailedInvocations`	계속해서 실패한 호출 수를 측정합니다. 여기에는 재시도하였거나, 재시도 이후 성공한 호출은 포함되지 않습니다. 또한 DeadLetterInvocations 계산에 포함된 실패 호출은 계산되지 않습니다. 유효한 차원: RuleName 단위: 개수
`TriggeredRules`	임의 이벤트와 일치하여 트리거된 규칙 수를 측정합니다. 유효한 차원: RuleName 단위: 개수
`MatchedEvents`	임의 규칙과 일치한 이벤트 수를 측정합니다. 유효한 차원: 없음 단위: 개수

지표	설명
`ThrottledRules`	병목 현상에 걸려 트리거된 규칙 수를 측정합니다. 유효한 차원: RuleName 단위: 개수

EventBridge 지표 차원

EventBridge 지표는 다음과 같은 차원을 하나 가집니다.

차원	설명
`RuleName`	규칙 이름을 기준으로 사용할 수 있는 지표를 필터링합니다.

Amazon EventBridge 관리형 규칙

다른 AWS 서비스는 AWS 계정에서 해당 서비스의 특정 기능에 필요한 관리형 EventBridge 규칙을 생성 및 관리할 수 있습니다. 이를 관리형 정책이라고 합니다.

서비스가 관리형 규칙을 생성한 경우 해당 서비스에 규칙을 생성하는 권한을 부여하는 IAM 정책을 생성할 수도 있습니다. 이러한 방식으로 생성된 IAM 정책은 필수 규칙만 생성하도록 허용하는 리소스 수준 권한을 통해 범위가 좁아집니다.

관리형 규칙은 강제 삭제 옵션을 사용하여 삭제할 수 있습니다. 다른 서비스에서 해당 규칙이 더 이상 필요하지 않다고 확신하는 경우에만 삭제하십시오. 그렇지 않은 경우 관리형 규칙을 삭제하면 해당 규칙을 사용하는 기능이 작동하지 않게 됩니다.

Amazon EventBridge의 보안

이 단원에서는 Amazon EventBridge 보안 및 인증에 대해 설명합니다.

주제

Amazon EventBridge는 IAM을 사용하여 다른 AWS 서비스 및 리소스에 대한 액세스를 제어합니다. IAM의 작동 방식에 대한 개요를 보려면 IAM 사용 설명서의 액세스 관리 개요를 참조하십시오. 보안 자격 증명에 대한 개요를 보려면 Amazon Web Services 일반 참조의 AWS 보안 자격 증명을 참조하십시오.

Amazon EventBridge의 데이터 보호

Amazon EventBridge는 AWS 공동 책임 모델을 준수하며, 여기에는 데이터 보호 관련 규정 및 지침이 포함됩니다. AWS는 모든 AWS 서비스를 실행하는 글로벌 인프라를 보호할 책임을 갖습니다. AWS는 고객 콘텐츠 및 개인 데이터의 처리를 위한 보안 구성 제어 등 이 인프라에서 호스팅되는 데이터에 대한 제어권을 유지합니다. 데이터 컨트롤러 또는 데이터 프로세서의 역할을 담당하는 AWS 고객과 APN 파트너는 AWS 클라우드에 올린 모든 개인 데이터에 대한 책임을 갖습니다.

데이터 보호를 위해 AWS 계정 자격 증명을 보호하고 AWS Identity and Access Management (IAM)을 사용하여 개별 사용자 계정을 설정하여 각 사용자에게 직무를 수행하는 데 필요한 권한만 부여하는 것이 좋습니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정마다 멀티 팩터 인증(MFA)을 사용합니다.
- SSL/TLS를 사용하여 AWS 리소스와 통신합니다.
- AWS CloudTrail로 API 및 사용자 활동 로깅을 설정합니다.
- AWS 암호화 솔루션을 AWS 서비스 내의 모든 기본 보안 컨트롤과 함께 사용합니다.
- Amazon S3에 저장된 개인 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용합니다.

이름 필드와 같은 자유 형식 필드에 고객 계정 번호와 같은 중요 식별 정보를 절대 입력하지 마십시오. 여기에는 EventBridge 또는 기타 AWS 서비스에서 콘솔, API, AWS CLI 또는 AWS SDK를 사용하여 작업하는 경우가 포함됩니다. EventBridge 또는 기타 서비스에 입력하는 모든 데이터는 진단 로그에 포함되도록 선택할 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 자격 증명 정보를 URL에 포함시키지 마십시오.

데이터 보호에 대한 자세한 내용은 AWS 보안 블로그에서 AWS 공동 책임 모델 및 GDPR 블로그 게시물을 참조하십시오.

저장 데이터 암호화

이벤트의 페이로드는 EventBridge에 내부적으로 저장됩니다. 이 내부 데이터는 암호화되지 않습니다.

전송 중 데이터 암호화

EventBridge와 기타 서비스 간에 전달되는 모든 데이터는 TLS(전송 계층 보안)를 사용하여 암호화됩니다.

태그 기반 정책

Amazon EventBridge는 태그를 기반으로 하는 정책을 지원합니다. 예를 들어 `environment` 키 및 `production` 값이 있는 태그가 포함된 리소스에 대한 액세스를 제한할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "events:PutRule",
                "events:DescribeRule",
                "events:DeleteRule",
                "events:CreateEventBus",
                "events:DescribeEventBus"
                "events:DeleteEventBus"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {"aws:ResourceTag/environment": "production"}
            }
        }
    ]
}
```

이 `Deny` 정책은 `environment/production` 태그가 지정된 리소스에 대한 태그, 규칙 또는 이벤트 버스를 생성, 삭제 또는 수정할 수 있습니다.

태그 지정에 대한 자세한 내용은 다음을 참조하십시오.

- Amazon EventBridge 리소스에 태그 지정 (p. 152)
- IAM 태그를 사용한 액세스 제어

Amazon EventBridge의 자격 증명 및 액세스 관리

Amazon EventBridge에 액세스하려면 AWS가 요청을 인증하는 데 사용할 수 있는 자격 증명이 필요합니다. 이 자격 증명에는 다른 AWS 리소스에서의 이벤트 데이터 검색과 같이 AWS 리소스에 액세스할 수 있는 권한이 포함되어야 합니다. 다음 단원에서는 AWS Identity and Access Management(IAM) 및 EventBridge를 사용하여 리소스에 액세스할 수 있는 대상을 제어하여 리소스를 보호하는 방법에 대해 자세히 설명합니다.

- 인증 (p. 121)
- 액세스 제어 (p. 122)

인증

다음과 같은 ID 유형으로 AWS에 액세스할 수 있습니다.

- AWS 계정 루트 사용자 – AWS에 가입할 때 계정과 연결된 이메일 주소 및 암호를 제공합니다. 이 두 가지가 루트 자격 증명이며 모든 AWS 리소스에 대한 전체 액세스 권한을 제공합니다.

  Important

  보안상 관리자, 즉 계정에 대한 전체 권한이 있는 IAM 사용자를 만들 때에만 루트 자격 증명을 사용하는 것이 좋습니다. 그러면 이 관리자를 사용하여 제한된 권한이 있는 다른 IAM 사용자 및 역할을 만들 수 있습니다. 자세한 내용은 IAM 사용 설명서의 IAM 모범 사례 및 관리자 및 그룹 만들기를 참조하십시오.
- IAM 사용자 – IAM 사용자는 특정 사용자 지정 권한이 있는 사용자 계정 내의 자격 증명입니다(예: EventBridge에서 이벤트 데이터를 대상에 전송할 권한). IAM 사용자 이름과 암호를 사용하여 AWS Management 콘솔, AWS 토론 포럼 또는 AWS Support Center 같은 AWS 웹 페이지에 로그인할 수 있습니다.

  사용자 이름과 암호 외에도 각 사용자에 대해 액세스 키를 생성할 수 있습니다. 여러 SDK 중 하나를 통해 또는 AWS Command Line Interface(AWS CLI)를 사용하여 AWS 서비스에 프로그래밍 방식으로 액세스할 때 이러한 키를 사용할 수 있습니다. SDK 및 AWS CLI 도구는 액세스 키를 사용하여 암호화 방식으로 요청에 서명합니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. EventBridge supports는 인바운드 API 요청을 인증하기 위한 프로토콜인 서명 버전 4를 지원합니다. 요청 인증에 대한 자세한 내용은 Amazon Web Services 일반 참조의 서명 버전 4 서명 프로세스를 참조하십시오.
- IAM 역할 – IAM 역할은 계정에 만들 수 있는 특정 권한이 있는 또 다른 IAM 자격 증명입니다. 이 역할은 IAM 사용자와 유사하지만 특정 개인과 연결되지 않습니다. IAM 역할을 사용하면 AWS 서비스와 리소스에 액세스하는 데 사용할 수 있는 임시 액세스 키를 얻을 수 있습니다. 임시 자격 증명을 가진 IAM 역할은 다음과 같은 상황에서 유용합니다.

  - 연합된 사용자 액세스 – IAM 사용자를 만드는 대신 AWS Directory Service, 엔터프라이즈 사용자 디렉터리 또는 웹 자격 증명 공급자(IdP)의 기존 자격 증명을 사용할 수 있습니다. 이 사용자를 연합된 사용자라고 합니다. AWS에서는 자격 증명 공급자를 통해 액세스가 요청되면 연합된 사용자에게 역할을 할당합니다. 연합된 사용자에 대한 자세한 내용은 IAM 사용 설명서의 연합된 사용자 및 역할을 참조하십시오.
  - 교차 계정 액세스 – 계정의 IAM 역할을 사용하여 다른 계정에 계정 리소스에 액세스할 권한을 부여할 수 있습니다. 예를 보려면 IAM 사용 설명서의 자습서: IAM 역할을 사용한 AWS 계정 간 액세스 권한 위임 단원을 참조하십시오.
  - AWS 제품 액세스 – 계정의 IAM 역할을 사용하여 AWS 제품에 계정의 리소스에 액세스할 권한을 부여할 수 있습니다. 예를 들어 Amazon Redshift에서 자동으로 Amazon S3 버킷에 액세스하도록 허용하는 역할을 만든 후 버킷에 저장된 데이터를 Amazon Redshift 클러스터에 로드할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 역할을 만들어 AWS 서비스에 권한 위임을 참조하십시오.
  - Amazon EC2에서 실행되는 애플리케이션 – EC2 인스턴스에서 실행되고 AWS API 요청을 하는 애플리케이션에서 사용할 수 있도록 EC2 인스턴스 내에 액세스 키를 저장하는 대신에, IAM 역할을 사용하여 이러한 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있도록 인스턴스에 연결된 인스턴스 프로파일을 만들 수 있습니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행되는 프로그램이 임시 자격 증명

을 얻을 수 있습니다. 자세한 내용은 IAM 사용 설명서의 Amazon EC2에서 실행되는 애플리케이션의 역할 사용하기 단원을 참조하십시오.

액세스 제어

요청을 인증할 수 있는 유효한 자격 증명이 있더라도 권한이 없으면 EventBridge 리소스를 생성하거나 액세스할 수 없습니다. 예를 들어 EventBridge 규칙과 연관된 AWS Lambda, Amazon Simple Notification Service(Amazon SNS) 및 Amazon Simple Queue Service(Amazon SQS) 대상을 호출할 수 있는 권한이 있어야 합니다.

다음 단원에서는 EventBridge에 대한 권한을 관리하는 방법을 설명합니다. 먼저 개요를 읽어 보면 도움이 됩니다.

EventBridge 리소스에 대한 액세스 권한 관리 개요

모든 AWS 리소스는 AWS 계정의 소유이고, 리소스 생성 또는 리소스 액세스 권한은 권한 정책에 따라 결정됩니다. 계정 관리자는 IAM 자격 증명(즉, 사용자, 그룹, 역할)에 권한 정책을 연결할 수 있고, 일부 서비스(예: AWS Lambda)에서는 리소스에 대한 권한 정책 연결도 지원합니다.

> Note
>
> 계정 관리자 또는 관리자 IAM 사용자는 관리자 권한이 있는 사용자입니다. 자세한 내용은 IAM 사용 설명서에서 IAM 모범 사례를 참조하십시오.

권한을 부여하려면 권한을 부여 받을 사용자, 권한 대상이 되는 리소스, 해당 리소스에 허용되는 특정 작업을 결정합니다.

주제

EventBridge 리소스 및 작업

EventBridge에서는 규칙이 기본 리소스입니다. EventBridge에서는 이벤트 같은 기본 리소스와 함께 사용할 수 있는 다른 리소스를 지원합니다. 이러한 리소스를 가리켜 하위 리소스(subresource)라 합니다. 이러한 리소스와 하위 리소스에는 고유한 Amazon 리소스 이름(ARN)이 연결됩니다. ARN에 대한 자세한 내용은 Amazon Web Services 일반 참조에서 Amazon 리소스 이름(ARN) 및 AWS 서비스 네임스페이스를 참조하십시오.

리소스 유형	ARN 형식
규칙	`arn:aws:events:region:account:rule/[event-bus-name]/rule-name`
이벤트 버스	`arn:aws:events:region:account:event-bus/event-bus-name`

리소스 유형	ARN 형식
모든 EventBridge 리소스	`arn:aws:events:*`
지정한 리전에서 지정한 계정이 소유한 모든 EventBridge 리소스	`arn:aws:events:region:account:*`

Note

대부분의 AWS 서비스는 콜론(:) 또는 슬래시(/)를 ARN에서 동일한 문자로 처리합니다. 그러나 EventBridge는 이벤트 패턴 및 규칙에서 정확히 일치를 사용합니다. 따라서 이벤트 패턴을 만들 때 일치시키려는 이벤트에서 ARN 구문이 일치하도록 정확한 ARN 문자를 사용해야 합니다.

예를 들어 명령문에서 다음과 같이 ARN을 사용하여 특정 규칙(*myRule*)를 나타낼 수 있습니다.

```
"Resource": "arn:aws:events:us-east-1:123456789012:rule/myRule"
```

다음과 같이 별표 (*) 와일드카드 문자를 사용하여 특정 계정에 속하는 모든 규칙들을 지정할 수도 있습니다.

```
"Resource": "arn:aws:events:us-east-1:123456789012:rule/*"
```

모든 리소스를 지정해야 하거나 특정 API 작업이 ARN을 지원하지 않는 경우 다음과 같이 `Resource` 요소에 별표(*) 와일드카드 문자를 사용합니다.

```
"Resource": "*"
```

일부 EventBridge API 작업에서는 여러 리소스 사용을 허용합니다(예: `PutTargets`). 단일 명령문에서 여러 리소스를 지정하려면 다음과 같이 각 ARN을 쉼표로 구분합니다.

```
"Resource": ["arn1", "arn2"]
```

EventBridge은(는) EventBridge 리소스를 처리하기 위한 작업을 제공합니다. 사용 가능한 작업 목록은 EventBridge 권한 참조 문서 (p. 136) 단원을 참조하십시오.

리소스 소유권 이해

계정은 리소스를 누가 생성했든 상관없이 계정에서 생성된 리소스를 소유합니다. 특히 리소스 소유자는 리소스 생성 요청을 인증하는 보안 주체 개체, 즉 계정 루트 사용자, IAM 사용자 또는 IAM 역할의 계정입니다. 다음 예에서는 이 계정의 작동 방식을 설명합니다.

- 계정의 루트 사용자 자격 증명을 사용하여 규칙을 생성하면, 계정이 EventBridge 리소스의 소유자가 됩니다.
- 계정에서 IAM 사용자를 생성하고 EventBridge 리소스를 생성할 수 있는 권한을 해당 사용자에게 부여하면 해당 사용자는 EventBridge 리소스를 생성할 수 있습니다. 하지만 EventBridge 리소스는 사용자가 속한 계정이 소유합니다.
- 계정에서 EventBridge 리소스를 생성할 권한이 있는 IAM 역할을 만드는 경우, 해당 역할을 담당할 수 있는 사람은 누구나 EventBridge 리소스를 생성할 수 있습니다. 이 경우 역할이 속한 계정이 EventBridge 리소스를 소유합니다.

리소스 액세스 관리

권한 정책은 누가 무엇에 액세스 할 수 있는지를 나타냅니다. 다음 단원에서는 권한 정책을 만드는 데 사용 가능한 옵션에 대해 설명합니다.

Note

이 단원에서는 EventBridge의 맥락에서 IAM을 사용하는 방법에 대해 설명하며, IAM 서비스에 대한 자세한 정보는 다루지 않습니다. 전체 IAM 설명서는 IAM의 IAM 사용 설명서이란 무엇입니까?를 참조하십시오. IAM 정책 구문과 설명에 대한 자세한 내용은 IAM 사용 설명서의 IAM 정책 참조를 참조하십시오.

IAM 자격 증명에 연결된 정책을 자격 증명 기반 정책(IAM 정책)이라고 하고, 리소스에 연결된 정책을 리소스 기반 정책이라고 합니다. EventBridge의 경우에는 자격 증명 기반 정책(IAM 정책)과 리소스 기반 정책을 모두 지원합니다.

주제

자격 증명 기반 정책(IAM 정책)

정책을 IAM 자격 증명에 연결할 수 있습니다. 예를 들면,

- 계정 내 사용자 또는 그룹에 권한 정책 연결 – Amazon CloudWatch 콘솔에서 규칙을 볼 수 있는 사용자 권한을 부여하려면 권한 정책을 사용자 또는 해당 사용자가 속한 그룹에 연결하면 됩니다.
- 역할에 권한 정책 연결(교차 계정 권한 부여) – 자격 증명 기반 권한 정책을 IAM 역할에 연결하여 교차 계정 권한을 부여할 수 있습니다. 예를 들어 계정 A의 관리자는 다음과 같이 다른 계정(예: 계정 B) 또는 AWS 서비스에 교차 계정 권한을 부여할 역할을 생성할 수 있습니다.
  1. 계정 A 관리자는 IAM 역할을 생성하고 계정 A의 리소스에 대한 권한을 부여하는 역할에 권한 정책을 연결합니다.
  2. 계정 A 관리자는 계정 B를 역할을 수임할 보안 주체로 식별하는 역할에 신뢰 정책을 연결합니다.
  3. 그런 다음 계정 B 관리자가 계정 B의 사용자에게 역할을 수임할 권한을 위임할 수 있습니다. 그러면 계정 B의 사용자가 계정 에서 리소스를 생성하거나 액세스할 수 있습니다. 신뢰 정책의 보안 주체가 AWS 제품 보안 주체가 되어 역할을 수임하는 데 필요한 권한을 AWS 제품에 부여할 수도 있습니다.

  IAM을 사용하여 권한을 위임하는 방법에 대한 자세한 정보는 IAM 사용 설명서의 액세스 관리를 참조하십시오.

특정 IAM 정책을 생성하여 계정의 사용자가 액세스할 수 있는 호출 및 리소스를 제한한 다음, 해당 정책을 IAM 사용자에게 연결합니다. IAM 역할 생성 방법 및 EventBridge용 IAM 정책 설명 예제를 살펴보는 방법에 대한 자세한 내용은 EventBridge 리소스에 대한 액세스 권한 관리 개요 (p. 122) 단원을 참조하십시오.

리소스 기반 정책(IAM 정책)

EventBridge에서 규칙이 트리거되면 이 규칙과 연관된 모든 대상들이 호출됩니다. 호출은 AWS Lambda 함수를 호출하여 Amazon SNS 주제에 게시하고 Amazon Kinesis 스트림에 이벤트를 릴레이합니다. 소유하고 있는 리소스에 대해 API 호출을 수행할 수 있으려면 EventBridge가 해당되는 권한을 가지고 있어야 합니다. Lambda, Amazon SNS 및 Amazon SQS 리소스에서 EventBridge는 리소스 기반 정책을 따릅니다. Kinesis 스트림에서 EventBridge는 IAM 역할을 따릅니다.

IAM 역할을 생성하고 EventBridge에 대한 리소스 기반 정책 명령문의 예제를 탐색하는 방법에 대한 자세한 내용은 EventBridge에서 리소스 기반 정책 사용 (p. 132) 단원을 참조하십시오.

정책 요소 지정: 작업, 효과, 보안 주체

각 EventBridge 리소스에 대해 서비스는 API 작업 세트를 정의합니다. 이러한 API 작업에 대한 권한을 부여하기 위해 EventBridge에서는 정책에서 지정할 수 있는 작업을 정의합니다. 일부 API 작업에서는 API 작업을 수행하기 위해 복수의 작업에 대한 권한이 필요할 수 있습니다. 리소스 및 API 작업에 대한 자세한 정보는 EventBridge 리소스 및 작업 (p. 122) 및 EventBridge 권한 참조 문서 (p. 136) 단원을 참조하십시오.

다음은 기본 정책 요소입니다.

- 리소스 – Amazon 리소스 이름(ARN)을 사용하여 정책을 적용할 리소스를 식별합니다. 자세한 내용은 EventBridge 리소스 및 작업 (p. 122) 단원을 참조하십시오.
- 작업 – 작업 키워드를 사용하여 허용 또는 거부할 리소스 작업을 식별합니다. 예를 들어, `events:Describe` 권한은 사용자에게 `Describe` 작업 수행 권한을 허용합니다.
- 효과 – 사용자가 특정 작업을 요청하는 경우 허용할지 아니면 거부할지 그 결과를 지정합니다. 명시적으로 리소스에 대한 액세스 권한을 부여(허용)하지 않는 경우, 액세스는 묵시적으로 거부됩니다. 한 리소스에 대한 액세스를 명시적으로 거부할 수도 있으며, 다른 정책에 따라 액세스 권한이 부여되더라도 사용자가 이 리소스에 액세스하지 못하도록 조치를 취할 수 있습니다.
- 보안 주체 – 자격 증명 기반 정책(IAM 정책)에서 정책이 연결되는 사용자는 암시적인 보안 주체입니다. 리소스 기반 정책의 경우 사용자, 계정, 서비스 또는 권한의 수신자인 기타 엔터티를 지정합니다(리소스 기반 정책에만 해당).

IAM 정책 구문과 설명에 대한 자세한 정보는 IAM 사용 설명서의 IAM JSON 정책 참조를 참조하십시오.

모든 EventBridge API 작업과 해당 작업이 적용되는 리소스를 보여주는 표는 EventBridge 권한 참조 문서 (p. 136) 단원을 참조하십시오.

정책에서 조건 지정

권한을 부여할 때 액세스 정책 언어를 사용하여 조건이 적용되는 조건을 지정할 수 있습니다. 예를 들어, 특정 날짜 이후에만 정책을 적용할 수 있습니다. 정책 언어에서의 조건 지정에 관한 자세한 내용은 IAM 사용 설명서의 조건 단원을 참조하십시오.

조건을 표시하려면 미리 정의된 조건 키를 사용합니다. AWS 차원 조건 키와 EventBridge–지정 키를 적절하게 사용할 수 있습니다. AWS 차원 키의 전체 목록은 IAM 사용 설명서의 사용 가능한 조건 키를 참조하십시오. EventBridge– 지정 키에 대한 전체 목록은 IAM 정책 조건을 사용하여 세부적인 액세스 제어 구현 (p. 138) 단원을 참조하십시오.

EventBridge에 대한 자격 증명 기반 정책(IAM 정책) 사용

이 단원에서는 계정 관리자가 IAM 자격 증명(즉 사용자, 그룹, 역할)에 권한 정책을 연결할 수 있는 자격 증명 기반 정책의 예를 제시합니다.

다음 예제는 사용자가 Kinesis에 이벤트 데이터를 입력할 수 있는 권한 정책을 보여줍니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CloudWatchEventsInvocationAccess",
            "Effect": "Allow",
            "Action": [
                "kinesis:PutRecord"
            ],
            "Resource": "*"
        }
    ]
}
```

이 주제의 단원에서는 다음 내용을 학습합니다.

주제

CloudWatch 콘솔 사용에 필요한 권한

사용자가 CloudWatch 콘솔에서 EventBridge를 작업하려면 자신의 계정에 대해 다른 AWS 리소스를 설명할 수 있는 최소한의 권한이 있어야 합니다. CloudWatch 콘솔에서 EventBridge를 완전히 사용하려면 다음 서비스에도 권한이 있어야 합니다.

- 자동화
- Amazon EC2 Auto Scaling
- AWS CloudTrail
- CloudWatch
- EventBridge
- IAM
- Kinesis
- Lambda
- Amazon SNS
- Amazon SWF

최소 필수 권한보다 더 제한적인 IAM 정책을 만들면 콘솔에서는 해당 IAM 정책에 연결된 사용자에 대해 의도대로 작동하지 않습니다. 이 사용자가 CloudWatch 콘솔을 사용할 수 있도록 하려면 `CloudWatchEventsReadOnlyAccess` 관리형 정책을 사용자에게 연결합니다(EventBridge에 대한 AWS 관리형(미리 정의된) 정책 (p. 127) 참조).

AWS CLI 또는 CloudWatch API만 호출하는 사용자에게 최소 콘솔 권한을 허용할 필요가 없습니다.

CloudWatch 콘솔에서 작업을 수행하는 데 필요한 전체 권한 세트는 아래와 같습니다.

- `automation:CreateAction`
- `automation:DescribeAction`
- `automation:UpdateAction`
- `autoscaling:DescribeAutoScalingGroups`
- `cloudtrail:DescribeTrails`
- `ec2:DescribeInstances`
- `ec2:DescribeVolumes`
- `events:DeleteRule`
- `events:DescribeRule`
- `events:DisableRule`
- `events:EnableRule`
- `events:ListRuleNamesByTarget`
- `events:ListRules`
- `events:ListTargetsByRule`
- `events:PutEvents`
- `events:PutRule`
- `events:PutTargets`

- `events:RemoveTargets`
- `events:TestEventPattern`
- `iam:ListRoles`
- `kinesis:ListStreams`
- `lambda:AddPermission`
- `lambda:ListFunctions`
- `lambda:RemovePermission`
- `sns:GetTopicAttributes`
- `sns:ListTopics`
- `sns:SetTopicAttributes`
- `swf:DescribeAction`
- `swf:ReferenceAction`
- `swf:RegisterAction`
- `swf:RegisterDomain`
- `swf:UpdateAction`

EventBridge에 대한 AWS 관리형(미리 정의된) 정책

AWS는 AWS에서 생성하고 관리하는 독립형 IAM 정책을 제공하여 많은 일반 사용 사례를 처리합니다. 관리형 정책은 사용자가 필요한 권한을 조사할 필요가 없도록 일반 사용 사례에 필요한 권한을 부여합니다. 자세한 내용은 IAM 사용 설명서의 AWS 관리형 정책 단원을 참조하십시오.

계정의 사용자에게 연결할 수 있는 다음 AWS 관리형 정책은 EventBridge에 고유합니다.

- CloudWatchEventsFullAccess – EventBridge에 대한 전체 액세스 권한을 부여합니다.
- CloudWatchEventsInvocationAccess – EventBridge가 계정에서 Amazon Kinesis Data Streams의 스트림에 이벤트를 릴레이하도록 허용합니다.
- CloudWatchEventsReadOnlyAccess – EventBridge에 대한 읽기 전용 액세스 권한을 부여합니다.
- CloudWatchEventsBuiltInTargetExecutionAccess – EventBridge의 기본 제공 대상이 사용자를 대신하여 Amazon EC2 작업을 수행하도록 허용합니다.

이벤트 전송을 위한 IAM 역할

EventBridge가 Kinesis 스트림 대상에 이벤트를 릴레이하려면 IAM 역할을 생성해야 합니다.

EventBridge를 전송하기 위한 IAM 역할을 생성하려면

1. https://console.aws.amazon.com/iam/에서 IAM 콘솔을 엽니다.
2. IAM 사용 설명서의 AWS 서비스에 대한 권한을 위임할 역할 생성 단계에 따라 IAM 역할을 생성합니다. 해당 단계에 따라 역할을 생성할 때 다음을 수행하십시오.
   - 역할 이름에서 계정 내의 고유한 이름(예: CloudWatchEventsSending)을 사용합니다.
   - Select Role Type(역할 유형 선택)에서 AWS Service Roles(AWS 서비스 역할)을 선택한 후 Amazon EventBridge를 선택합니다. 이렇게 하면 역할을 수임할 EventBridge 권한이 부여됩니다.
   - 정책 연결에서 CloudWatchEventsInvocationAccess를 선택합니다.

EventBridge 작업 및 리소스에 대한 권한을 허용하는 고유의 사용자 지정 IAM 정책을 생성할 수도 있습니다. 해당 권한이 필요한 IAM 사용자 또는 그룹에 이러한 사용자 지정 정책을 연결할 수 있습니다. IAM 정책에 대한 자세한 내용은 IAM 사용 설명서의 IAM 정책 개요를 참조하십시오. 사용자 지정 IAM 정책 관리 및 생성에 대한 자세한 내용은 IAM 사용 설명서의 IAM 정책 관리 단원을 참조하십시오.

EventBridge가 특정 대상을 액세스하는 데 필요한 권한

EventBridge가 특정 대상에 액세스하려면 해당 대상에 액세스하기 위한 IAM 역할을 지정해야 하고, 해당 역할에는 특정한 정책이 연결되어야 합니다.

대상이 Kinesis 스트림이면 그 대상에게 이벤트 데이터를 전송하는 데 사용된 역할에 다음 정책이 포함되어야 합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kinesis:PutRecord"
            ],
            "Resource": "*"
        }
    ]
}
```

대상이 시스템 관리자 Run Command이고 해당 명령에 하나 이상의 InstanceIds 값을 지정하는 경우에는 지정한 역할에 다음 정책이 반드시 포함되어야 합니다.

```
{
     "Version": "2012-10-17",
     "Statement": [
           {
                "Action": "ssm:SendCommand",
                "Effect": "Allow",
                "Resource": [
                     "arn:aws:ec2:{{region}}:{{accountId}}:instance/[[instanceIds]]",
                     "arn:aws:ssm:{{region}}:*:document/{{documentName}}"
                ]
           }
     ]
}
```

대상이 시스템 관리자 Run Command이고 해당 명령에 하나 이상의 태그를 지정하는 경우에는 지정한 역할에 다음 정책이 반드시 포함되어야 합니다.

```
{
     "Version": "2012-10-17",
     "Statement": [
           {
                "Action": "ssm:SendCommand",
                "Effect": "Allow",
                "Resource": [
                     "arn:aws:ec2:{{region}}:{{accountId}}:instance/*"
                ],
                "Condition": {
                    "StringEquals": {
                        "ec2:ResourceTag/*": [
                            "[[tagValues]]"
                        ]
                    }
                }
           },
           {
                "Action": "ssm:SendCommand",
                "Effect": "Allow",
```

```
                "Resource": [
                     "arn:aws:ssm:{{region}}:*:document/{{documentName}}"
                ]
            }
     ]
}
```

대상이 AWS Step Functions 상태 시스템인 경우에는 지정한 역할에 다음 정책이 포함되어야 합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
             "Action": [ "states:StartExecution" ],
            "Resource": [ "arn:aws:states:*:*:stateMachine:*" ]
        }
     ]
}
```

대상이 ECS 작업인 경우에는 지정한 역할에 다음 정책이 포함되어야 합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "ecs:RunTask"
        ],
        "Resource": [
            "arn:aws:ecs:*:{{account-id}}:task-definition/{{task-definition-name}}"
        ],
        "Condition": {
            "ArnLike": {
                "ecs:cluster": "arn:aws:ecs:*:{{account-id}}:cluster/{{cluster-name}}"
            }
        }
    }]
}
```

고객 관리형 정책 예

이 단원에서는 다양한 EventBridge 작업에 대한 권한을 부여하는 사용자 정책의 예를 찾을 수 있습니다. 이러한 정책은 EventBridge API, AWS SDK 또는 AWS CLI를 사용하는 경우에 적용됩니다.

> **Note**
>
> 모든 예는 미국 서부(오레곤) 지역(us-west-2)을 사용하며 가상의 계정 ID를 포함합니다.

다음 샘플 IAM 정책을 사용하면 IAM 사용자 및 역할에 대한 EventBridge 액세스를 제한할 수 있습니다.

예제

- 예제 1: CloudWatchEventsBuiltInTargetExecutionAccess (p. 130)
- 예제 2: CloudWatchEventsInvocationAccess (p. 130)
- 예제 3: CloudWatchEventsConsoleAccess (p. 130)
- 예제 4: CloudWatchEventsFullAccess (p. 131)
- 예제 5: CloudWatchEventsReadOnlyAccess (p. 131)
- 예제 6: 태그 지정을 사용하여 특정 규칙에 대한 액세스 제어 (p. 132)

예제 1: CloudWatchEventsBuiltInTargetExecutionAccess

다음 정책은 EventBridge의 기본 제공 대상이 사용자를 대신하여 Amazon EC2 작업을 수행하도록 허용합니다.

> **Important**
>
> 기본 제공 대상을 통한 규칙 생성은 AWS Management 콘솔에서만 지원됩니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CloudWatchEventsBuiltInTargetExecutionAccess",
            "Effect": "Allow",
            "Action": [
                "ec2:Describe*",
                "ec2:RebootInstances",
                "ec2:StopInstances",
                "ec2:TerminateInstances",
                "ec2:CreateSnapshot"
            ],
            "Resource": "*"
        }
    ]
}
```

예제 2: CloudWatchEventsInvocationAccess

다음 정책은 EventBridge가 계정에서 Kinesis 스트림에 이벤트를 릴레이하도록 허용합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CloudWatchEventsInvocationAccess",
            "Effect": "Allow",
            "Action": [
                "kinesis:PutRecord"
            ],
            "Resource": "*"
        }
    ]
}
```

예제 3: CloudWatchEventsConsoleAccess

다음 정책은 IAM 사용자가 EventBridge 콘솔을 사용할 수 있도록 해줍니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CloudWatchEventsConsoleAccess",
            "Effect": "Allow",
            "Action": [
                "automation:CreateAction",
                "automation:DescribeAction",
                "automation:UpdateAction",
                "autoscaling:DescribeAutoScalingGroups",
                "cloudtrail:DescribeTrails",
```

```
                "ec2:DescribeInstances",
                "ec2:DescribeVolumes",
                "events:*",
                "iam:ListRoles",
                "kinesis:ListStreams",
                "lambda:AddPermission",
                "lambda:ListFunctions",
                "lambda:RemovePermission",
                "sns:GetTopicAttributes",
                "sns:ListTopics",
                "sns:SetTopicAttributes",
                "swf:DescribeAction",
                "swf:ReferenceAction",
                "swf:RegisterAction",
                "swf:RegisterDomain",
                "swf:UpdateAction"
            ],
            "Resource": "*"
        },
        {
            "Sid": "IAMPassRoleForCloudWatchEvents",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::*:role/AWS_Events_Invoke_Targets",
                "arn:aws:iam::*:role/AWS_Events_Actions_Execution"
            ]
        }
    ]
}
```

예제 4: CloudWatchEventsFullAccess

다음 정책은 AWS CLI 및 SDK를 통해 EventBridge에 대한 작업을 수행하도록 허용합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CloudWatchEventsFullAccess",
            "Effect": "Allow",
            "Action": "events:*",
            "Resource": "*"
        },
        {
            "Sid": "IAMPassRoleForCloudWatchEvents",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/AWS_Events_Invoke_Targets"
        }
    ]
}
```

예제 5: CloudWatchEventsReadOnlyAccess

다음 정책은 EventBridge에 대한 읽기 전용 액세스 권한을 제공합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CloudWatchEventsReadOnlyAccess",
```

```
            "Effect": "Allow",
            "Action": [
                "events:Describe*",
                "events:List*",
                "events:TestEventPattern"
            ],
            "Resource": "*"
        }
    ]
}
```

예제 6: 태그 지정을 사용하여 특정 규칙에 대한 액세스 제어

사용자에게 지정된 EventBridge 규칙에 대한 액세스 권한을 부여하고 동시에 다른 규칙에는 액세스하지 못하게 할 수 있습니다. 이렇게 하려면 규칙에 태그를 지정하고 해당 태그를 참조하는 IAM 정책을 사용하십시오.

EventBridge 리소스 태그 지정에 대한 자세한 내용은 Amazon EventBridge 리소스에 태그 지정 (p. 152) 단원을 참조하십시오.

EventBridge 규칙에 태그를 지정할 때 특정 태그가 있는 규칙에만 액세스할 수 있도록 사용자에게 IAM 정책에 대한 권한을 부여할 수 있습니다. 예를 들어, 다음 정책 명령문은 태그 키 `Stack`에 대한 값이 `Prod`인 규칙에만 액세스하도록 권한을 부여합니다.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "events:*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "events:ResourceTag/Stack": "Prod"
                }
            }
        }
    ]
}
```

IAM 정책 명령문 사용에 대한 자세한 내용은 IAM 사용 설명서의 정책을 사용하여 액세스 제어를 참조하십시오.

EventBridge에서 리소스 기반 정책 사용

EventBridge에서 규칙이 트리거되면 이 규칙과 연관된 모든 대상들이 호출됩니다. 호출은 AWS Lambda 함수를 호출하여 Amazon SNS 주제에 게시하고 Kinesis 스트림에 이벤트를 릴레이합니다. 소유하고 있는 리소스에 대해 API 호출을 수행할 수 있으려면 EventBridge가 해당되는 권한을 가지고 있어야 합니다. Lambda, Amazon SNS, Amazon SQS 및 Amazon CloudWatch Logs 리소스에서 EventBridge는 리소스 기반 정책을 따릅니다. Kinesis 스트림에서 EventBridge는 IAM 역할을 따릅니다.

다음과 같은 권한을 사용하여 EventBridge 규칙과 연관된 대상을 호출할 수 있습니다. 다음 절차는 AWS CLI를 사용하여 대상에 권한을 추가합니다. AWS CLI의 설치 및 구성 방법에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의 AWS Command Line Interface 초기 설정을 참조하십시오.

주제

- CloudWatch Logs 권한 (p. 136)

AWS Lambda 권한

EventBridge 규칙을 사용하여 AWS Lambda 함수를 호출하려면 Lambda 함수의 정책에 다음 권한을 추가합니다.

```
{
  "Effect": "Allow",
  "Action": "lambda:InvokeFunction",
  "Resource": "arn:aws:lambda:region:account-id:function:function-name",
  "Principal": {
    "Service": "events.amazonaws.com"
  },
  "Condition": {
    "ArnLike": {
      "AWS:SourceArn": "arn:aws:events:region:account-id:rule/rule-name"
    }
  },
  "Sid": "TrustCWEToInvokeMyLambdaFunction"
}
```

EventBridge가 Lambda 함수를 호출하도록 허용하는 권한을 추가하려면

- 명령 프롬프트에서 다음 명령을 입력합니다.

```
aws lambda add-permission --statement-id "TrustCWEToInvokeMyLambdaFunction" \
--action "lambda:InvokeFunction" \
--principal "events.amazonaws.com" \
--function-name "arn:aws:lambda:region:account-id:function:function-name" \
--source-arn "arn:aws:events:region:account-id:rule/rule-name"
```

EventBridge가 Lambda 함수를 호출하도록 허용하는 권한을 설정하는 방법에 대한 자세한 내용은 AWS Lambda Developer Guide의 AddPermission 및 예약된 이벤트에서 Lambda 사용을 참조하십시오.

Amazon SNS 권한

EventBridge가 Amazon SNS 주제를 게시하도록 허용하려면 `aws sns get-topic-attributes` 및 `aws sns set-topic-attributes` 명령을 사용합니다.

> **Note**
>
> EventBridge는 Amazon SNS 주제 정책에서 `Condition`을 사용하는 것을 지원하지 않습니다.

EventBridge가 SNS 주제를 게시하도록 허용하는 권한을 추가하려면

1. 먼저, SNS 주제 속성을 나열합니다. 명령 프롬프트에서 다음을 입력합니다.

```
aws sns get-topic-attributes --topic-arn "arn:aws:sns:region:account-id:topic-name"
```

이 명령은 SNS 주제의 모든 속성을 반환합니다. 다음 예제는 새로 생성된 SNS 주제의 결과를 보여줍니다.

```
{
    "Attributes": {
        "SubscriptionsConfirmed": "0",
```

```
        "DisplayName": "",
        "SubscriptionsDeleted": "0",
        "EffectiveDeliveryPolicy": "{\"http\":{\"defaultHealthyRetryPolicy\":
{\"minDelayTarget\":20,\"maxDelayTarget\":20,\"numRetries\":3,\"numMaxDelayRetries
\":0,\"numNoDelayRetries\":0,\"numMinDelayRetries\":0,\"backoffFunction\":\"linear\"},
\"disableSubscriptionOverrides\":false}}",
        "Owner": "account-id",
        "Policy": "{\"Version\":\"2012-10-17\",\"Id\":\"__default_policy_ID\",
\"Statement\":[{\"Sid\":\"__default_statement_ID\",\"Effect\":\"Allow\",\"Principal
\":{\"AWS\":\"*\"},\"Action\":[\"SNS:GetTopicAttributes\",\"SNS:SetTopicAttributes
\",\"SNS:AddPermission\",\"SNS:RemovePermission\",\"SNS:DeleteTopic\",\"SNS:Subscribe
\",\"SNS:ListSubscriptionsByTopic\",\"SNS:Publish\",\"SNS:Receive\"],\"Resource
\":\"arn:aws:sns:region:account-id:topic-name\",\"Condition\":{\"StringEquals\":
{\"AWS:SourceOwner\":\"account-id\"}}}]}",
        "TopicArn": "arn:aws:sns:region:account-id:topic-name",
        "SubscriptionsPending": "0"
    }
}
```

2. 다음 단계로 다음 명령문을 문자열로 변환하고 `"Policy"` 속성 내의 `"Statement"` 모음에 이를 추가합니다.

```
{
  "Sid": "TrustCWEToPublishEventsToMyTopic",
  "Effect": "Allow",
  "Principal": {
    "Service": "events.amazonaws.com"
  },
  "Action": "sns:Publish",
  "Resource": "arn:aws:sns:region:account-id:topic-name"
}
```

   이 명령문이 문자열로 변환되면 다음과 유사하게 화면에 나타납니다.

```
{\"Sid\":\"TrustCWEToPublishEventsToMyTopic\",\"Effect\":\"Allow\",\"Principal\":
{\"Service\":\"events.amazonaws.com\"},\"Action\":\"sns:Publish\",\"Resource\":
\"arn:aws:sns:region:account-id:topic-name\"}
```

3. 명령문 모음에 명령문 문자열을 추가하고 난 후에 `aws sns set-topic-attributes` 명령을 사용하여 새 정책을 설정합니다.

```
aws sns set-topic-attributes --topic-arn "arn:aws:sns:region:account-id:topic-name" \
--attribute-name Policy \
--attribute-value "{\"Version\":\"2012-10-17\",\"Id\":\"__default_policy_ID\",
\"Statement\":[{\"Sid\":\"__default_statement_ID\",\"Effect\":\"Allow\",\"Principal
\":{\"AWS\":\"*\"},\"Action\":[\"SNS:GetTopicAttributes\",\"SNS:SetTopicAttributes
\",\"SNS:AddPermission\",\"SNS:RemovePermission\",\"SNS:DeleteTopic\",\"SNS:Subscribe
\",\"SNS:ListSubscriptionsByTopic\",\"SNS:Publish\",\"SNS:Receive\"],\"Resource
\":\"arn:aws:sns:region:account-id:topic-name\",\"Condition\":{\"StringEquals\":
{\"AWS:SourceOwner\":\"account-id\"}}}, {\"Sid\":\"TrustCWEToPublishEventsToMyTopic\",
\"Effect\":\"Allow\",\"Principal\":{\"Service\":\"events.amazonaws.com\"},\"Action\":
\"sns:Publish\",\"Resource\":\"arn:aws:sns:region:account-id:topic-name\"}]}"
```

자세한 내용은 Amazon Simple Notification Service API Reference의 SetTopicAttributes 작업을 참조하십시오.

Amazon SQS 권한

EventBridge 규칙이 Amazon SQS 대기열을 호출하도록 허용하려면 `aws sqs get-queue-attributes` 및 `aws sqs set-queue-attributes` 명령을 사용합니다.

EventBridge 규칙이 SQS 대기열을 호출하도록 허용하는 권한을 추가하려면

1. 먼저, SQS 대기열 속성을 나열합니다. 명령 프롬프트에서 다음을 입력합니다.

```
aws sqs get-queue-attributes \
--queue-url https://sqs.region.amazonaws.com/account-id/queue-name \
--attribute-names Policy
```

   새로운 SQS 대기열에서 정책은 기본적으로 비어 있습니다. 명령문을 추가하는 것 외에도 이 명령문이 포함된 정책을 생성해야 합니다.

2. 아래 명령문은 EventBridge가 SQS 대기열에 메시지를 전송하도록 해줍니다.

```
{
  "Sid": "TrustCWEToSendEventsToMyQueue",
  "Effect": "Allow",
  "Principal": {
     "Service": "events.amazonaws.com"
  },
  "Action": "sqs:SendMessage",
  "Resource": "arn:aws:sqs:region:account-id:queue-name",
  "Condition": {
    "ArnEquals": {
      "aws:SourceArn": "arn:aws:events:region:account-id:rule/rule-name"
    }
  }
}
```

3. 그런 다음 위의 명령문을 문자열로 변환합니다. 정책이 문자열로 변환되면 다음과 유사하게 화면에 나타납니다.

```
{\"Sid\": \"TrustCWEToSendEventsToMyQueue\", \"Effect\": \"Allow\", \"Principal\":
 {\"Service\": \"events.amazonaws.com\"}, \"Action\": \"sqs:SendMessage\", \"Resource
\": \"arn:aws:sqs:region:account-id:queue-name\", \"Condition\": {\"ArnEquals\":
 {\"aws:SourceArn\": \"arn:aws:events:region:account-id:rule/rule-name\"}}
```

4. 다음 콘텐츠를 통해 `set-queue-attributes.json`라는 파일을 생성합니다.

```
{
    "Policy": "{\"Version\":\"2012-10-17\",\"Id\":\"arn:aws:sqs:region:account-
id:queue-name/SQSDefaultPolicy\",\"Statement\":[{\"Sid\":
 \"TrustCWEToSendEventsToMyQueue\", \"Effect\": \"Allow\", \"Principal\": {\"Service
\": \"events.amazonaws.com\"}, \"Action\": \"sqs:SendMessage\", \"Resource\":
 \"arn:aws:sqs:region:account-id:queue-name\", \"Condition\": {\"ArnEquals\":
 {\"aws:SourceArn\": \"arn:aws:events:region:account-id:rule/rule-name\"}}}]}"
}
```

5. `set-queue-attributes.json` 파일을 입력으로 사용하여 정책 속성을 설정합니다. 명령 프롬프트에서 다음을 입력합니다.

```
aws sqs set-queue-attributes \
--queue-url https://sqs.region.amazonaws.com/account-id/queue-name \
--attributes file://set-queue-attributes.json
```

   SQS 대기열에 이미 정책이 있는 경우에는 원래 정책을 복사해서 `set-queue-attributes.json` 파일에 있는 새 명령문과 결합하고 앞에 나온 명령을 실행해서 정책을 업데이트해야 합니다.

자세한 내용은 Amazon Simple Queue Service 개발자 안내서의 Amazon SQS 정책 예제를 참조하십시오.

CloudWatch Logs 권한

CloudWatch Logs가 이 규칙의 대상일 경우, EventBridge는 로그 스트림을 생성하며 CloudWatch Logs는 로그 항목으로 트리거된 이벤트에서 텍스트를 저장합니다. EventBridge에서 로그 스트림을 생성하고 이벤트를 기록하도록 허용하려면 CloudWatch Logs에 EventBridge가 CloudWatch Logs에 기록할 수 있도록 허용하는 리소스 기반 정책이 포함되어야 합니다. AWS Management 콘솔을 사용하여 CloudWatch Logs를 규칙의 대상으로 추가하는 경우, 이 정책은 자동으로 생성됩니다. AWS CLI를 사용하여 대상을 추가하는 경우, 정책이 이미 존재하지 않을 경우, 이 정책을 생성해야 합니다. 다음 예는 필요한 정책을 보여줍니다. 이 예제는 EventBridge가 `/aws/events/`로 시작하는 이름을 가진 모든 로그 그룹에 기록하도록 허용합니다. 이러한 종류의 로그에 대해 다른 로그 그룹 명명 정책을 사용하는 경우, 정책을 이에 따라 조정해야 합니다.

```
{
    "Statement": [
        {
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Effect": "Allow",
            "Principal": {
                "Service": "events.amazonaws.com"
            },
            "Resource": "arn:aws:logs:{{region}}:{{account}}:log-group:/aws/events/*:*",
            "Sid": "TrustEventsToStoreLogEvent"
        }
    ],
    "Version": "2012-10-17"
}
```

자세한 내용은 IAM 사용 설명서의 리소스에 대한 액세스 제어를 참조하십시오.

EventBridge 권한 참조 문서

IAM 자격 증명에 연결할 수 있는 액세스 제어 (p. 122) 및 쓰기 권한 정책(자격 증명 기반 정책)을 설정할 때 다음 표를 참조로 사용할 수 있습니다. 표에는 각 EventBridge API 작업과 이 작업을 수행할 수 있는 권한을 부여할 수 있는 작업이 나와 있습니다. 정책의 `Action` 필드에서 작업을 지정하고, 정책의 `Resource` 필드에서 리소스 값으로 와일드카드 문자(`*`)를 지정합니다.

EventBridge 정책에서 AWS 차원 조건 키를 사용하여 조건을 표시할 수 있습니다. AWS 차원 키의 전체 목록은 IAM 사용 설명서의 사용할 수 있는 키 단원을 참조하십시오.

> **Note**
>
> 작업을 지정하려면 `events:` 접두사 다음에 API 작업 이름을 사용합니다. 예: `events:PutRule`, `events:EnableRule` 또는 `events:*`(모든 EventBridge 작업의 경우).

명령문 하나에 여러 작업을 지정하려면 다음과 같이 쉼표로 구분합니다.

```
"Action": ["events:action1", "events:action2"]
```

와일드카드를 사용하여 여러 작업을 지정할 수도 있습니다. 예를 들어 다음과 같이 이름이 `"Put"`으로 시작되는 모든 작업을 지정할 수 있습니다.

```
"Action": "events:Put*"
```

모든 EventBridge API 작업을 지정하려면 다음과 같이 `*` 와일드카드를 사용합니다.

```
"Action": "events:*"
```

다음 표는 EventBridge에 사용할 IAM 정책에서 지정할 수 있는 작업을 나열합니다.

EventBridge API 작업 및 작업에 필요한 권한

EventBridge API 작업	필요한 권한(API 작업)
DeleteRule	`events:DeleteRule` 규칙을 삭제하는 데 필요합니다.
DescribeEventBus	`events:DescribeEventBus` 현재 계정의 이벤트 버스에 이벤트를 기록할 수 있는 계정을 나열해야 합니다.
DescribeRule	`events:DescribeRule` 규칙에 대한 세부 사항을 나열하는 데 필요합니다.
DisableRule	`events:DisableRule` 규칙을 비활성화하는 데 필요합니다.
EnableRule	`events:EnableRule` 규칙을 활성화하는 데 필요합니다.
ListRuleNamesByTarget	`events:ListRuleNamesByTarget` 대상과 연관된 규칙을 나열하는 데 필요합니다.
ListRules	`events:ListRules` 계정에서 모든 그룹을 나열하는 데 필요합니다.
ListTagsForResource	`events:ListTagsForResource` EventBridge 리소스와 연결된 모든 태그를 나열하는 데 필요합니다. 현재는 규칙에만 태그를 지정할 수 있습니다.
ListTargetsByRule	`events:ListTargetsByRule` 규칙과 연관된 모든 대상을 나열하는 데 필요합니다.
PutEvents	`events:PutEvents` 규칙에 일치시킬 수 있는 사용자 지정 이벤트를 추가하는 데 필요합니다.
PutPermission	`events:PutPermission` 이 계정의 기본 이벤트 버스에 이벤트를 기록할 수 있는 계정 권한을 하나 더 부여해야 합니다.
PutRule	`events:PutRule` 규칙을 생성 또는 업데이트하는 데 필요합니다.

EventBridge API 작업	필요한 권한(API 작업)
PutTargets	`events:PutTargets` 규칙에 대상을 추가하는 데 필요합니다.
RemovePermission	`events:RemovePermission` 이 계정의 기본 이벤트 버스에 이벤트를 기록할 수 있는 다른 계정의 권한을 취소해야 합니다.
RemoveTargets	`events:RemoveTargets` 규칙에서 대상을 제거하는 데 필요합니다.
TestEventPattern	`events:TestEventPattern` 특정 이벤트를 기준으로 이벤트 패턴을 테스트하는 데 필요합니다.

IAM 정책 조건을 사용하여 세부적인 액세스 제어 구현

권한을 부여할 때 IAM 정책 언어를 사용하여 정책이 적용되는 조건을 지정할 수 있습니다. 정책 명령문에서 정책이 적용되는 시점을 제어하는 조건을 지정할 수 있습니다. 각 조건에는 하나 이상의 키-값 쌍이 포함됩니다. 조건 키에는 대/소문자가 구분되지 않습니다. 예를 들어, 특정 날짜 이후에만 정책을 적용할 수 있습니다.

여러 조건을 지정하거나 조건 하나에 여러 키를 지정하는 경우 논리적 AND 연산을 적용하여 조건을 평가합니다. 조건 하나에서 키 하나에 여러 값을 지정하면 논리적 OR 연산자를 적용하여 조건을 평가합니다. 모든 조건이 충족되어야 권한이 부여됩니다.

조건을 지정할 때 자리표시자를 사용할 수도 있습니다. 자세한 내용은 IAM 사용 설명서에서 정책 변수를 참조하십시오. IAM 정책 언어에서의 조건 지정에 관한 자세한 내용은 IAM 사용 설명서의 조건을 참조하십시오.

기본적으로 IAM 사용자와 역할은 계정의 이벤트에 전혀 액세스할 수 없습니다. 이벤트를 사용하려면 `PutRule` API 작업에 대해 허가를 받아야 합니다. 정책을 통해 IAM 사용자나 역할에 `events:PutRule` 작업을 허용하면 특정 이벤트와 일치하는 규칙을 생성할 수 있습니다. 규칙에 대상을 추가해야 합니다. 대상이 없는 규칙은 들어오는 이벤트와 일치할 때 CloudWatch 지표를 게시하는 것 외에 어떤 작업도 수행할 수 없습니다. IAM 사용자나 역할은 `events:PutTargets` 작업에 대한 권한이 있어야 합니다.

특정 리소스와 이벤트 유형에 대한 권한 부여의 범위를 설정하여 이벤트에 대한 액세스 권한을 제한할 수 있습니다(`events:source` 및 `events:detail-type` 조건 키 사용). IAM 사용자나 역할이 특정한 소스 세트 및 세부 유형에만 일치하는 규칙을 생성할 수 있도록 정책 명령문에 조건을 제공할 수 있습니다.

마찬가지로 정책 명령문에서의 조건 설정을 통해 IAM 사용자나 역할이 규칙에 추가할 수 있는 (`events:TargetArn` 조건 키 사용) 계정의 특정 리소스를 확인할 수 있습니다. 예를 들어 계정에서 CloudTrail를 활성화하고 CloudTrail 스트림이 존재하면 EventBridge를 통해 계정의 사용자가 CloudTrail 이벤트를 사용할 수 있게 됩니다. 사용자가 EventBridge를 사용하고 모든 CloudTrail 이벤트를 액세스할 수 있게 하려면 사용자나 역할이 생성한 어떤 규칙도 CloudTrail 이벤트 유형과 일치할 수 없도록 하는 조건문을 통해 `PutRule` API 작업에 거부 명령문을 추가할 수 있습니다.

CloudTrail 이벤트의 경우, 원래 API 호출을 요청한 특정 보안 주체로 액세스를 제한할 수 있습니다 (`events:detail.userIdentity.principalId` 조건 키 사용). 예를 들어 사용자가 감사나 과학 수사에서 사용하는 계정의 특정 IAM 역할에서 생성된 이벤트를 제외하고 모든 CloudTrail 이벤트를 볼 수 있도록 할 수 있습니다.

조건 키	키/값 페어	평가 유형
`events:source`	`"events:source":"source "` 여기에서 *source*는 `"aws.ec2"`, `"aws.s3"` 같은 이벤트의 소스 필드를 위한 리터럴 문자열입니다. *source*에 사용 가능한 값을 더 보려면 EventBridge 지원되는 AWS 서비스의 이벤트 예제 (p. 57) 단원에서 예제 이벤트를 참조하십시오.	소스, null
`events:detail-type`	`"events:detail-type":"detail-type "` 여기에서 *detail-type*은 `"AWS API Call via CloudTrail"` 및 `"EC2 Instance State-change Notification"` 같은 이벤트의 detail-type 필드를 위한 리터럴 문자열입니다. *detail-type*에 사용 가능한 값을 더 보려면 EventBridge 지원되는 AWS 서비스의 이벤트 예제 (p. 57) 단원에서 예제 이벤트를 참조하십시오.	세부 유형, null
`events:detail.userIdentity.principalId`	`"events:detail.userIdentity.principalId":"principal-id"` 여기에서 *principal-id*는 detail-type이 `"AWS API Call via CloudTrail"`(예: `"AROAIDPPEZS35WEXAMPLE:AssumedRoleSessionName."`)인 이벤트의 detail.userIdentity.principalId 필드를 위한 리터럴 문자열입니다.	보안 주체 ID, null
`events:detail.service`	`"events:detail.service":"service"` *service*는 이벤트의 detail.service 필드에 대한 리터럴 문자열(예: `"ABUSE"`)입니다.	service, Null
`events:detail.eventTypeCode`	`"events:detail.eventTypeCode":"eventTypeCode"` *eventTypeCode*는 이벤트의 detail.eventTypeCode 필드에 대한 리터럴 문자열(예: `"AWS_ABUSE_DOS_REPORT"`)입니다.	eventTypeCode, Null
`events:TargetArn`	`"events:TargetArn":"target-arn "` 여기에서 *target-arn*은 `"arn:aws:lambda:*:*:function:*"` 같은 규칙에 입력할 수 있는 대상의 ARN입니다.	ARN, Null

EventBridge용 예제 정책 명령문은 EventBridge 리소스에 대한 액세스 권한 관리 개요 (p. 122) 단원을 참조하십시오.

주제

예제 1: 특정 리소스로 액세스 권한 제한

다음은 IAM 사용자에게 연결할 수 있는 정책의 예제입니다. 정책 A는 모든 이벤트에서 `PutRule` API 작업을 허용하는 반면, 정책 B는 생성 중인 규칙의 이벤트 패턴이 Amazon EC2 이벤트에 일치하는 경우에만 `PutRule`을 허용합니다.

정책 A:-모든 이벤트 허용 —

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutRuleForAllEvents",
            "Effect": "Allow",
            "Action": "events:PutRule",
            "Resource": "*"
        }
    ]
    }
```

정책 B:-Amazon EC2의 이벤트만 허용 —

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutRuleForAllEC2Events",
            "Effect": "Allow",
            "Action": "events:PutRule",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "events:source": "aws.ec2"
                }
            }
        }
    ]
}
```

`EventPattern`은 `PutRule`에 대한 필수 인수입니다. 따라서 정책 B의 사용자가 다음과 같은 이벤트 패턴을 통해 `PutRule`을 호출하는 경우에는

```
{
    "source": [ "aws.ec2" ]
}
```

정책이 이러한 특정 소스(예: `"aws.ec2"`)를 허용하기 때문에 규칙을 생성할 수 있습니다. 그러나 정책 B를 사용하는 사용자가 다음과 같은 이벤트 패턴으로 `PutRule`을 호출하면 정책이 이 특정 소스(즉, `"aws.s3"`)를 허용하지 않으므로 규칙 생성이 거부됩니다.

```
{
    "source": [ "aws.s3" ]
}
```

기본적으로 정책 B의 사용자만 Amazon EC2에서 호출된 이벤트와 일치하는 규칙을 생성할 수 있습니다. 따라서 이들만 Amazon EC2에서 이벤트 액세스가 허용됩니다.

정책 A와 정책 B를 비교하려면 다음 표를 참고하십시오.

이벤트 패턴	정책 A에서 허용	정책 B에서 허용
`{ "source": [ "aws.ec2" ] }`	예	예
`{ "source": [ "aws.ec2", "aws.s3" ] }`	예	아니요(소스 aws.s3이 허용되지 않음)
`{ "source": [ "aws.ec2" ], "detail-type": [ "EC2 Instance State-change Notification" ] }`	예	예
`{ "detail-type": [ "EC2 Instance State-change Notification" ] }`	예	아니요(소스를 지정해야 함)

예제 2: 이벤트 패턴에서 개별적으로 사용할 수 있는 소스를 여러 개 정의

다음 정책은 Amazon EC2 또는 Amazon ECS에서 나온 이벤트를 허용합니다. 즉, IAM 사용자나 역할이 `EventPattern`의 소스가 `"aws.ec2"` 또는 `"aws.ecs"`로 지정되도록 규칙을 생성할 수 있습니다. 소스를 정의하지 않으면 "거부"가 됩니다.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutRuleIfSourceIsEC2OrECS",
            "Effect": "Allow",
            "Action": "events:PutRule",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "events:source": [ "aws.ec2", "aws.ecs" ]
                }
            }
        }
    ]
}
```

다음 표에는 이 정책에서 허용 또는 거부되는 이벤트 패턴의 예제가 나와 있습니다.

이벤트 패턴	정책에서 허용
`{ "source": [ "aws.ec2" ] }`	예
`{ "source": [ "aws.ecs" ] }`	예
`{ "source": [ "aws.s3" ] }`	아니요
`{ "source": [ "aws.ec2", "aws.ecs" ] }`	아니요
`{ "detail-type": [ "AWS API Call via CloudTrail" ] }`	아니요

예제 3: 이벤트 패턴에서 사용할 수 있는 소스와 DetailType을 정의

다음 정책은 DetailType가 `EC2 instance state change notification`인 `aws.ec2` 소스에서 나온 이벤트만 허용합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid":
 "AllowPutRuleIfSourceIsEC2AndDetailTypeIsInstanceStateChangeNotification",
            "Effect": "Allow",
            "Action": "events:PutRule",
```

```
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "events:source": "aws.ec2",
                    "events:detail-type": "EC2 Instance State-change Notification"
                }
            }
        }
    ]
}
```

다음 표에는 이 정책에서 허용 또는 거부되는 이벤트 패턴의 예제가 나와 있습니다.

이벤트 패턴	정책에서 허용
`{ "source": [ "aws.ec2" ] }`	아니요
`{ "source": [ "aws.ecs" ] }`	아니요
`{ "source": [ "aws.ec2" ], "detail-type": [ "EC2 Instance State-change Notification" ] }`	예
`{ "source": [ "aws.ec2" ], "detail-type": [ "EC2 Instance Health Failed" ] }`	아니요
`{ "detail-type": [ "EC2 Instance State-change Notification" ] }`	아니요

예제 4: 이벤트 패턴에서 소스가 정의되었는지 확인

다음 정책은 소스 필드가 반드시 포함되어 있는 `EventPatterns`으로 규칙을 생성하는 것을 허용합니다. 즉, IAM 사용자나 역할은 특정 소스를 제공하지 않는 `EventPattern`으로는 규칙을 생성할 수 없습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutRuleIfSourceIsSpecified",
            "Effect": "Allow",
            "Action": "events:PutRule",
            "Resource": "*",
            "Condition": {
                "Null": {
```

```
                    "events:source": "false"
                }
            }
        }
    ]
}
```

다음 표에는 이 정책에서 허용 또는 거부되는 이벤트 패턴의 예제가 나와 있습니다.

이벤트 패턴	정책에서 허용

<pre>{
    "source": ["aws.ec2"],
    "detail-type": ["EC2 Instance
 State-change Notification"]
}</pre>
	예

<pre>{
    "source": ["aws.ecs",
 "aws.ec2"]
}</pre>
	예

<pre>{
    "detail-type": ["EC2 Instance
 State-change Notification"]
}</pre>
	아니요

예제 5: 여러 개의 소스를 가진 이벤트 패턴에서 허용되는 소스의 목록을 정의

다음 정책은 여러 개의 소스를 가질 수 있는 `EventPatterns`으로 규칙을 생성하는 것을 허용합니다. 이벤트 패턴에 나열된 각 소스는 해당 조건에서 제공되는 목록의 구성원이어야 합니다. `ForAllValues` 조건을 사용할 때는 이 조건의 항목 중 적어도 하나가 반드시 정의되어 있어야 합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutRuleIfSourceIsSpecifiedAndIsEitherS3OrEC2OrBoth",
            "Effect": "Allow",
            "Action": "events:PutRule",
            "Resource": "*",
            "Condition": {
                "ForAllValues:StringEquals": {
                    "events:source": [ "aws.ec2", "aws.s3" ]
                },
                "Null": {
                    "events:source": "false"
                }
            }
        }
    ]
}
```

다음 표에는 이 정책에서 허용 또는 거부되는 이벤트 패턴의 예제가 나와 있습니다.

이벤트 패턴	정책에서 허용

<pre>{
    "source": ["aws.ec2"]
}</pre>
	예

<pre>{
    "source": ["aws.ec2", "aws.s3"]
}</pre>
	예

<pre>{
    "source": ["aws.ec2",
 "aws.autoscaling"]
}</pre>
	아니요

<pre>{
    "detail-type": ["EC2 Instance
 State-change Notification"]
}</pre>
	아니요

예제 6: detail.service로 PutRule 액세스 제한

IAM 사용자 또는 역할에 대해 `events:details.service` 필드에 특정 값이 있는 이벤트에 대해서만 규칙을 생성하도록 제한할 수 있습니다. `events:details.service`의 값이 반드시 AWS 서비스의 이름일 필요는 없습니다.

이 정책 조건은 보안 또는 위반과 관련된 AWS 상태의 이벤트를 사용할 때 유용합니다. 이 정책 조건을 사용하면 이러한 기밀 경보에 대한 액세스를, 해당 경보를 볼 필요가 있는 사용자로만 제한할 수 있습니다.

예를 들어 다음 정책은 `events:details.service` 값이 `ABUSE`인 경우에만 이벤트에 대한 규칙 생성을 허용합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutRuleEventsWithDetailServiceEC2",
            "Effect": "Allow",
            "Action": "events:PutRule",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "events:detail.service": "ABUSE"
                }
            }
        }
    ]
}
```

예제 7: detail.eventTypeCode로 PutRule 액세스 제한

IAM 사용자 또는 역할에 대해 `events:details.eventTypeCode` 필드에 특정 값이 있는 이벤트에 대해서만 규칙을 생성하도록 제한할 수 있습니다. 이 정책 조건은 보안 또는 위반과 관련된 AWS 상태의 이벤트를 사용할 때 유용합니다. 이 정책 조건을 사용하면 이러한 기밀 경보에 대한 액세스를, 해당 경보를 볼 필요가 있는 사용자로만 제한할 수 있습니다.

예를 들어 다음 정책은 `events:details.eventTypeCode` 값이 `AWS_ABUSE_DOS_REPORT`인 경우에만 이벤트에 대한 규칙 생성을 허용합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutRuleEventsWithDetailServiceEC2",
            "Effect": "Allow",
            "Action": "events:PutRule",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "events:detail.eventTypeCode": "AWS_ABUSE_DOS_REPORT"
                }
            }
        }
    ]
}
```

예제 8: 특정 PrincipalId에서의 API 호출에 대한 AWS CloudTrail 이벤트가 사용되는지 확인

모든 AWS CloudTrail 이벤트는 이벤트의 `detail.userIdentity.principalId` 경로에서 API 호출(`PrincipalId`)을 한 사용자의 ID를 가지고 있습니다. `events:detail.userIdentity.principalId` 조건 키를 사용하여 IAM 사용자나 역할이 특정 계정에서 들어오는 호출에 대한 CloudTrail 이벤트만 액세스할 수 있도록 제한할 수 있습니다.

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutRuleOnlyForCloudTrailEventsWhereUserIsASpecificIAMUser",
            "Effect": "Allow",
            "Action": "events:PutRule",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "events:detail-type": [ "AWS API Call via CloudTrail" ],
                    "events:detail.userIdentity.principalId": [ "AIDAJ45Q7YFFAREXAMPLE" ]
                }
            }
        }
    ]
}
```

다음 표에는 이 정책에서 허용 또는 거부되는 이벤트 패턴의 예제가 나와 있습니다.

이벤트 패턴	정책에서 허용
`{ "detail-type": [ "AWS API Call via CloudTrail" ] }`	아니요
`{ "detail-type": [ "AWS API Call via CloudTrail" ],`	예

이벤트 패턴	정책에서 허용

<pre>"detail.userIdentity.principalId":
["AIDAJ45Q7YFFAREXAMPLE"]
}</pre>	

<pre>{
"detail-type": ["AWS API Call
via CloudTrail"],
 	
"detail.userIdentity.principalId":
["AROAIDPPEZS35WEXAMPLE:AssumedRoleSessionName"]
}</pre>
	아니요

예제 9: 대상에 대한 액세스 제한

IAM 사용자나 역할이 `events:PutTargets` 권한을 가지고 있는 경우에는 동일한 계정을 가진 모든 대상을 액세스를 허용하는 규칙에 추가할 수 있습니다. 예를 들어 다음 정책은 특정 규칙(`123456789012` 계정 하의 `MyRule`)에만 대상을 추가하도록 제한합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutTargetsOnASpecificRule",
            "Effect": "Allow",
            "Action": "events:PutTargets",
            "Resource": "arn:aws:events:us-east-1:123456789012:rule/MyRule"
        }
    ]
}
```

규칙에 추가할 수 있는 대상을 제한하려면 `events:TargetArn` 조건 키를 사용합니다. 예를 들어 다음 예제에서와 같이 Lambda 함수로만 대상을 제한할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutTargetsOnASpecificRuleAndOnlyLambdaFunctions",
            "Effect": "Allow",
            "Action": "events:PutTargets",
            "Resource": "arn:aws:events:us-east-1:123456789012:rule/MyRule",
            "Condition": {
                "ArnLike": {
                    "events:TargetArn": "arn:aws:lambda:*:*:function:*"
                }
            }
        }
    ]
}
```

Amazon EventBridge의 로깅 및 모니터링

Amazon EventBridge은 EventBridge에서 사용자, 역할 또는 AWS 서비스가 수행한 작업에 대한 레코드를 제공하는 서비스인 AWS CloudTrail과 통합됩니다. CloudTrail은 AWS 계정에 의해 실행되거나 AWS 계정을 대

신하여 실행되는 API 호출을 기록합니다. 캡처되는 호출에는 CloudWatch 콘솔로부터의 호출과 EventBridge API 작업에 대한 코드 호출이 포함됩니다. 추적을 생성하면 EventBridge에 대한 이벤트를 포함해 CloudTrail 이벤트를 Amazon S3 버킷에 지속적으로 제공할 수 있습니다. 추적을 구성하지 않은 경우 이벤트 기록에서 CloudTrail 콘솔의 최신 이벤트를 볼 수도 있습니다. CloudTrail에서 수집하는 정보를 사용하여 EventBridge에 수행된 요청, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다.

그 구성 및 활성화 방법을 포함하여 CloudTrail에 대한 자세한 내용은 AWS CloudTrail User Guide를 참조하십시오.

주제

CloudTrail의 EventBridge 정보

CloudTrail은 계정 생성 시 AWS 계정에서 활성화됩니다. 지원되는 이벤트 활동이 EventBridge에서 수행되면 해당 활동은 Event history(이벤트 이력)의 다른 AWS 서비스 이벤트와 함께 CloudTrail 이벤트에 기록됩니다. AWS 계정에서 최신 이벤트를 확인, 검색 및 다운로드할 수 있습니다. 자세한 내용은 CloudTrail 이벤트 기록에서 이벤트 보기를 참조하십시오.

EventBridge 이벤트를 포함하여 AWS 계정에 이벤트를 지속적으로 기록하려는 경우 추적을 생성합니다. 추적은 CloudTrail이 Amazon S3 버킷으로 로그 파일을 전송할 수 있도록 합니다. 콘솔에서 추적을 생성하면 기본적으로 모든 AWS 리전에 추적이 적용됩니다. 추적은 AWS 파티션에 있는 모든 리전의 이벤트를 로깅하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 또는 CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 정보는 다음을 참조하십시오.

- 추적 생성 개요
- CloudTrail 지원 서비스 및 통합
- CloudTrail에 대한 Amazon SNS 알림 구성
- 여러 리전에서 CloudTrail 로그 파일 받기 및 여러 계정에서 CloudTrail 로그 파일 받기

EventBridge에서는 다음 작업을 CloudTrail 로그 파일에 이벤트로 로깅할 수 있습니다.

- DeleteRule
- DescribeEventBus
- DescribeRule
- DisableRule
- EnableRule
- ListRuleNamesByTarget
- ListRules
- ListTargetsByRule
- PutPermission
- PutRule
- PutTargets
- RemoveTargets
- TestEventPattern

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 자격 증명 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트로 했는지 아니면 AWS Identity and Access Management(IAM) 사용자 자격 증명으로 했는지 여부
- 역할 또는 연합된 사용자에 대한 임시 보안 자격 증명을 사용하여 요청이 생성되었는지 여부.
- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 CloudTrail userIdentity 요소를 참조하십시오.

예제: EventBridge 로그 파일 항목

추적은 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 제공할 수 있도록 해 주는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함됩니다. 이벤트는 어떤 소스로부터의 단일 요청을 나타내며 요청된 작업, 작업 날짜와 시간, 요청 파라미터 등에 대한 정보가 포함되어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 호출의 주문 스택 추적이 아니므로 특정 순서로 표시되지 않습니다.

아래 CloudTrail 로그 파일 항목은 사용자가 EventBridge의 PutRule 작업을 호출했음을 보여줍니다.


```
{
        "eventVersion":"1.03",
        "userIdentity":{
           "type":"Root",
           "principalId":"123456789012",
           "arn":"arn:aws:iam::123456789012:root",
           "accountId":"123456789012",
           "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
           "sessionContext":{
              "attributes":{
                 "mfaAuthenticated":"false",
                 "creationDate":"2015-11-17T23:56:15Z"
              }
           }
        },
        "eventTime":"2015-11-18T00:11:28Z",
        "eventSource":"events.amazonaws.com",
        "eventName":"PutRule",
        "awsRegion":"us-east-1",
        "sourceIPAddress":"AWS Internal",
        "userAgent":"AWS CloudWatch Console",
        "requestParameters":{
           "description":"",
           "name":"cttest2",
           "state":"ENABLED",
           "eventPattern":"{\"source\":[\"aws.ec2\"],\"detail-type\":[\"EC2 Instance
 State-change Notification\"]}",
           "scheduleExpression":""
        },
        "responseElements":{
           "ruleArn":"arn:aws:events:us-east-1:123456789012:rule/cttest2"
        },
        "requestID":"e9caf887-8d88-11e5-a331-3332aa445952",
        "eventID":"49d14f36-6450-44a5-a501-b0fdcdfaeb98",
        "eventType":"AwsApiCall",
        "apiVersion":"2015-10-07",
        "recipientAccountId":"123456789012"
}
```

Amazon EventBridge의 규정 준수 확인

타사 감사자는 여러 AWS 규정 준수 프로그램의 일환으로 Amazon EventBridge의 보안 및 규정 준수를 평가합니다. 여기에는 SOC, PCI, FedRAMP, HIPAA 등이 포함됩니다.

특정 규정 준수 프로그램의 범위 내에 있는 AWS 서비스 목록은 규정 준수 프로그램 제공 범위 내 AWS 서비스를 참조하십시오. 일반적인 내용은 AWS 규정 준수 프로그램을 참조하십시오.

AWS Artifact를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 AWS 아티팩트에서 보고서 다운로드를 참조하십시오.

EventBridge 사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표 및 관련 법률과 규정에 따라 결정됩니다. AWS에서는 규정 준수를 지원할 다음과 같은 리소스를 제공합니다.

- 보안 및 규정 준수 빠른 시작 안내서 – 이 배포 가이드에서는 아키텍처 고려 사항에 대해 설명하고 보안 및 규정 준수에 중점을 둔 기본 AWS 환경을 배포하기 위한 단계를 제공합니다.
- HIPAA 보안 및 규정 준수 기술 백서 설계 – 이 백서는 기업에서 AWS를 사용하여 HIPAA를 준수하는 애플리케이션을 만드는 방법을 설명합니다.
- AWS 규정 준수 리소스 – 이 워크북 및 안내서 모음은 귀사가 속한 업계 및 위치에 적용될 수 있습니다.
- AWS Config 개발자 안내서의 규칙을 사용하여 리소스 평가 – 이 AWS Config 서비스는 리소스 구성이 내부 사례, 업계 지침, 규정을 얼마나 잘 준수하는지 평가합니다.
- AWS Security Hub – 이 AWS 제품으로 보안 업계 표준 및 모범 사례 규정 준수 여부를 확인하는 데 도움이 되는 AWS 내 보안 상태에 대한 포괄적인 관점을 제공합니다.

Amazon EventBridge의 복원성

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다. AWS 리전에서는 물리적으로 분리되고 격리된 다수의 가용 영역을 제공하며 이러한 가용 영역은 짧은 지연 시간, 높은 처리량 및 높은 중복성을 갖춘 네트워크에 연결되어 있습니다. 가용 영역을 사용하면 중단 없이 영역 간에 자동으로 장애 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS 리전 및 가용 영역에 대한 자세한 내용은 AWS 글로벌 인프라를 참조하십시오.

EventBridge는 AWS 글로벌 인프라 외에도 데이터 복원성과 백업 요구 사항을 지원하는 다양한 기능을 제공합니다.

Amazon EventBridge의 인프라 보안

관리형 서비스인 Amazon EventBridge는 Amazon Web Services: 보안 프로세스 개요 백서에 설명된 AWS 글로벌 네트워크 보안 절차로 보호됩니다.

AWS에서 게시한 API 호출을 사용하여 네트워크를 통해 EventBridge에 액세스합니다. 클라이언트가 TLS(전송 계층 보안) 1.0 이상을 지원해야 합니다. TLS 1.2 이상을 권장합니다. 클라이언트는 Ephemeral Diffie-Hellman(DHE) 또는 Elliptic Curve Ephemeral Diffie-Hellman(ECDHE)과 같은 PFS(전달 완전 보안, Perfect Forward Secrecy)가 포함된 암호 제품군도 지원해야 합니다. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 요청은 액세스 키 ID 및 IAM 주체와 관련된 보안 액세스 키를 사용하여 서명해야 합니다. 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 보안 자격 증명을 생성하여 요청에 서명할 수 있습니다.

이러한 API 작업은 모든 네트워크 위치에서 호출할 수 있지만, EventBridge는 소스 IP 주소를 기반으로 하는 제한을 포함할 수 있는 리소스 기반 액세스 정책을 지원합니다. EventBridge 정책을 사용하여 특정 Amazon

Virtual Private Cloud(Amazon VPC) 엔드포인트 또는 특정 VPC에서 액세스를 제어할 수도 있습니다. 그러면 AWS 네트워크의 특정 VPC에서만 주어진 EventBridge 리소스에 대한 네트워크 액세스가 효과적으로 격리됩니다.

Amazon EventBridge에서 구성 및 취약성 분석

구성 및 IT 제어는 AWS와 고객 간의 공동 책임입니다. 자세한 내용은 AWS 공동 책임 모델을 참조하십시오.

Amazon EventBridge 리소스에 태그 지정

태그는 사용자 또는 AWS가 AWS 리소스에 할당하는 사용자 지정 속성 레이블입니다. 각 태그에는 두 가지 부분이 있습니다.

- 태그 키 (예: `CostCenter`, `Environment` 또는 `Project`) 태그 키는 대/소문자를 구별합니다.
- 태그 값(예: `111122223333` 또는 `Production`)으로 알려진 선택적 필드 태그 값을 생략하는 것은 빈 문자열을 사용하는 것과 같습니다. 태그 키처럼 태그 값은 대/소문자를 구별합니다.

태그는 다음을 지원합니다.

- AWS 리소스를 식별하고 정리합니다. 많은 AWS 서비스가 태그 지정을 지원하므로 다른 서비스의 리소스에 동일한 태그를 할당하여 해당 리소스의 관련 여부를 나타낼 수 있습니다. 예를 들어, EC2 인스턴스에 할당한 EventBridge 규칙에 동일한 태그를 할당할 수 있습니다.
- AWS 비용을 추적합니다. AWS Billing and Cost Management 대시보드에서 이러한 태그를 활성화합니다. AWS는 태그를 사용하여 비용을 분류하고 월별 비용 할당 보고서를 전달합니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 비용 할당 태그 사용을 참조하십시오.

다음 섹션에서는 EventBridge의 태그에 대한 추가 정보를 제공합니다.

EventBridge에서 지원되는 리소스

EventBridge의 다음 리소스는 태그 지정을 지원합니다.

- 규칙
- 이벤트 버스

태그 추가 및 관리에 대한 자세한 내용은 태그 관리 (p. 152) 단원을 참조하십시오.

태그 관리

태그는 리소스의 `Key` 및 `Value` 속성으로 구성됩니다. CloudWatch 콘솔, AWS CLI 또는 EventBridge API를 사용하여 이러한 속성의 값을 추가, 편집 또는 삭제할 수 있습니다. 태그 작업에 대한 자세한 내용은 다음을 참조하십시오.

- Amazon CloudWatch Events API 참조의 TagResource, UntagResource 및 ListTagsForResource
- Amazon CloudWatch CLI Reference의 tag-resource, untag-resource 및 list-tags-for-resource
- 리소스 그룹 사용 설명서의 Tag Editor 작업

태그 이름 지정 및 사용 규칙

다음 기본 이름 지정 및 사용 규칙은 EventBridge 리소스와 함께 태그를 사용할 때 적용합니다.

- 각 리소스는 최대 50개의 태그를 보유할 수 있습니다.
- 각 리소스에 대해 각 태그 키는 고유하며 하나의 값만 가질 수 있습니다.
- 태그 키의 최대 길이는 UTF-8 형식의 유니코드 문자 128자입니다.
- 태그 값의 최대 길이는 UTF-8 형식의 유니코드 문자 256자입니다.
- 허용되는 문자는 UTF-8로 표현할 수 있는 문자, 숫자, 공백 및 . : + = @ _ / -(하이픈) 문자도 있습니다.
- 태그 키와 값은 대/소문자를 구분합니다. 모범 사례는 태그를 대문자로 사용할 것을 전략으로 결정하고 모든 리소스 유형에 대해 일관되게 해당 전략을 구현하는 것입니다. 예를 들어, `Costcenter`, `costcenter` 또는 `CostCenter`를 사용할지 결정하고 모든 태그에 대해 동일한 규칙을 사용합니다. 대소문자가 일치하지 않는 유사한 태그를 사용하지 마십시오.
- `aws:` 접두사는 AWS가 사용하도록 예약되어 있으므로 태그 사용이 금지되어 있습니다. 이 접두사가 지정된 태그 키나 값은 편집하거나 삭제할 수 없습니다. 이 접두사가 지정된 태그는 리소스당 태그 수 제한에 포함되지 않습니다.

Amazon EventBridge 할당량

EventBridge에는 다음과 같은 할당량이 있습니다.

리소스	기본 한도
이벤트 게시 API 요청	PutEvents 작업은 AWS 리전에 따라 제한됩니다. 리전별 PutEvents 할당량 (p. 155)을(를) 참조하십시오. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
다른 모든 API 요청	`PutEvents` 이외의 모든 EventBridge API는 기본적으로 초당 50개의 요청으로 제한됩니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
이벤트 버스	계정당 이벤트 버스가 최대 100개로 제한됩니다.
이벤트 버스 - 기타 할당량	AWS 서비스 또는 다른 AWS 계정으로부터 수신할 수 있는 이벤트 속도에는 제한이 없습니다. `PutEvents` API를 사용하여 사용자 지정 이벤트를 이벤트 버스로 전송하는 경우 `PutEvents` API 할당량 (p. 155)이 적용됩니다. 계정 내 규칙의 대상으로 전송되는 모든 이벤트는 호출 할당량에 포함됩니다. 이벤트 버스의 정책 크기는 10240자로 제한됩니다. 이 정책 크기는 다른 계정에 대한 액세스 권한을 부여할 때마다 증가합니다. `DescribeEventBus` API를 사용하여 현재 정책 및 크기를 확인할 수 있습니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
이벤트 패턴	최대 2,048자.
호출	호출은 규칙과 일치하고 규칙의 대상으로 전송되는 이벤트입니다. 할당량은 리전에 따라 다릅니다. 리전별 호출 할당량 (p. 155)을(를) 참조하십시오. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
ListRuleNamesByTarget	요청에 대해 페이지당 최대 100개로 결과가 제한됩니다.
ListRules	요청에 대해 페이지당 최대 100개로 결과가 제한됩니다.
ListTargetsByRule	요청에 대해 페이지당 최대 100개로 결과가 제한됩니다.
PutTargets	요청당 항목 10개. 규칙당 대상 최대 5개.
RemoveTargets	요청당 항목 10개
규칙	이벤트 버스당 300개. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.

리소스	기본 한도
	할당량 증가를 요청하기 전에 규칙을 검사합니다. 각각 매우 특정한 이벤트와 일치하는 여러 규칙이 있을 수 있습니다. EventBridge의 이벤트 및 이벤트 패턴 (p. 36)에서 더 적은 수의 식별자를 사용하여 범위를 확장해 보십시오. 또한 규칙은 이벤트와 일치할 때마다 여러 대상을 호출할 수 있습니다. 더 많은 대상을 규칙에 추가해 보십시오.
시스템 관리자 Run Command 대상	대상 키 1개 및 대상 값 1개 시스템 관리자 Run Command는 현재 여러 대상 값을 지원하지 않습니다.
Targets	규칙당 대상 최대 5개.

리전별 PutEvents 할당량

Regions	초당 트랜잭션
• 미국 동부(버지니아 북부) • 미국 서부(오레곤) • 미국 동부(오하이오) • 유럽(아일랜드) • 유럽(프랑크푸르트)	PutEvents는 기본적으로 이러한 리전에서 초당 요청이 2,400개로 제한되어 있습니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
• 미국 서부(캘리포니아 북부 지역) • 유럽(런던) • 아시아 태평양(시드니) • 아시아 태평양(도쿄) • 아시아 태평양(싱가포르)	PutEvents는 기본적으로 이러한 리전에서 초당 요청이 1,200개로 제한되어 있습니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
• 캐나다(중부) • 유럽(파리) • 유럽(스톡홀름) • 남아메리카(상파울루) • 아시아 태평양(서울) • 아시아 태평양(뭄바이) • 아시아 태평양(홍콩) • 중동(바레인)	PutEvents는 기본적으로 이러한 리전에서 초당 요청이 600개로 제한되어 있습니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
• 중국(닝샤) • 중국(베이징) • 아시아 태평양(오사카-로컬)	PutEvents는 기본적으로 이러한 리전에서 초당 요청이 400개로 제한되어 있습니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.

리전별 호출 할당량

호출은 규칙과 일치하고 규칙의 대상으로 전송되는 이벤트입니다. 대상 서비스 문제, 계정 조절 등으로 인해 대상 호출이 실패하면 특정 호출에 대해 최대 24시간 동안 새로운 시도가 이루어집니다.

다른 계정에서 이벤트를 수신하는 경우 계정의 규칙과 일치하고 규칙의 대상으로 전송되는 각 이벤트는 계정의 초당 호출 할당량을 기준으로 계산됩니다.

도달한 후 리전 호출에서 다음 호출 할당량 중 하나가 제한됩니다. 계속 발생하지만 지연됩니다.

Regions	초당 호출 수
• 미국 동부(버지니아 북부) • 미국 서부(오레곤) • 미국 동부(오하이오) • 유럽(아일랜드) • 유럽(프랑크푸르트)	이러한 리전에 대한 호출 할당량은 기본적으로 이러한 리전에서 초당 4,500개의 요청으로 제한됩니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
• 미국 서부(캘리포니아 북부 지역) • 유럽(런던) • 아시아 태평양(시드니) • 아시아 태평양(도쿄) • 아시아 태평양(싱가포르)	이러한 리전에 대한 호출 할당량은 기본적으로 이러한 리전에서 초당 2,250개의 요청으로 제한됩니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
• 캐나다(중부) • 남아메리카(상파울루) • 유럽(파리) • 유럽(스톡홀름) • 아시아 태평양(서울) • 아시아 태평양(뭄바이) • 아시아 태평양(홍콩) • 중동(바레인)	이러한 리전에 대한 호출 할당량은 기본적으로 이러한 리전에서 초당 1,100개의 요청으로 제한됩니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.
• 중국(닝샤) • 중국(베이징) • 아시아 태평양(오사카-로컬)	이러한 리전에 대한 호출 할당량은 기본적으로 이러한 리전에서 초당 750개의 요청으로 제한됩니다. 할당량 증가를 요청할 수 있습니다. 지침은 AWS 서비스 할당량을 참조하십시오.

Amazon EventBridge 문제 해결

이 단원에 나와 있는 단계에 따라 Amazon EventBridge에서 문제를 해결할 수 있습니다.

주제

내 규칙이 트리거되었지만 내 Lambda 함수는 호출되지 않았음

Lambda 함수에 대해 올바른 권한이 설정되었는지 확인합니다. AWS CLI를 사용하여 다음 명령을 실행합니다. 함수 이름을 사용자의 함수로 바꾸고 해당 함수가 있는 AWS 리전을 사용하십시오.

```
aws lambda get-policy --function-name MyFunction --region us-east-1
```

다음과 유사한 출력 화면이 표시되어야 합니다.

```
{
    "Policy": "{\"Version\":\"2012-10-17\",
    \"Statement\":[
        {\"Condition\":{\"ArnLike\":{\"AWS:SourceArn\":\"arn:aws:events:us-
east-1:123456789012:rule/MyRule\"}},
        \"Action\":\"lambda:InvokeFunction\",
```

```
        \"Resource\":\"arn:aws:lambda:us-east-1:123456789012:function:MyFunction\",
        \"Effect\":\"Allow\",
        \"Principal\":{\"Service\":\"events.amazonaws.com\"},
        \"Sid\":\"MyId\"}
    ],
    \"Id\":\"default\"}"
}
```

다음과 같은 화면이 나타날 경우:

```
A client error (ResourceNotFoundException) occurred when calling the GetPolicy operation:
 The resource you requested does not exist.
```

또는 이러한 출력 화면은 나타났지만 정책에서 신뢰할 수 있는 개체로서 events.amazonaws.com의 위치를 찾을 수 없는 경우에는 다음 명령을 실행합니다.

```
aws lambda add-permission \
--function-name MyFunction \
--statement-id MyId \
--action 'lambda:InvokeFunction' \
--principal events.amazonaws.com \
--source-arn arn:aws:events:us-east-1:123456789012:rule/MyRule
```

> **Note**
>
> 정책이 올바르지 않은 경우에는 이를 제거하거나 규칙에 다시 추가하는 등 EventBridge 콘솔에서 규칙을 편집할 수도 있습니다. EventBridge 콘솔은 대상에 올바른 권한을 설정하게 됩니다. 특정한 Lambda 별칭 또는 버전을 사용하고 있는 경우에는 `aws lambda get-policy` 및 `aws lambda add-permission` 명령에 `--qualifier` 파라미터를 추가해야 합니다.
>
> ```
> aws lambda add-permission \
> --function-name MyFunction \
> --statement-id MyId \
> --action 'lambda:InvokeFunction' \
> --principal events.amazonaws.com \
> --source-arn arn:aws:events:us-east-1:123456789012:rule/MyRule
> --qualifier alias or version
> ```

Lambda 함수가 트리거에 실패하는 또 다른 이유는 `get-policy`를 실행할 때 나타나는 정책에 `SourceAccount` 필드가 포함되어 있기 때문입니다. `SourceAccount` 설정은 EventBridge가 함수 호출을 할 수 없도록 막습니다.

방금 규칙을 생성/수정했지만, 테스트 이벤트와 일치하지 않습니다.

규칙이나 규칙의 대상을 변경한다고 수신 이벤트가 즉시 새로운 규칙이나 업데이트된 규칙에 대한 매칭을 시작 또는 중지하지는 않습니다. 변경 사항이 적용될 때까지 잠시 기다리십시오. 잠시 후에도 여전히 이벤트가 일치되지 않으면 규칙의 CloudWatch 지표(예: `TriggeredRules`, `Invocations` 및 `FailedInvocations`)를 점검해서 추가적인 디버깅이 필요한지 확인합니다. 이러한 지표에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서의 Amazon CloudWatch Events 지표 및 차원을 참조하십시오.

규칙이 AWS 서비스의 이벤트에 의해 트리거되는 경우 `TestEventPattern` 작업을 사용하여 테스트 이벤트에서 규칙의 이벤트 패턴을 테스트함으로써 규칙의 이벤트 패턴이 올바르게 설정되었는지 확인할 수도 있습니다. 자세한 내용은 Amazon CloudWatch Events API 참조의 TestEventPattern을 참조하십시오.

ScheduleExpression에 지정된 시간에 내 규칙이 자체 트리거 되지 않았음

ScheduleExpressions는 UTC 기준입니다. UTC 시간대에 규칙이 자체 트리거 되도록 예약이 되어 있는지 확인합니다. ScheduleExpression이 올바르면 방금 규칙을 생성/수정했지만, 테스트 이벤트와 일치하지 않습니다. (p. 158)의 단계들을 따릅니다.

> Note
>
> 캐싱에 따라 예약된 규칙의 첫 번째 인스턴스가 삭제될 수 있습니다.
> 규칙이 적용되려면 짧은 시간이 걸릴 수 있습니다. 이 기간 내에 도착하는 트리거는 생성되거나 업데이트된 일정과 일치하지 않을 수 있습니다.

내 규칙이 예상된 시간에 트리거되지 않음

EventBridge는 매 기간마다 실행되는 규칙을 생성할 때 정확한 시작 시간을 설정하도록 지원하지 않습니다. 규칙이 생성되는 즉시 런타임에 대한 카운트다운이 시작됩니다.

Cron 표현식을 사용하여 지정된 시간에 대상을 호출할 수 있습니다. 예를 들어 Cron 표현식을 사용하여 0분도 틀리지 않고 4시간마다 트리거되는 규칙을 생성할 수 있습니다. CloudWatch 콘솔에서는 Cron 표현식 `0 0/4 * * ? *`을, AWS CLI에서 Cron 표현식 `cron(0 0/4 * * ? *)`을 사용했습니다. 예를 들어 AWS CLI를 사용하여 4시간마다 트리거되는 TestRule라는 규칙을 생성하려면 명령 프롬프트에 다음과 같이 입력하면 됩니다.

```
aws events put-rule --name TestRule --schedule-expression 'cron(0 0/4 * * ? *)'
```

`0/5 * * * ? *` Cron 표현식을 사용해서 5분마다 규칙을 트리거할 수 있습니다. 예:

```
aws events put-rule --name TestRule --schedule-expression 'cron(0/5 * * * ? *)'
```

EventBridge는 일정 표현식에 초 단위의 정밀성을 제공하지 않습니다. Cron 표현식을 사용해 가장 정밀하게 설정할 수 있는 단위가 1분입니다. EventBridge와 대상 서비스가 분산되어 있기 때문에 예약된 규칙이 트리거 되는 시간과 대상 서비스가 대상 리소스 실행을 인식하는 시간 간에는 몇 초의 지연이 있을 수 있습니다. 설정된 분 내에 예약된 규칙이 트리거 되지만 초 단위로 설정할 수는 없습니다.

내 규칙이 IAM API 호출과 일치하지만 트리거되지 않았음

IAM 서비스는 미국 동부(버지니아 북부) 지역에서만 사용 가능하기 때문에 IAM에서 나온 모든 AWS API 호출 이벤트는 해당 리전에서만 사용할 수 있습니다. 자세한 내용은 EventBridge 지원되는 AWS 서비스의 이벤트 예제 (p. 57) 단원을 참조하십시오.

규칙이 트리거될 때 규칙과 연관된 IAM 역할이 무시되었기 때문에 내 규칙이 적용되지 않고 있음

규칙에 대한 IAM 역할은 Kinesis 스트림에 이벤트를 연결하는 용도로만 사용됩니다. Lambda 함수와 Amazon SNS 주제에 대해 리소스 기반 권한을 제공해야 합니다.

리전 AWS STS 엔드포인트가 활성화되었는지 확인합니다. 사용자가 제공한 IAM 역할을 맡고 있을 때 EventBridge는 리전 AWS STS 엔드포인트와 통신을 합니다. 자세한 내용은 IAM 사용 설명서의 AWS 리전에서 AWS STS 활성화 및 비활성화를 참조하십시오.

리소스와 일치하는 것으로 추정되는 EventPattern을 통해 규칙을 생성했지만 규칙과 일치하는 어떤 이벤트도 발견하지 못함

AWS의 서비스는 대부분 Amazon 리소스 이름에서 콜론(:)이나 슬래시(/)를 동일한 문자로 처리합니다. 그러나 EventBridge는 이벤트 패턴 및 규칙에서 정확히 일치를 사용합니다. 따라서 이벤트 패턴을 만들 때 일치시킬 이벤트에서 ARN 구문과 일치하도록 정확한 ARN 문자를 사용해야 합니다.

뿐만 아니라 리소스 필드에 내용이 입력되어 있지 않은 이벤트도 있습니다(예: CloudTrail에서의 AWS API 호출 이벤트).

내 이벤트를 대상에 제공할 때 지연을 경험함

대상 리소스가 제한되는 시나리오를 제외하고 EventBridge에서는 최대 24시간 동안 이벤트를 대상에 전달하려고 합니다. 이벤트가 이벤트 스트림에 도착하는 즉시 첫 번째 시도가 이루어집니다. 그러나 대상 서비스에 문제가 있는 경우에는 EventBridge가 향후 또 다른 제공 일정을 자동으로 예약합니다. 이벤트가 도착하고 24시간이 지나면 더 이상 시도가 예약되지 않고 `FailedInvocations` 지표가 CloudWatch에 게시됩니다. `FailedInvocations` 지표에 CloudWatch 경보를 생성하는 것이 좋습니다.

일부 이벤트가 내 대상으로 전달되지 않음

EventBridge 규칙의 대상이 장기간 제한되어 있는 경우 EventBridge는 전달을 재시도하지 않을 수 있습니다. 예를 들어 대상이 수신 이벤트 트래픽을 처리하도록 프로비저닝되어 있지 않으며 대상 서비스가 EventBridge에서 사용자를 대신하여 하는 요청을 조절하는 경우, EventBridge는 전달을 재시도하지 않을 수 있습니다.

한 개의 이벤트에 응답하기 위해 내 규칙이 한 번 이상 트리거됩니다. EventBridge는 규칙을 트리거하거나 대상에 이벤트를 제공하기 위해 어떤 보장을 제공합니까?

드문 경우기는 하지만, 단일 이벤트 또는 예약된 시간에서 동일한 규칙을 한 번 이상 트리거 하거나 트리거된 특정 규칙에서 동일한 대상을 한 번 이상 호출할 수 있습니다.

무한 루프 방지

EventBridge에서는 무한 루프, 즉 반복 실행되는 규칙이 생성될 수 있습니다. 예를 들어 규칙이 S3 버킷에서 ACL이 바뀐 것을 감지할 경우 소프트웨어를 트리거하여 ACL을 원하는 상태로 변경합니다. 이때 규칙이 부주의하게 작성되면 ACL에 대한 변경이 이어져 규칙을 다시 실행하면서 무한 루프에 빠지게 됩니다.

이를 방지하려면 트리거된 작업이 동일한 규칙을 다시 실행하지 못하도록 규칙을 작성해야 합니다. 예를 들어 ACL이 변경 이후가 아니고 잘못된 상태일 때만 규칙이 실행되도록 할 수 있습니다.

무한 루프는 예상보다 높은 요금을 빠르게 야기할 수 있습니다. 따라서 요금이 지정한 한도를 초과할 경우 이를 알려줄 수 있는 예산 관리를 사용하는 것이 좋습니다. 자세한 내용은 예산을 통해 비용 관리 단원을 참조하십시오.

내 이벤트가 대상 Amazon SQS 대기열에 전달되지 않음

Amazon SQS 대기열은 암호화되었을 수 있습니다. 암호화된 Amazon SQS 대기열이 포함된 규칙을 대상으로 생성한 경우 암호화 대기열에 성공적으로 제공된 이벤트에 대한 KMS 키 정책에 다음 섹션이 포함되어야 합니다.

```
{
                "Sid": "Allow CWE to use the key",
                "Effect": "Allow",
                "Principal": {
                                "Service": "events.amazonaws.com"
                },
                "Action": [
                                "kms:Decrypt",
                                "kms:GenerateDataKey"
                ],
                "Resource": "*"
}
```

내 규칙이 트리거 중이지만 내 Amazon SNS 주제에 어떤 메시지도 게시되지 않음

Amazon SNS 주제에 대해 올바른 권한이 설정되었는지 확인합니다. AWS CLI를 사용하여 다음 명령을 실행합니다. 주제 ARN을 사용자의 항목으로 바꾸고 해당 주제가 있는 AWS 리전을 사용하십시오.

```
aws sns get-topic-attributes --region us-east-1 --topic-arn "arn:aws:sns:us-
east-1:123456789012:MyTopic"
```

다음과 유사한 정책 속성이 나타납니다.

```
"{\"Version\":\"2012-10-17\",
\"Id\":\"__default_policy_ID\",
\"Statement\":[{\"Sid\":\"__default_statement_ID\",
\"Effect\":\"Allow\",
\"Principal\":{\"AWS\":\"*\"},
\"Action\":[\"SNS:Subscribe\",
\"SNS:ListSubscriptionsByTopic\",
\"SNS:DeleteTopic\",
\"SNS:GetTopicAttributes\",
\"SNS:Publish\",
\"SNS:RemovePermission\",
\"SNS:AddPermission\",
\"SNS:Receive\",
\"SNS:SetTopicAttributes\"],
\"Resource\":\"arn:aws:sns:us-east-1:123456789012:MyTopic\",
```

```
\"Condition\":{\"StringEquals\":{\"AWS:SourceOwner\":\"123456789012\"}}},{\"Sid\":
\"Allow_Publish_Events\",
\"Effect\":\"Allow\",
\"Principal\":{\"Service\":\"events.amazonaws.com\"},
\"Action\":\"sns:Publish\",
\"Resource\":\"arn:aws:sns:us-east-1:123456789012:MyTopic\"}]}"
```

다음과 유사한 정책이 나타나면 기본 정책만 설정이 된 것입니다.

```
"{\"Version\":\"2008-10-17\",
\"Id\":\"__default_policy_ID\",
\"Statement\":[{\"Sid\":\"__default_statement_ID\",
\"Effect\":\"Allow\",
\"Principal\":{\"AWS\":\"*\"},
\"Action\":[\"SNS:Subscribe\",
\"SNS:ListSubscriptionsByTopic\",
\"SNS:DeleteTopic\",
\"SNS:GetTopicAttributes\",
\"SNS:Publish\",
\"SNS:RemovePermission\",
\"SNS:AddPermission\",
\"SNS:Receive\",
\"SNS:SetTopicAttributes\"],
\"Resource\":\"arn:aws:sns:us-east-1:123456789012:MyTopic\",
\"Condition\":{\"StringEquals\":{\"AWS:SourceOwner\":\"123456789012\"}}}]}"
```

정책의 Publish 권한에 `events.amazonaws.com`이 보이지 않으면 AWS CLI를 사용하여 주제 정책 속성을 설정하십시오.

현재 정책을 복사하고 명령문 목록에 다음 명령문을 추가합니다.

```
{\"Sid\":\"Allow_Publish_Events\",
\"Effect\":\"Allow\",\"Principal\":{\"Service\":\"events.amazonaws.com\"},
\"Action\":\"sns:Publish\",
\"Resource\":\"arn:aws:sns:us-east-1:123456789012:MyTopic\"}
```

앞에서 설명한 정책과 비슷하게 새 정책이 표시됩니다.

AWS CLI로 주제 속성을 설정하십시오.

```
aws sns set-topic-attributes --region us-east-1 --topic-arn "arn:aws:sns:us-
east-1:123456789012:MyTopic" --attribute-name Policy --attribute-value NEW_POLICY_STRING
```

Note

정책이 올바르지 않은 경우에는 이를 제거하거나 규칙에 다시 추가하는 등 EventBridge 콘솔에서 규칙을 편집할 수도 있습니다. EventBridge에서 대상에 올바른 권한을 설정합니다.

Amazon SNS 주제와 연관된 규칙을 삭제했는데도 내 Amazon SNS 주제가 여전히 EventBridge에 대한 권한을 가지고 있음

Amazon SNS를 대상으로 해서 규칙을 생성하면 EventBridge가 사용자를 대신해 Amazon SNS 주제에 권한을 추가합니다. 규칙을 생성하고 조금 후에 규칙을 삭제한 경우에는 EventBridge가 Amazon SNS 주제에서

권한을 제거할 수 없습니다. 이 경우에는 aws sns set-topic-attributes 명령을 사용하여 주제에서 권한을 제거할 수 있습니다. 이벤트 전송을 위한 리소스 기반 권한에 대한 자세한 내용은 EventBridge에서 리소스 기반 정책 사용 (p. 132) 단원을 참조하십시오.

EventBridge에서 사용할 수 있는 IAM 조건 키 유형

EventBridge는 AWS 전체의 조건 키(IAM 사용 설명서의 사용 가능한 키 참조) 및 다음과 같은 서비스별 조건 키를 지원합니다. 자세한 내용은 IAM 정책 조건을 사용하여 세부적인 액세스 제어 구현 (p. 138) 단원을 참조하십시오.

EventBridge 규칙 위반 시 이를 알아챌 수 있는 방법

다음 경보를 사용하여 EventBridge 규칙 위반 시 이를 알릴 수 있습니다.

규칙 위반 시 이를 알리기 위해 경보를 생성하려면

1. https://console.aws.amazon.com/cloudwatch/에서 CloudWatch 콘솔을 엽니다.
2. 경보 생성을 선택합니다. 범주별 CloudWatch 지표 창에서 Events Metrics(이벤트 지표)를 선택합니다.
3. 지표 목록에서 FailedInvocations를 선택합니다.
4. 그래프 위에서 통계, 합계를 선택합니다.
5. 기간으로 5분 등의 값을 선택합니다. 다음을 선택합니다.
6. 경보 임계값 아래에서 이름에 myFailedRules 등 고유한 경보 이름을 입력합니다. 설명에는 Rules are not delivering events to targets(규칙이 대상에 이벤트를 전달하지 않음) 등의 경보 설명을 입력합니다.
7. 조건에 대해 >= 및 1을 선택합니다. 기간에는 10을 입력합니다.
8. 작업의 이 경보가 발생할 경우 항상에서 상태가 ALARM입니다를 선택합니다.
9. 알림 보내기에 대해 기존 Amazon SNS 주제를 선택하거나 새로 만듭니다. 주제를 새로 생성하려면 새 목록을 선택합니다. 새 Amazon SNS 주제의 이름을 입력합니다(예: myFailedRules).
10. ALARM 상태로 경보가 변경되면 알릴 이메일 주소를 쉼표로 구분하여 이메일 목록에 입력합니다.
11. 경보 생성을 선택합니다.

문서 기록

다음 표에는 2019년 7월부터 적용되는 Amazon EventBridge 사용 설명서의 각 릴리스에서 변경된 중요 사항이 나와 있습니다. 이 설명서에 대한 업데이트 알림을 받으려면 RSS 피드를 구독하면 됩니다.

변경 사항	설명	릴리스 날짜
이벤트 버스 태그	이 릴리스에서는 이벤트 버스에 대한 태그를 만들고 관리할 수 있습니다. 이벤트 버스를 만들 때 태그를 추가하고 관련 API를 호출하여 기존 태그를 추가하거나 관리할 수 있습니다. 자세한 정보는 다음을 참조하십시오. • Amazon EventBridge 리소스에 태그 지정 (p. 152) • 태그 기반 정책 (p. 120) • `TagResource` • `UntagResource` • `ListTagsForResource`	2020년 2월 24일
서비스 할당량 증가	Amazon EventBridge는 호출 및 `PutEvents`에 대한 할당량을 증가시켰습니다. 할당량은 리전에 따라 다르며 필요한 경우 늘릴 수 있습니다. • Amazon EventBridge 할당량 (p. 154) • 리전별 PutEvents 할당량 (p. 155) • 리전별 호출 할당량 (p. 155) • 할당량 증가 요청	2020년 2월 11일
대상 입력 변환에 대한 새로운 주제를 추가하고 Application Auto Scaling 이벤트에 대한 링크를 추가했습니다.	입력 변환기에 대한 설명서가 개선되었습니다. • 대상 입력 변환 (p. 47) • 입력 변환기를 사용하여 이벤트에서 데이터를 추출하고 해당 데이터를 대상에 입력 • 자습서: 입력 변환기를 사용하여 이벤트 대상에 전달된 것을 사용자 지정 (p. 19) Application Auto Scaling 이벤트에 대한 링크를 추가했습니다. • Application Auto Scaling 이벤트 및 EventBridge • EventBridge 지원되는 AWS 서비스의 이벤트 예제 (p. 57)	2019년 12월 20일
콘텐츠 기반 필터링	Amazon EventBridge에서는 이제 이벤트 패턴을 사용한 콘텐츠 기반 필터링을 지원합니다. 자세한 내용은 이벤트 패턴을 사용한 콘텐츠 기반 필터링 (p. 42) 단원을 참조하십시오.	2019년 12월 19일
Amazon Augmented AI 이벤트 예제에 대한	Amazon Augmented AI에 대한 예제 이벤트를 제공하는 Amazon SageMaker 개발자 안내서에서 Amazon Augmented AI 주제에 대한 링크가 추가되었습니다. 자세한 정보는 다음을 참조하십시오.	2019년 12월 13일

변경 사항	설명	릴리스 날짜
링크가 추가되었습니다.	• Amazon Augmented AI에서 이벤트 사용 • EventBridge 지원되는 AWS 서비스의 이벤트 예제 (p. 57)	
Amazon Chime 이벤트 예제에 대한 링크가 추가되었습니다.	해당 서비스에 대한 예제 이벤트를 제공하는 Amazon Chime 주제에 대한 링크가 추가되었습니다. 자세한 정보는 다음을 참조하십시오. • EventBridge를 사용한 Amazon Chime 자동화 • EventBridge 지원되는 AWS 서비스의 이벤트 예제 (p. 57)	2019년 12월 12일
Amazon EventBridge 스키마	이제 Amazon EventBridge에서 스키마를 관리하고 이벤트에 대한 코드 바인딩을 생성할 수 있습니다. 자세한 정보는 다음을 참조하십시오. • Amazon EventBridge 스키마 레지스트리 (p. 50) • EventBridge 스키마 API 참조 • AWS CloudFormation의 EventSchemas 리소스 유형 참조	2019년 12월 1일
이벤트 버스에 대한 AWS CloudFormation 지원	AWS CloudFormation는 이제 EventBus 리소스를 지원합니다. 또한 EventBusPolicy 및 Rule 리소스 모두에서 EventBusName 파라미터를 지원합니다. 자세한 내용은 Amazon EventBridge 리소스 유형 참조를 참조하십시오.	2019년 10월 7일
새로운 서비스	Amazon EventBridge의 최초 릴리스입니다.	2019년 7월 11일

AWS Glossary

For the latest AWS terminology, see the AWS Glossary in the AWS General Reference.