

## Elastic Load Balancing (ELB)

수신되는 애플리케이션 트래픽을 여러 Amazon EC2 인스턴스, 컨테이너 및 IP 주소에 걸쳐 분산하는 관리형 로드 밸런싱 서비스.

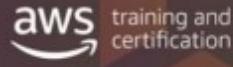
The diagram shows a user icon with a blue arrow pointing to an orange rectangular icon labeled 'ELB'. From the ELB icon, three blue arrows branch out to three separate orange square icons, each containing the Korean character '앱' (App).

Elastic Load  
Balancing

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

웹 계층의 기반은 아키텍처에서 ELB의 사용을 포함합니다. 이러한 로드 밸런서는 EC2 인스턴스로 트래픽을 전송할 뿐만 아니라, AWS가 제공하는 관리형 모니터링 서비스인 Amazon CloudWatch로 지표를 전송할 수 있습니다. Amazon EC2 및 ELB의 지표는 트리거의 역할을 할 수 있습니다. 따라서 자연 시간이 유난히 길거나 AWS 서버 사용률이 지나치게 높아지고 있음을 알게 되는 경우, Auto Scaling을 활용하여 AWS 웹 서버 집합에 용량을 추가할 수 있습니다.

## ELB: 기능



**Elastic Load Balancing**

- HTTP, HTTPS, TCP, UDP 및 SSL(보안 TCP, UDP) 프로토콜을 사용합니다.
- 외부 또는 내부에 위치할 수 있습니다.
- 각 로드 밸런서에 DNS 이름이 부여됩니다.
- 비정상 인스턴스를 인식하고 이에 대응합니다.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

ELB는 수신되는 애플리케이션 트래픽을 Amazon EC2 인스턴스, 컨테이너, IP 주소 등 여러 대상으로 자동 분산시킵니다. 단일 가용 영역 또는 여러 가용 영역에서 애플리케이션 트래픽의 다양한 로드를 처리할 수 있습니다. ELB가 제공하는 세 가지 로드 밸런서는 모두 애플리케이션의 내결함성에 필요한 고가용성, 자동 확장/축소, 강력한 보안을 갖추고 있습니다.

## ELB: 옵션

The slide features a large watermark 'NOT COPY' diagonally across the center. In the top right corner is the AWS logo with the text 'training and certification'. The main content area has a blue header 'Application Load Balancer' containing a circular icon with 'HTTP' and 'HTTPS' text. Below this is a bulleted list:

- 유연한 애플리케이션 관리
- HTTP 및 HTTPS 트래픽의 고급 로드 밸런싱
- 요청수준(계층 7)에서 운영됨
- (계층 7)

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

ELB는 Application Load Balancer, Network Load Balancer, Classic Load Balancer 등 세 가지 유형의 로드 밸런서를 지원합니다. 애플리케이션 요구 사항에 따라 로드 밸런서를 선택할 수 있습니다.

**애플리케이션 로드 밸런서는 개방형 시스템 간 상호 연결(OSI) 모델의 일곱 번째 계층인 애플리케이션 계층에서 작동합니다.** Application Load Balancer는 콘텐츠 기반 라우팅을 지원하고 컨테이너에서 실행되는 애플리케이션을 지원합니다. 이들은 HTTP 또는 HTTPS를 통해 그리고 HTTPS 리스너를 사용하는 HTTP/2를 통해 기본 Web Socket을 지원합니다. 또한 대상이 EC2 인스턴스이든 컨테이너이든 관계없이 그 상태를 확인합니다. EC2 인스턴스 또는 컨테이너에서 실행되는 웹 사이트 및 모바일 앱은 Application Load Balancer를 사용하는 이점을 누릴 수 있습니다.

**Network Load Balancer는 사용자가 아무런 조치를 하지 않아도 높은 처리량과 매우 짧은 지연 시간을 유지하면서 초당 수천만 개의 요청을 처리하도록 설계되었습니다.** 클라이언트로부터 수신되는 트래픽을 수락하고 이 트래픽을 **동일한** 가용 영역 내 대상 전체로 분산합니다. Network Load Balancer는 연결 수준(계층 4)에서 작동하며 IP 프로토콜 데이터에 따라 연결을 대상, 즉 Amazon EC2 인스턴스, 컨테이너 및 IP 주소로 라우팅합니다. Network Load Balancer는 완전 프로그래밍 방식의 대상 그룹 및 대상 제어를 포함해 Application Load Balancer와 API 호환이 가능합니다.

Network Load Balancer는 TCP, UDP 트래픽을 로드 밸런싱하는 데 적합합니다.  
Network Load Balancer는 가용 영역당 하나의 정적 IP 주소를 사용하면서  
갑작스럽고 변동이 심한 트래픽 패턴을 처리하는 데 최적화되어 있습니다.

Classic Load Balancer는 여러 가용 영역의 EC2 인스턴스 사이에서 기본적인 로드  
밸런싱을 제공하며, OSI의 요청 수준 및 연결 수준 모두에서 작동합니다.

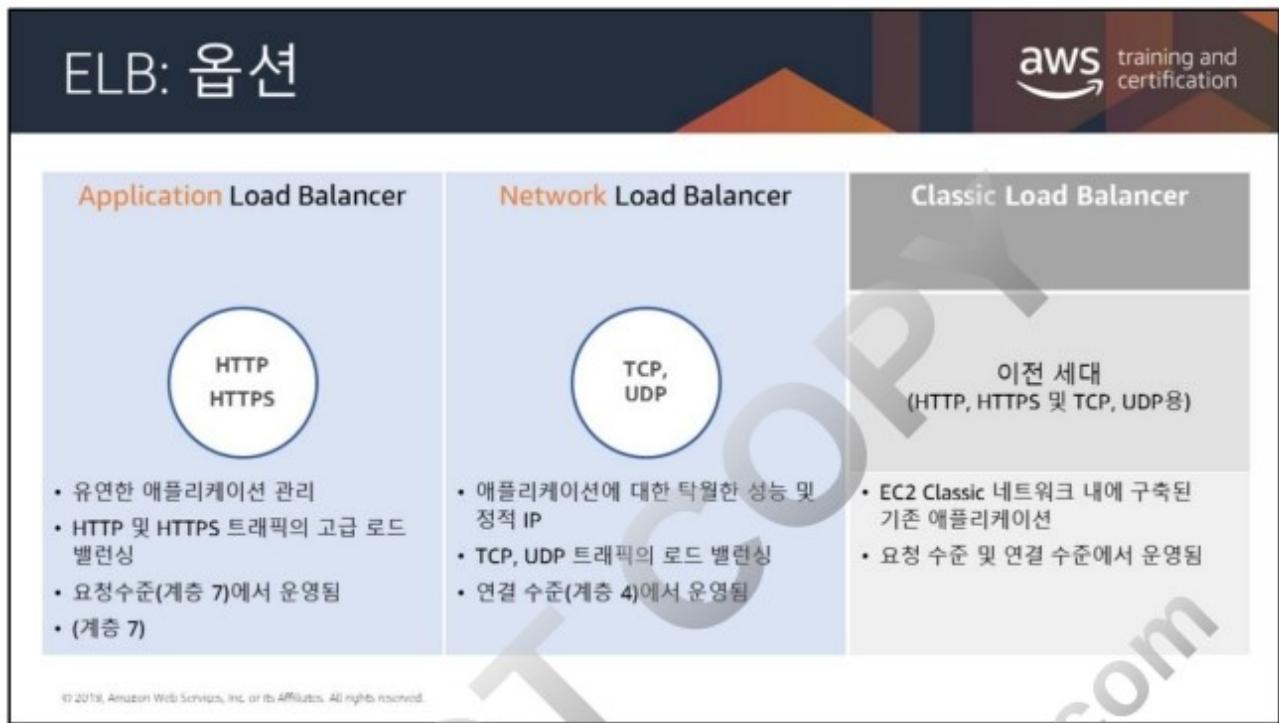
DO NOT COPY  
zlagusdbs@gmail.com

## ELB: 옵션

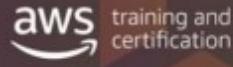
The diagram compares two types of AWS Load Balancers:

- Application Load Balancer**: Handles **HTTP** and **HTTPS** traffic. It manages application-level requests and can route them to multiple back-end servers. It is typically used at the Application Layer (Layer 7) of the OSI model.
- Network Load Balancer**: Handles **TCP** and **UDP** traffic. It routes raw network traffic between multiple back-end servers. It is typically used at the Transport Layer (Layer 4) of the OSI model.

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



## ELB를 사용해야 하는 이유



The slide features four icons with corresponding Korean labels below them:

- 고가용성 (High Availability): Represented by a stack of three green squares.
- 상태 확인 (Health Check): Represented by a white medical kit with a red cross.
- 보안 기능 (Security Features): Represented by a person icon holding a shield.
- TLS 종료 (TLS Termination): Represented by a blue wallet with a gear and a circular arrow icon.

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

### 고가용성

ELB는 트래픽을 단일 가용 영역 또는 여러 가용 영역에 있는 여러 대상(Amazon EC2 인스턴스, 컨테이너, IP 주소)에 자동으로 분산합니다.

### 상태 확인

Amazon EC2 인스턴스의 가용성을 확인하기 위해 로드 밸런서는 주기적으로 핑을 보내거나, 연결을 시도하거나, 요청을 전송하여 Amazon EC2 인스턴스를 테스트합니다. 이러한 테스트를 상태 확인이라고 부릅니다. 등록된 각 Amazon EC2 인스턴스가 상태 확인의 대상에 HTTP 상태 코드 200으로 응답해야 로드 밸런서가 인스턴스를 정상으로 간주합니다.

### 보안 기능

Amazon VPC 내에 프로비저닝된 ELB 로드 밸런서는 보안 그룹과 같은 네트워크 보안 그룹을 활용할 수 있습니다.

### 전송 계층 보안 종료

ELB는 통합 인증 관리 및 SSL 복호화를 지원하여 사용자가 로드 밸런서의 SSL 설정을 중앙 집중식으로 관리하고 애플리케이션으로부터 CPU 집약적 작업을 오프로드할 수 있는 유연성을 제공합니다.

### 계층 4 또는 계층 7 로드 밸런싱

계층 7 전용 기능에 대해 HTTP/HTTPS 애플리케이션을 로드 밸런싱하거나 TCP, UDP 프로토콜에만 의존하는 애플리케이션에 대해 엄격한 계층 4 로드 밸런싱을 사용할 수 있습니다.

일목요연한 기능 비교는

<https://aws.amazon.com/elasticloadbalancing/details/#compare>를 참조하십시오.

## 등록 취소 지연

프로덕션 플릿에서 인스턴스를 제거해야 하지만 사용자에게 영향을 미치지 않으려는 경우:

영향을 받는 백엔드 인스턴스는 등록 취소 전에 진행 중인 요청을 완료합니다.

등록 취소 지연 활성화

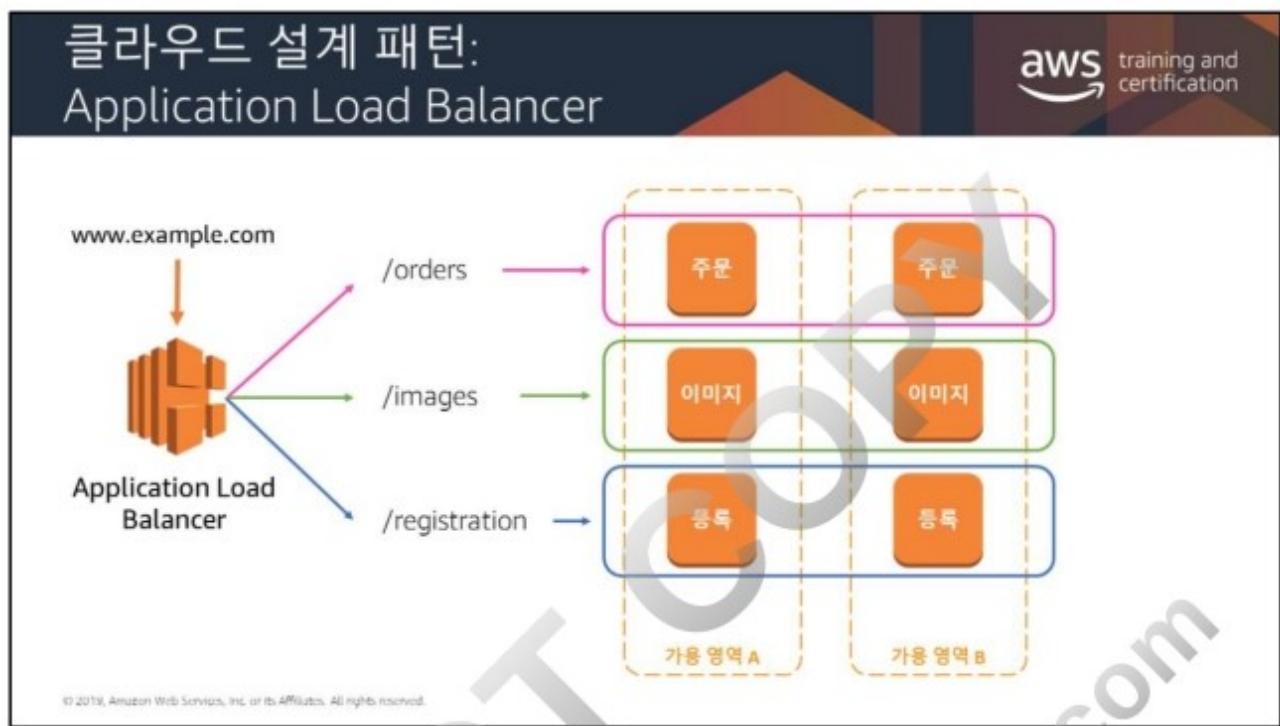
© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

로드 밸런서에서 등록 취소 지연을 활성화하면, 등록이 취소될 백엔드 인스턴스는 등록 취소 전에 진행 중인 요청을 먼저 완료합니다. 마찬가지로 백엔드 인스턴스가 상태 확인에 실패할 경우, 로드 밸런서는 비정상 인스턴스에 새 요청을 보내지 않습니다. 이를 통해 진행 중인 요청을 계속 처리하면서 기존 요청을 완료할 수 있습니다. 즉, 고객 경험에 영향을 주지 않고 소프트웨어 업그레이드 배포 또는 백엔드 인스턴스 교체와 같은 유지 관리 작업을 수행할 수 있습니다.

등록 취소 지연은 또한 Auto Scaling과 통합되므로 로드 밸런서 뒤에서 용량을 훨씬 쉽게 관리할 수 있습니다. 등록 취소 지연이 활성화되면 Auto Scaling은 처리 중인 요청이 완료되길 기다렸다가 인스턴스를 종료합니다.

### 자세한 내용은

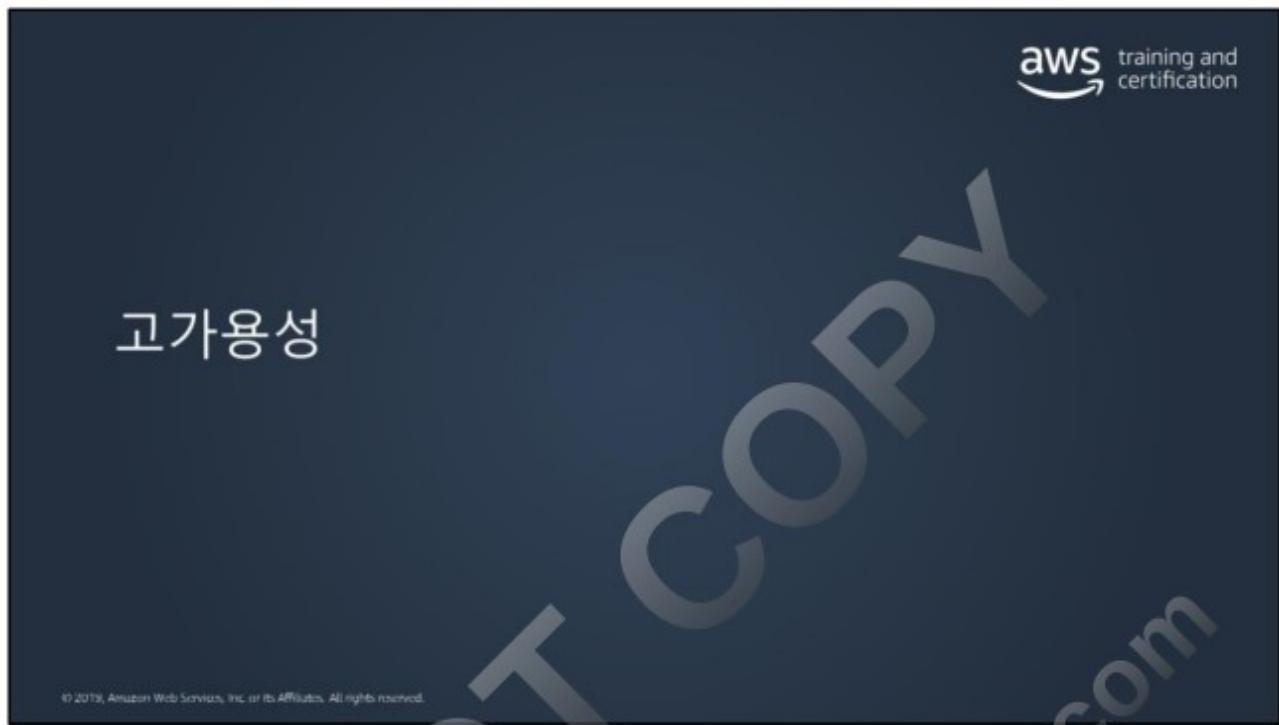
<https://docs.aws.amazon.com/elasticloadbalancing/latest/application/load-balancer-target-groups.html#deregistration-delay>를 참조하십시오(내용 아래 등록 취소 지연 클릭).



자세한 내용은 다음을 참조하십시오.

<https://aws.amazon.com/blogs/devops/introducing-application-load-balancer-unlocking-and-optimizing-architectures/>

이제 Application Load Balancer가 고급 요청 라우팅 기능을 지원합니다. 다음을 참조하십시오. <https://aws.amazon.com/about-aws/whats-new/2019/03/application-load-balancers-now-support-advanced-request-routing/>



## 고가용성이란 무엇일까요?

애플리케이션은 허용되는 성능 저하 시간 내에 장애로부터 복구하거나 보조 소스로 이동할 수 있습니다.

가동률	연간 최대 가동 중간 시간	일일 기준 가동 중단 시간
90%	36.5일	2.4시간
99%	3.65일	14분
99.9%	8.76시간	86초
99.99%	52.6분	8.6초
99.999%	5.25분	0.86초

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

## 고가용성 예제

aws training and certification

모든 기능은 장애가 발생한다는 가정하에 역방향으로 설계합니다.

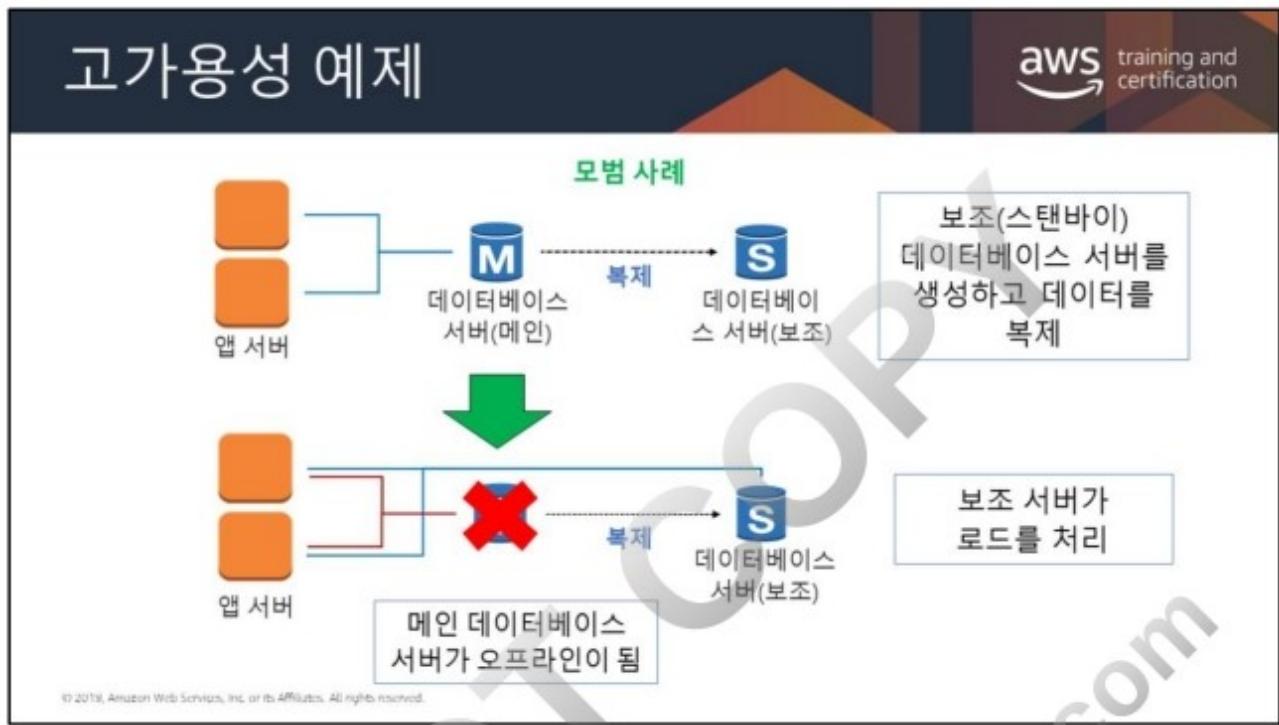
가능한 모든 지점에서 중복성을 구현하여, 단일 장애로 인해 전체 시스템이 중단되지 않도록 합니다.

애플리케이션 서버

데이터베이스 서버

안티 패턴

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



## 얼마나 많은 가용 영역을 사용해야 합니까?

**aws** training and certification

AWS 리전당 2개의 가용 영역을 시작합니다.

한 가용 영역의 리소스에 접근할 수 없더라도 애플리케이션에 장애가 발생해서는 안 됩니다.

The diagram illustrates a VPC (Virtual Private Cloud) represented by a dashed orange rectangle. Inside the VPC, there are two orange boxes labeled 'EC2'. Below these boxes, the text '가용 영역 A' and '가용 영역 B' indicates two separate availability zones within the VPC. Above the VPC, a grey cloud icon represents the '인터넷 게이트웨이' (Internet Gateway), which is connected to the VPC. This setup ensures that even if one availability zone fails, traffic can still be directed through the other.

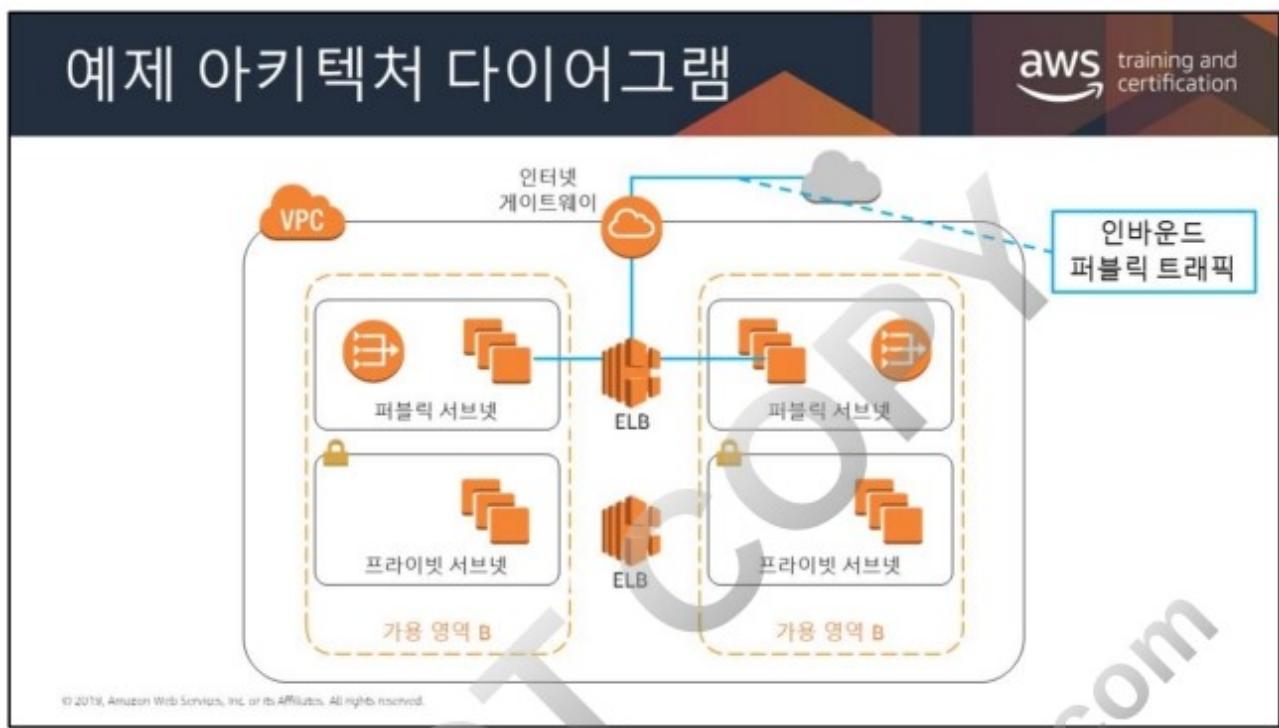
© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

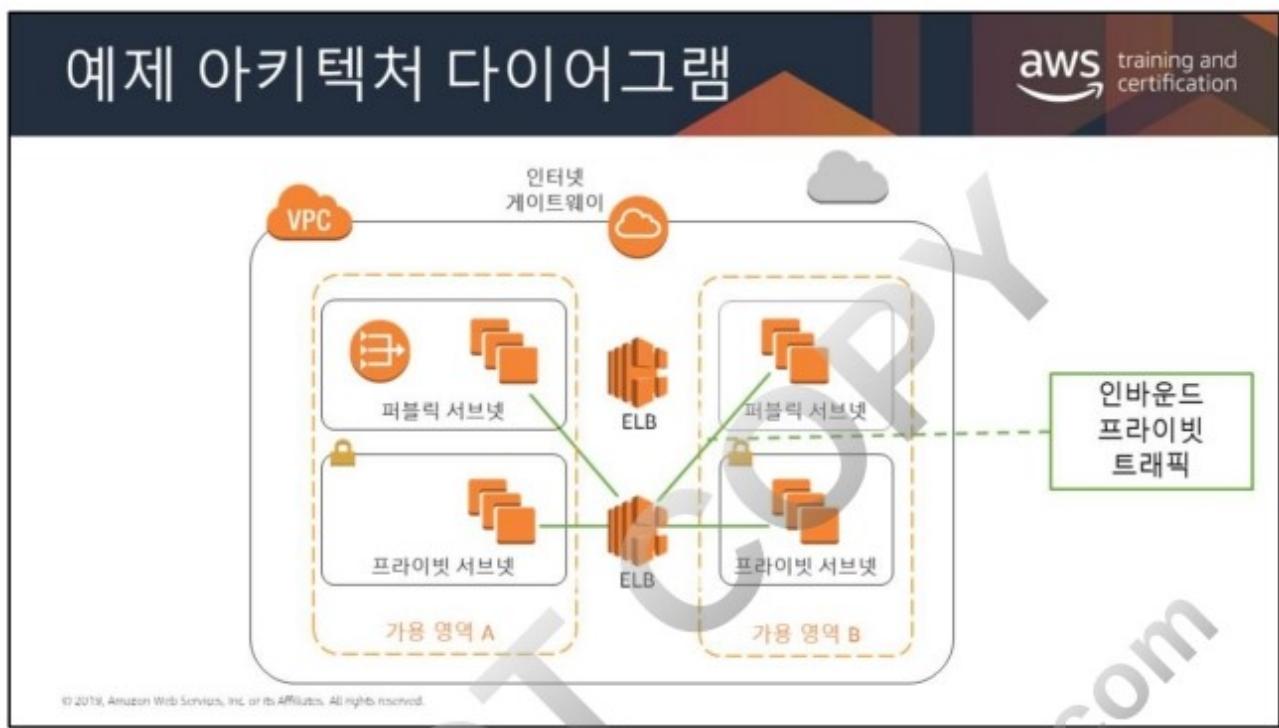
대부분의 애플리케이션은 두 개의 가용 영역을 지원하도록 설계할 수 있습니다. 기본/보조 장애 조치만 지원하는 데이터 소스를 사용할 경우 이보다 많은 가용 영역을 사용하더라도 도움이 되지 않을 수 있습니다. 가용 영역은 물리적으로 분산되어 있으므로 한 AWS 리전에서 3개 이상의 가용 영역에 리소스를 복제할 때 누릴 수 있는 이점은 별로 없습니다.

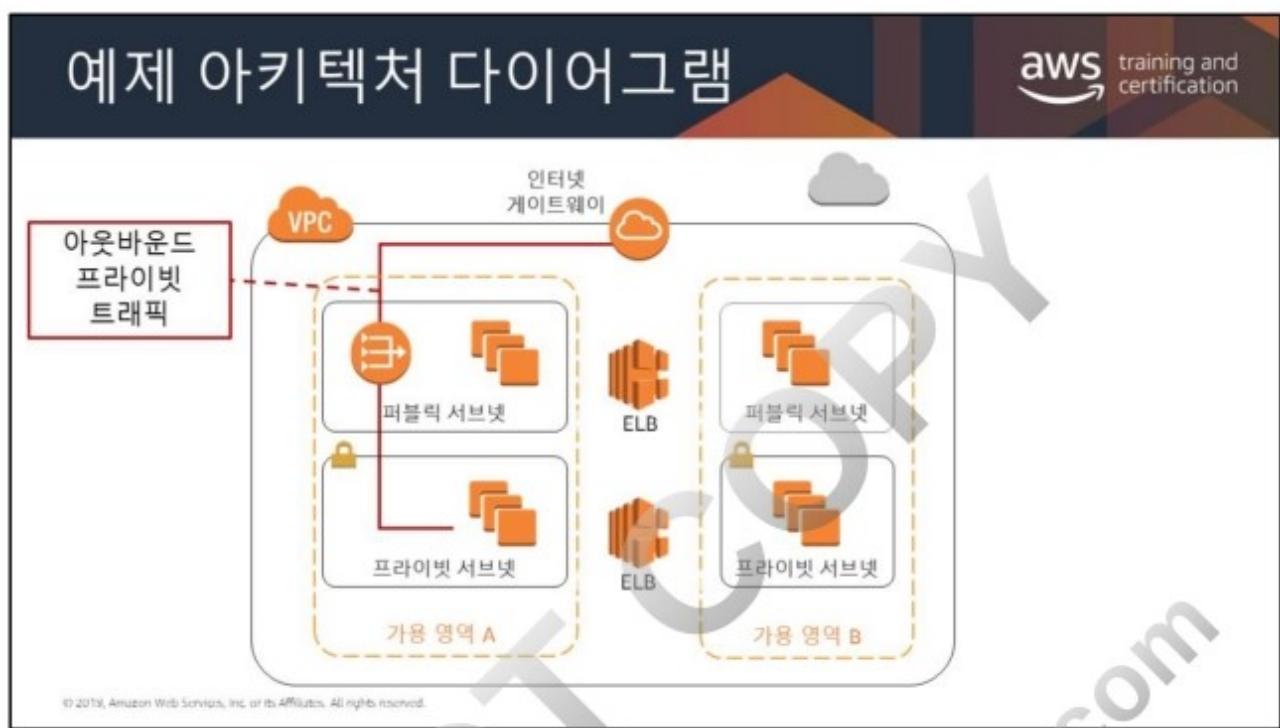
Amazon EC2 스팟 인스턴스 사용량이 많거나 Amazon DynamoDB와 같이 액티브/패시브를 넘어서는 데이터 소스의 경우, 2개를 초과하는 가용 영역을 사용하는 이점이 있을 수 있습니다.

이 기본 패턴에서는 2개의 웹 서버(Amazon EC2)가 ELB 로드 밸런서 뒤에 위치하며, 이 로드 밸런서가 서버 간에 트래픽을 분산합니다. 서버 중 하나에 장애가 발생하면, 로드 밸런서가 이를 인식합니다. 로드 밸런서는 비정상 인스턴스로 트래픽을 분산하는 작업을 중단합니다. 이렇게 하면 구성 요소가 상주하는 가용 영역 중 하나에 문제가 발생하더라도 애플리케이션을 계속 사용할 수 있습니다.

다른 방법을 사용해 인프라의 가용성을 더 높일 수도 있습니다. 이러한 방법은 이후 모듈에서 다룹니다.









## Amazon Route 53

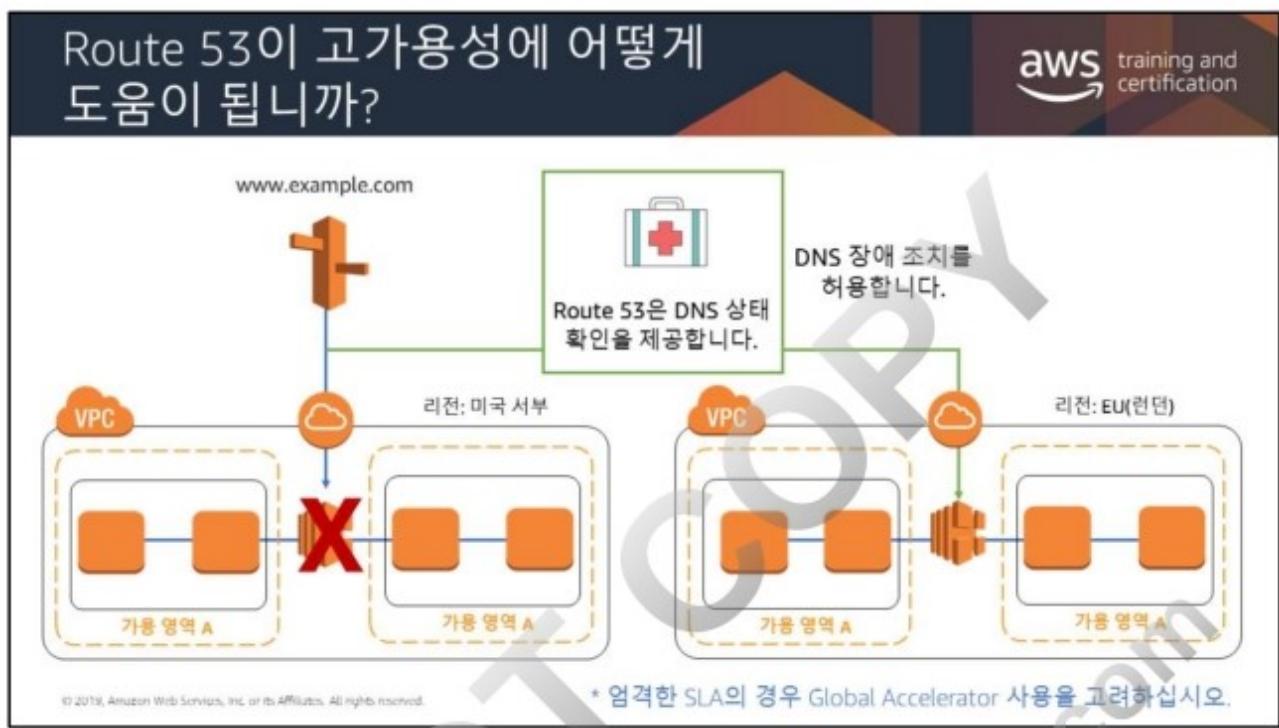


Route 53은 가용성과 확장성이 뛰어난 클라우드 Domain Name System (DNS) 서비스입니다.

- DNS는 도메인 이름을 IP 주소로 변환합니다.
- 도메인 이름을 구입하여 관리하고 DNS 설정을 자동으로 구성할 수 있습니다.
- AWS에서 유연한 고성능, 고가용성 아키텍처를 위한 도구를 제공합니다.
- 멀티플 라우팅 옵션

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Amazon Route 53은 도메인 이름 시스템(DNS), 도메인 이름 등록 및 웹 서비스 상태 확인을 제공합니다. 이 서비스는 최종 사용자를 인터넷 애플리케이션으로 라우팅할 수 있는 안정적이고 비용 효율적인 방법을 개발자와 기업에 제공하기 위해 설계되었습니다. 이 서비스는 *example.com*과 같은 이름을 192.0.2.1과 같이 컴퓨터 간 연결에 사용되는 IP 주소로 변환합니다. DNS를 상태 확인 서비스와 결합하여 정상적인 엔드포인트로 트래픽을 라우팅하거나 개별적으로 엔드포인트에 대한 모니터링 또는 경보를 설정할 수 있습니다. *example.com*과 같은 도메인 이름을 구매 및 관리하고 도메인에 대한 DNS 설정을 자동으로 구성할 수도 있습니다. Route 53은 사용자 요청을 Amazon EC2 인스턴스, ELB 로드 밸런서 또는 Amazon S3 버킷처럼 AWS에서 실행되는 인프라에 효과적으로 연결하며, 사용자를 AWS 외부의 인프라로 라우팅하는 데에도 사용할 수 있습니다.



SLA가 엄격하거나 애플리케이션에 최대한 빠른 장애 조치가 필요한 경우, 아키텍처에 Global Accelerator를 추가하는 것을 고려하십시오.

Global Accelerator는 장애 조치에서 DNS의 역할을 제거하여 네트워크의 복원력을 높입니다. 또한 사용자와 애플리케이션을 캐싱 문제에서 보호하고 거의 즉각적으로 트래픽을 정상적인 엔드포인트로 리디렉션 할 수 있습니다. 또한 아키텍처에 추가하는 새 엔드포인트는 DNS 전파를 기다리지 않고 즉시 트래픽을 수신할 수 있습니다.

Anycast부터 AWS 엣지 로케이션까지 정적 IP 주소를 사용하는 Global Accelerator는 고정 진입점 주소를 제공하여 사용자와 가장 가까운 엣지 로케이션에서 트래픽을 수신하도록 합니다.

## Route 53 라우팅 옵션

간단한 라운드 로빈  
가중치 기반 라운드 로빈  
지연 시간 기반 라우팅  
상태 확인 및 DNS 장애 조치  
지리 위치 라우팅  
트래픽 바이어스를 통한 지리 근접 라우팅  
다중 값 응답

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

**단순 라우팅(라운드 로빈)**은 다수의 요청을 모든 참여 서버로 최대한 균일하게 분산합니다.

**가중치 기반 라운드 로빈**에서는 각 응답이 처리되는 빈도를 지정하기 위해 리소스 레코드 세트에 가중치를 할당할 수 있습니다. 이 기능을 사용하면 소프트웨어를 변경한 서버로 소규모 트래픽을 전송하여 A/B 테스트를 수행할 수 있습니다. 예를 들어 하나의 DNS 이름과 연결된 두 개의 레코드 세트가 있다고 가정해 보겠습니다. 하나에는 가중치 3, 다른 하나에는 가중치 1을 부여합니다. 이 경우, Amazon Route 53이 75%까지 가중치 3 레코드 세트를 반환하고 25%는 가중치 1 레코드 세트를 반환합니다. 가중치는 0부터 255 사이의 숫자로 지정할 수 있습니다.

**지연 시간 기반 라우팅(LBR)**을 사용하면 전 세계 사용자를 대상으로 애플리케이션의 성능을 향상할 수 있습니다. LBR은 애플리케이션이 실행되고 있는 여러 AWS 리전의 실제 성능 측정치를 기준으로 가장 빠른 환경을 제공하는 AWS 엔드포인트(예: Amazon EC2 인스턴스, 탄력적 IP 주소 또는 로드 밸런서)로 고객을 라우팅합니다.

**Amazon Route 53 상태 확인**은 웹 애플리케이션, 웹 서버 및 기타 리소스의 상태와 성능을 모니터링합니다. 상태 확인을 각각 생성하여 다음 중 하나를 모니터링할 수 있습니다.

- 지정한 리소스(예: 웹 서버)의 상태
- 다른 상태 확인의 상태
- Amazon CloudWatch 경보의 상태

상태 확인을 생성하면 상태 확인의 상태를 받고, 상태가 변경될 때 알림을 받으며, DNS 장애 조치를 구성할 수 있습니다.

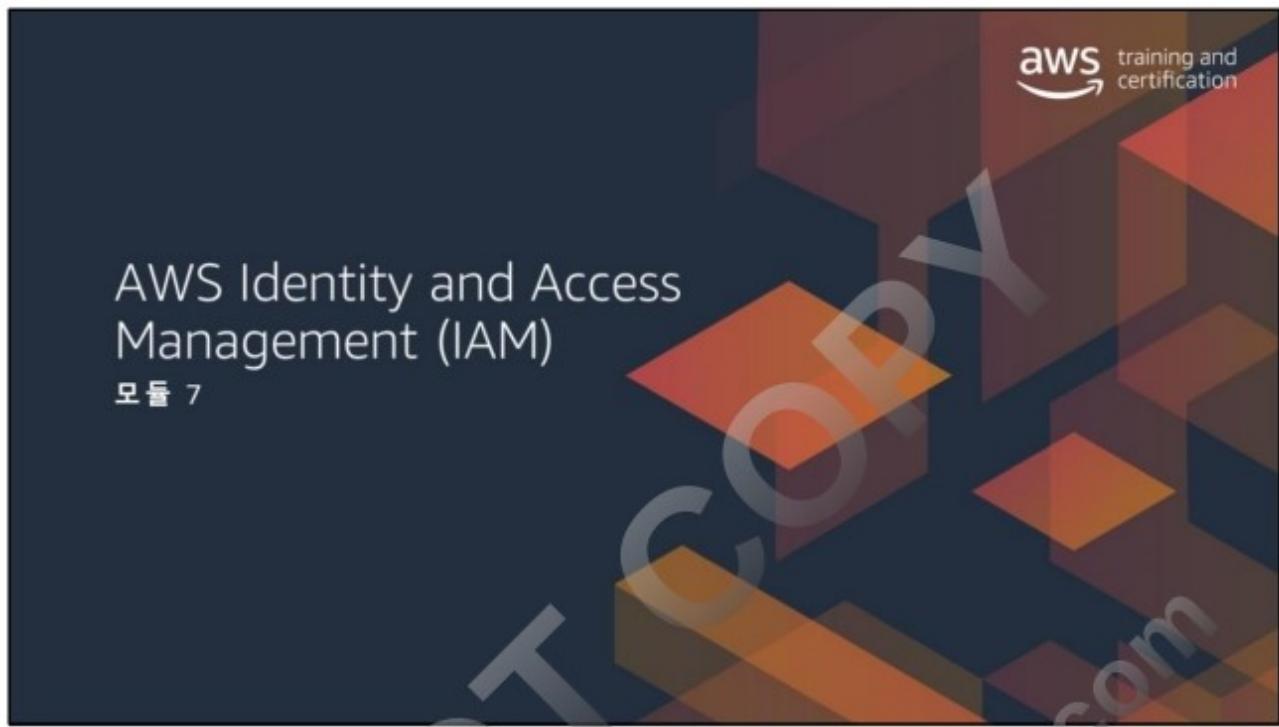
**지리 위치 라우팅**을 사용하면 사용자의 지리적 위치(DNS 쿼리의 오리진)에 따라 트래픽을 지원할 리소스를 선택할 수 있습니다. 지리 위치 라우팅을 사용할 때 콘텐츠를 현지화하고 웹 사이트의 일부 또는 전체를 사용자의 언어로 표시할 수 있습니다. 또한, 지리 위치 라우팅을 사용하여 배포 권한이 있는 위치에만 콘텐츠를 배포하도록 제한할 수 있습니다. 또한 예측 가능하고 관리가 쉬운 방법으로 엔드포인트 간에 로드를 분산함으로써, 각 최종 사용자 위치를 일관되게 동일한 엔드포인트로 라우팅할 수 있습니다.

**DNS 장애 조치**의 경우, Amazon Route 53은 웹 사이트의 가동 중단을 탐지하고 최종 사용자를 애플리케이션이 제대로 작동하는 대체 위치로 리디렉션할 수 있습니다. 이 기능을 활성화하면, Amazon Route 53 상태 확인 에이전트가 가용성을 확인하기 위해 애플리케이션의 각 위치/엔드포인트를 모니터링합니다. 이 기능을 활용하여 고객 사용 애플리케이션의 가용성을 높일 수 있습니다.

**지리 근접 라우팅**은 Route 53 트래픽 흐름을 사용할 경우 사용자와 리소스 사이의 물리적 거리를 기반으로 트래픽을 라우팅할 수 있게 해줍니다. 또한 양 또는 음의 바이어스를 지정하여 각 리소스로 라우팅되는 트래픽을 증감할 수도 있습니다. 트래픽 흐름 정책을 생성할 때 AWS 리전(AWS 리소스를 사용하는 경우) 또는 각 엔드포인트의 위도 및 경도를 지정할 수 있습니다.

**다중 값 응답을 사용하면**, 트래픽을 거의 무작위적으로 웹 서버 같은 다수의 리소스로 라우팅하려는 경우 각 리소스마다 하나씩 다중 값 응답 레코드를 생성하고, 선택적으로 Amazon Route 53 상태 확인을 각 레코드에 연결할 수 있습니다. 예를 들어, 각각 자체 IP 주소를 갖는 12개의 웹 서버로 HTTP 웹 서비스를 관리하는 경우를 생각해보겠습니다. 어떤 웹 서버도 단독으로 모든 트래픽을 처리할 수는 없습니다. 하지만 다중 값 응답 레코드를 생성할 경우, Amazon Route 53이 최대 8개의 정상 레코드로 각 DNS 쿼리에 응답합니다. Amazon Route 53은 DNS 해석기마다 다른 응답을 제공합니다. 해석기가 응답을 캐시한 후 한 웹 서버가 사용 불가능해질 경우 클라이언트 소프트웨어는 응답에 포함된 다른 IP 주소를 시도할 수 있습니다.





## 모듈 7



### 아키텍처 측면에서의 필요성

팀원이 전문적인 역할을 맡고 있을 만큼 충분히 큰 규모의 조직입니다. 필수 권한을 통한 보호 및 액세스 제어 기능이 필요합니다.

#### 모듈 개요

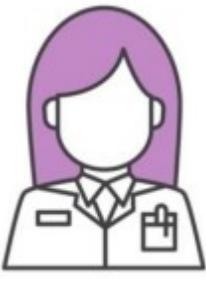
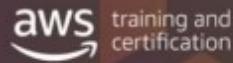
- IAM 사용자, 그룹 및 역할
- 연동 자격 증명 관리
- Amazon Cognito
- AWS Organizations

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

- 어떤 사용자는 AWS Management Console에 액세스해야 하고, 또 어떤 사용자는 AWS 명령줄 인터페이스(AWS CLI)를 사용해야 합니다.
- 각 환경(개발/테스트/프로덕션)은 액세스 요구 사항이 서로 다릅니다.
- 온프레미스 인증은 SSO 자격 증명 연동을 통해 관리됩니다.
- 보안 운영 팀은 AWS 클라우드에서 누가 데이터에 손을 대는지 확인할 수 있어야 합니다.
- 외부 감사자는 로그에만 액세스해야 하며 다른 어떤 것도 액세스할 수 없어야 합니다.
- 모바일 앱은 수천 명의 사용자를 인증해야 합니다.



## AWS 계정 루트 사용자



이 계정은 **모든** AWS 서비스 및 리소스에 대한 **전체 액세스 권한**을 갖습니다.

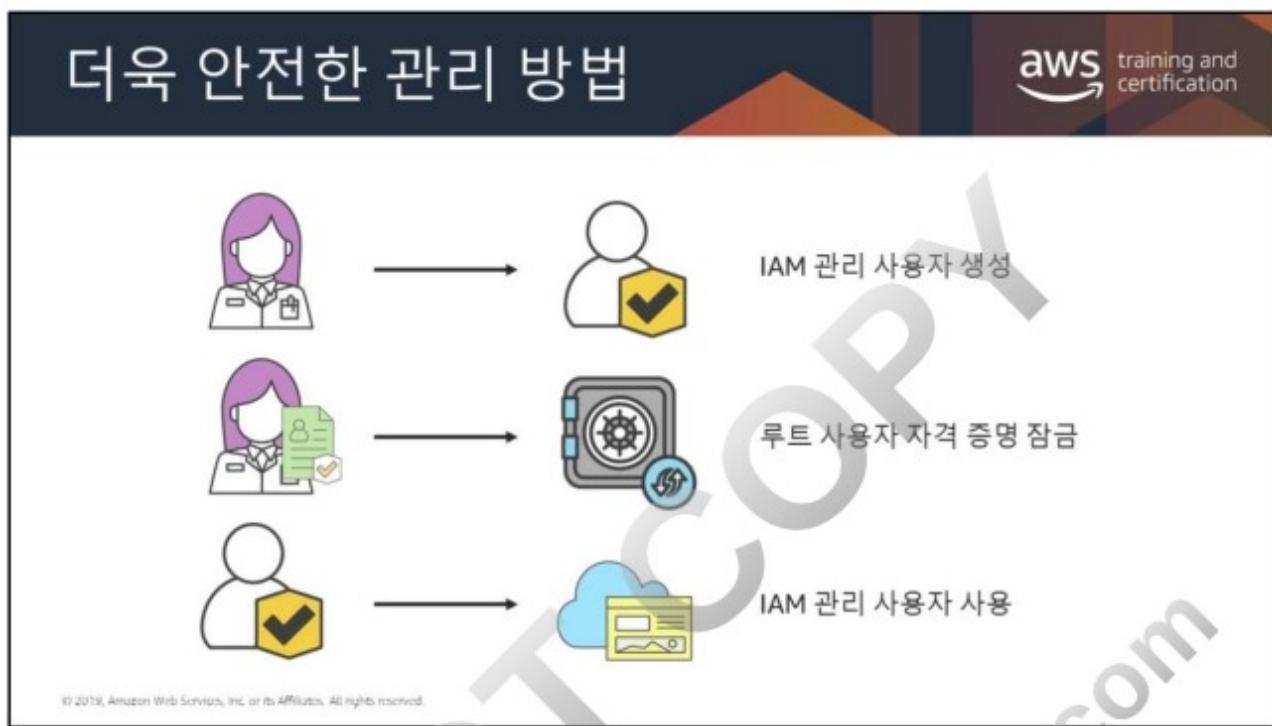
- 결제 정보
- 개인 데이터
- 전체 아키텍처 및 해당 구성 요소

**AWS 계정 루트 사용자는 강력한 권한을 가지며 제한을 받지 않습니다.**

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS 계정을 처음 생성할 때 루트 사용자로 시작합니다. 이 사용자는 계정의 모든 AWS 서비스 및 리소스에 대한 전체 액세스 권한을 가집니다. 이 자격증명은 AWS 계정 루트 사용자라고 하며, 계정을 생성할 때 제공한 이메일 주소 및 암호로 로그인하여 액세스합니다.

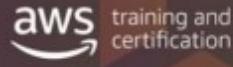
AWS 계정 루트 사용자에 대한 자세한 내용은 [https://docs.aws.amazon.com/IAM/latest/UserGuide/id\\_root-user.html](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_root-user.html)를 참조하십시오.



AWS 계정 루트 사용자는 계정의 모든 리소스에 대한 전체 액세스 권한을 가지며, 사용자는 루트 계정 자격 증명의 권한을 제어할 수 없습니다.

AWS와의 일상적인 상호 작용에는 루트 계정 자격 증명을 사용하지 않는 것이 좋습니다. IAM 사용자는 비교적 쉽게 관리될 수 있으며 감사될 수 있습니다. IAM 계정 보안 주체(나중에 설명)에 더 많은 권한이 필요할 경우 권한을 추가할 수 있습니다. 마찬가지로 권한을 제거 또는 취소해야 할 경우, 환경에 미치는 영향을 최소화하며 해당 작업을 수행할 수 있습니다.

IAM을 사용하여 추가 사용자를 생성하고, 이러한 사용자에게 권한을 지정하여, 최소한의 권한 원칙을 적용하십시오.

전체 제어 권한은 모든 사람이 원합니다. 

문제: 액세스 권한을 세분화하여 제어할 수 있어야 합니다.

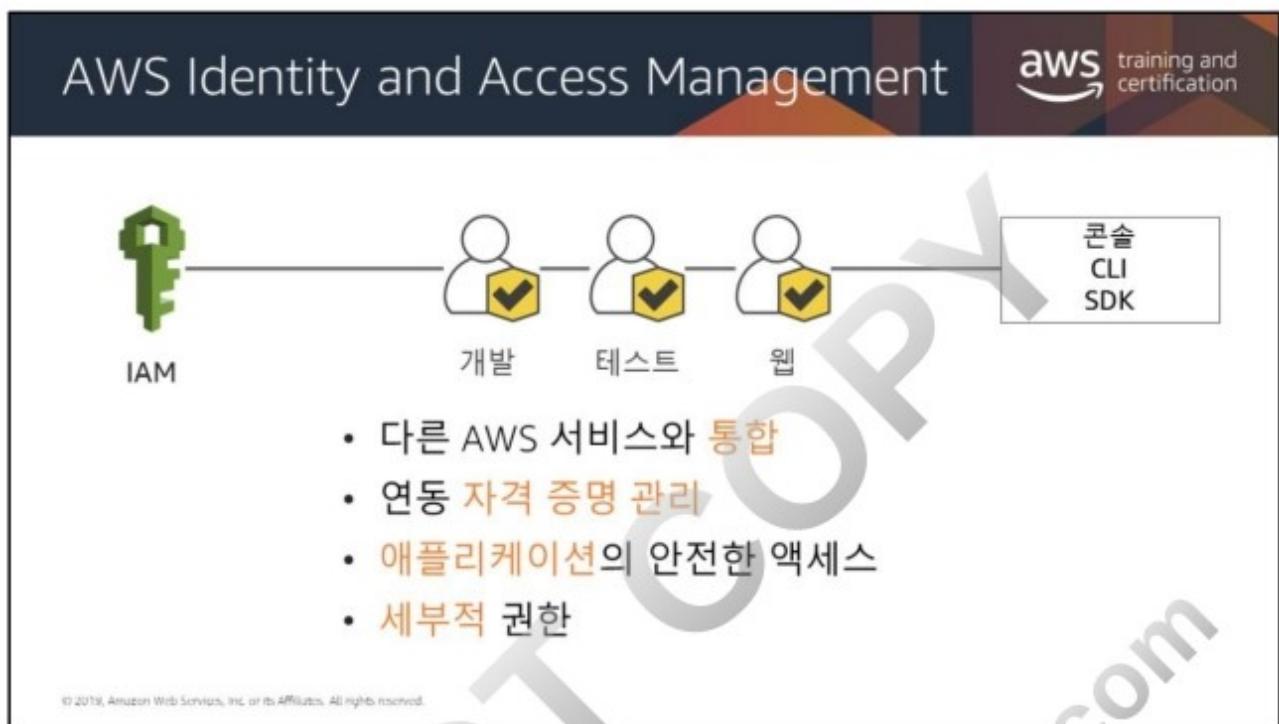


© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

이제 AWS 프로파일에서 가장 중요한 계정을 보호하고 관리자 계정을 생성했으므로 간편한 액세스 및 보안을 위해 추가 계정을 생성합니다. 최소 권한 부여가 표준이 되어야 합니다.

AWS 계정 루트 사용자와 비슷한 권한을 갖는 계정을 생성할 수 있지만 필요한 기능만 제어하는 관리자 계정을 생성하는 것이 더 좋습니다. DBA가 EC2 인스턴스를 프로비저닝할 수 있어야 합니까? 그렇지 않다면 계정을 적절히 프로비저닝합니다.

다양한 계정 유형으로 다양한 계정을 보유하는 것이 유용할 수 있습니다. 권한은 필요에 따라 추가 또는 제거할 수 있습니다.



AWS 자격 증명 관리 시스템에서 사용자를 생성하거나, 사용자에게 개별 보안 자격 증명(예: 액세스 키, 암호, 멀티 팩터 인증(MFA) 디바이스)을 할당하거나, AWS 서비스 및 리소스에 대한 액세스를 제공할 수 있도록 임시 보안 자격 증명을 요청할 수 있습니다. 사용자가 수행할 수 있는 작업을 제어하는 권한을 지정할 수 있습니다.

연합 ID 관리의 경우, 기업 디렉터리에 의해 관리되는 사용자를 위해 만기 구성이 가능한 보안 자격 증명을 요청할 수 있습니다. 이는 직원 및 애플리케이션에게 보안 액세스를 제공합니다. 그러므로 직원 및 애플리케이션을 위해 IAM 사용자 계정을 생성하지 않아도 이들이 AWS 계정 내 리소스에 액세스할 수 있습니다. 이 보안 자격 증명에 권한을 지정하여 사용자가 수행할 수 있는 작업을 제어합니다.

IAM은 AWS 리소스에 대한 액세스를 제어할 수 있게 해주는 서비스입니다. IAM을 사용하면 다양한 수준의 계정 권한 및 권한 부여로 사용자 계정 및 자격 증명을 생성할 수 있습니다.

IAM은 콘솔, AWS CLI, AWS SDK 및 보안 API 엔드포인트에서 액세스할 수 있습니다.



보안 주체란 AWS 리소스에 대해 작업을 수행할 수 있는 엔터티입니다. 시간이 지나면서 사용자 및 서비스가 역할을 맡도록 허용할 수 있습니다. 연동 사용자를 지원하거나 애플리케이션이 AWS 계정에 액세스하도록 허용하는 프로그래밍 방식 액세스를 지원할 수 있습니다. 사용자, 역할, 연동 사용자 및 애플리케이션은 모두 AWS 보안 주체입니다.

보안 주체는 AWS IAM 사용자나 AWS 서비스(예: Amazon EC2, SAML 공급자 또는 자격 증명 공급자(IdP))일 수 있습니다.

AWS 계정에서 IAM 사용자를 생성하는 대신 IdP를 사용할 수 있습니다. IdP를 사용하면 AWS 외부의 사용자 자격 증명을 관리하고(예: Login with Amazon, Google 및 Facebook) 이러한 외부 사용자 자격 증명에 계정의 AWS 리소스를 사용할 권한을 제공할 수 있습니다.

이미지 출처:

Login with Amazon - <https://login.amazon.com/button-guide>

Continue with Facebook - <https://developers.facebook.com/docs/facebook-login/web/login-button>

Sign in with Google - <https://developers.google.com/identity/sign-in/web/build-button>

## IAM 사용자

IAM 사용자는 별도의 AWS 계정이 아니라 계정 내 사용자입니다.

각 사용자는 자체 자격 증명을 갖습니다.

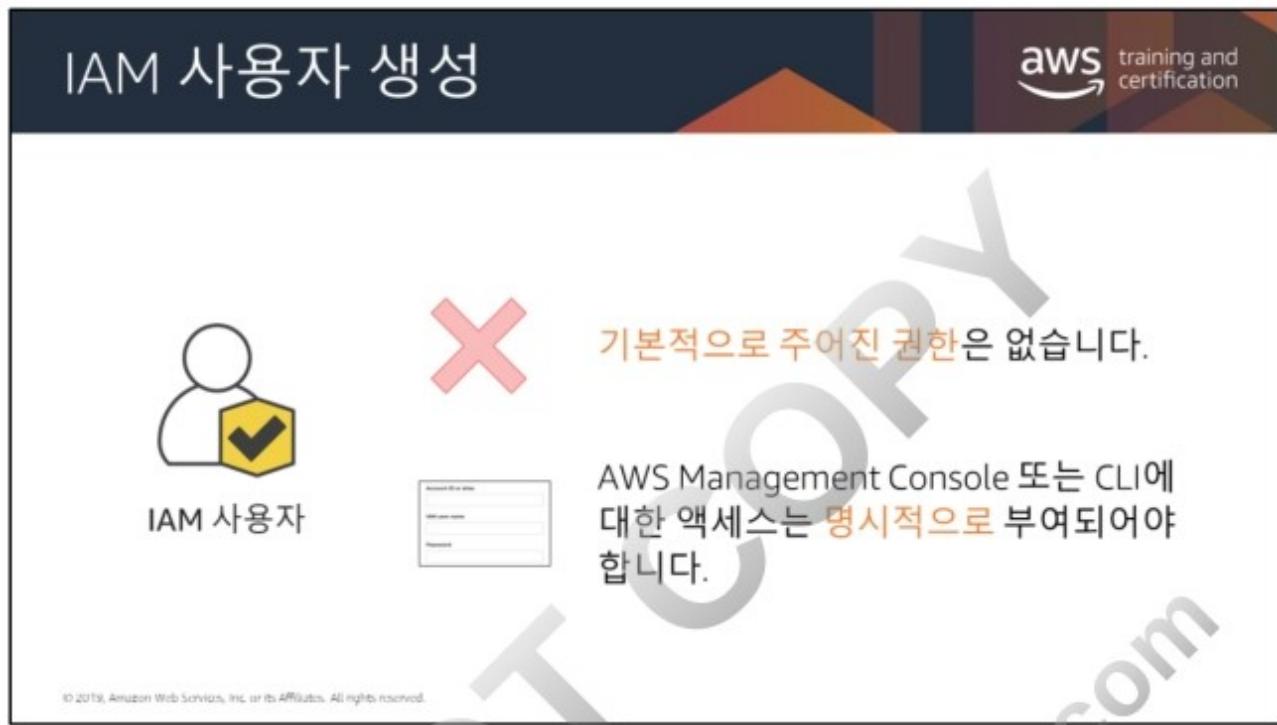
IAM 사용자는 부여된 권한을 기준으로 특정 AWS 작업을 수행할 권리가 있습니다.

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

IAM 사용자는 별도의 AWS 계정이 아니라 계정 안의 보안 주체입니다. 각 IAM 사용자는 콘솔에 액세스하기 위한 고유의 암호를 가집니다. 또한 사용자가 계정 내 리소스에 대한 작업을 프로그래밍 방식으로 요청할 수 있도록 각 사용자마다 개별 액세스 키를 생성할 수도 있습니다. CLI 액세스 권한이 없어도 콘솔에 액세스할 수 있습니다. 반대의 경우도 마찬가지입니다. 콘솔 액세스 권한이 없어도 CLI에 액세스할 수 있습니다.

새롭게 생성된 IAM 사용자에게는 자신을 인증하고 AWS 리소스에 액세스하는 데 사용할 기본 자격 증명이 없습니다. 먼저 IAM 사용자에게 인증을 위한 보안 자격 증명을 지정한 다음, AWS 작업을 수행하거나 AWS 리소스에 액세스할 수 있는 권한을 부여해야 합니다. 사용자를 위해 생성하는 자격 증명은 사용자가 AWS에서 자신을 고유하게 식별하는 데 사용됩니다.

AWS 계정에서 IAM 사용자를 생성하는 대신 IAM IdP를 사용할 수 있습니다. IdP를 사용하면 AWS 외부의 사용자 자격 증명을 관리하고(예: Amazon.com, Google 및 Facebook) 이러한 외부 사용자 자격 증명에 계정의 AWS 리소스를 사용할 권한을 제공할 수 있습니다.



IAM 보안 주체에는 기본 권한이 없습니다. 모든 사용자에게 관리자 권한을 부여하는 것은 권장하지 않습니다. 최소 권한 원칙을 따르는 것이 중요합니다.

## 권한 부여

aws training and certification



정책

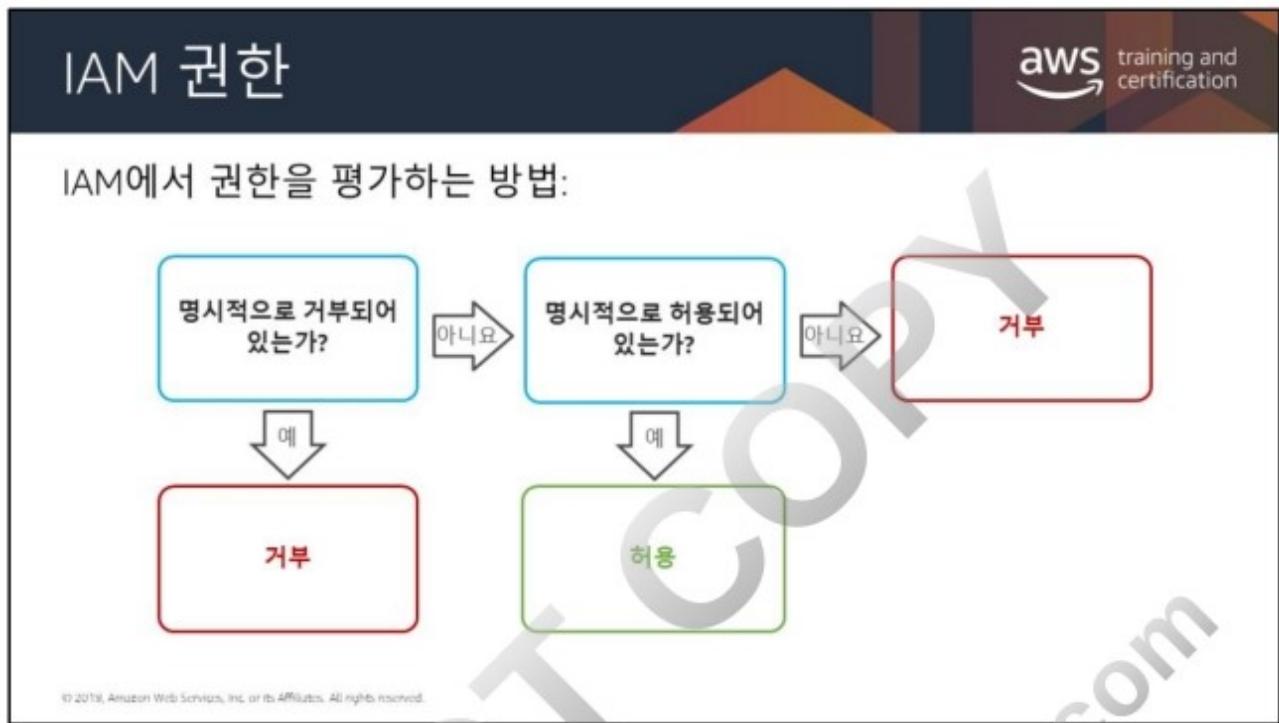
- 하나 이상의 권한에 대한 형식 선언
- 요청 시에 평가됨
- IAM 정책은 AWS 서비스에 대한 액세스만 제어합니다.
- IAM에는 하이퍼바이저에 대한 가시성이 없습니다.

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

정책은 자격 증명 또는 리소스에 연결될 때 해당 권한을 정의하는 AWS의 개체입니다. AWS는 사용자와 같은 보안 주체가 요청할 때 이러한 정책을 평가합니다.

IAM 정책은 AWS 서비스에 대한 액세스만 제어합니다. IAM은 하이퍼바이저 계층 이상으로는 가시성이 없으며 AWS를 벗어날 수 없습니다.

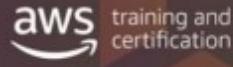
OS 지원을 원할 경우 LDAP, SAML 또는 기타 자격 증명 관리 시스템을 사용합니다.



정책을 통해 IAM 사용자, 그룹, 역할에 부여된 권한을 세부적으로 조정할 수 있습니다. 정책은 JSON 형식으로 저장되므로, 버전 관리 시스템과 함께 사용할 수 있습니다. 각 사용자, 그룹 또는 역할에 대해 최소한의 액세스 권한을 정의하는 것이 좋습니다. 그런 다음 권한 부여 정책을 사용하여 특정 리소스에 대한 액세스를 사용자 정의할 수 있습니다.

권한이 허용되었는지 결정할 때, IAM은 먼저 명시적 거부 정책을 확인합니다. 명시적 거부 정책이 없는 경우, IAM은 다음으로 명시적 허용 정책을 확인합니다. 명시적 거부 정책이나 명시적 허용 정책이 둘 다 없는 경우, 기본 설정인 암시적 거부로 되돌아갑니다.

## 권한 부여



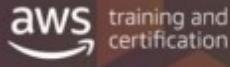
정책

- 리소스 기반 – 연결된 AWS 리소스
- 자격 증명 기반 – 연결된 IAM 보안 주체

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

정책 권한은 요청이 허용되는지 또는 거부되는지 결정합니다. 정책은 JSON 문서로 AWS에 저장되며 자격 증명 기반 정책으로 보안 주체에 연결되거나 리소스 기반 정책으로 리소스에 연결된 됩니다.

## 자격 증명 기반 정책



**연결 대상:**

- 사용자
- 그룹
- 역할

**제어:**

- 수행 작업
- 리소스 대상
- 필요한 조건

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

### 자격 증명 기반 정책

자격 증명 기반 정책은 IAM 사용자, 역할 또는 그룹과 같은 보안 주체(또는 자격 증명)에 연결할 수 있는 권한 정책입니다. 이러한 정책은 자격 증명이 수행할 수 있는 작업, 대상 리소스 및 관련 조건을 제어합니다.

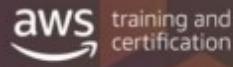
자격 증명 기반 정책을 추가로 분류할 수 있습니다.

**관리형 정책**은 AWS 계정의 여러 사용자, 그룹 및 역할에 연결할 수 있는 독립형 자격 증명 기반 정책입니다. 두 가지 유형의 관리형 정책을 사용할 수 있습니다.

- **AWS 관리형 정책**은 AWS에서 생성하고 관리하는 관리형 정책입니다. 정책 사용이 처음이라면 AWS 관리형 정책을 사용하여 시작하는 것이 좋습니다.
- **고객 관리형 정책**은 고객이 AWS 계정에서 생성하고 관리하는 관리형 정책입니다. 고객 관리형 정책은 AWS 관리형 정책이 아닌 정책을 보다 정밀하게 제어합니다. IAM 정책은 시각적 편집기를 사용하여 또는 JSON 정책 문서를 직접 생성하여 생성 및 편집할 수 있습니다.

**인라인 정책**은 사용자가 생성하고 관리하며, 단일 사용자, 그룹 또는 역할에 직접 포함되는 정책입니다.

## 리소스 기반 정책



**연결 대상:**

- AWS 리소스(예: Amazon S3, Amazon Glacier 및 AWS KMS)

  
리소스 기반  
정책

**제어:**

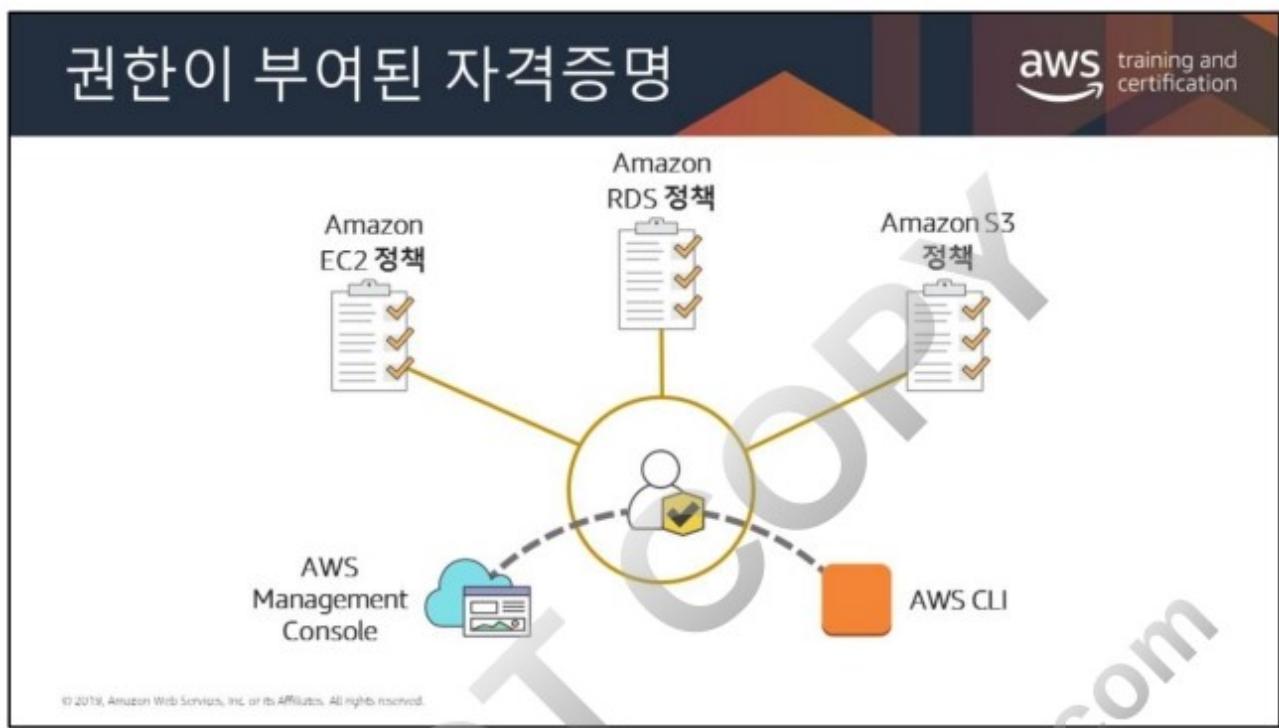
- 특정 보안 주체가 허용한 작업
- 필요한 조건
- 항상 인라인 정책임
- AWS 관리형 리소스 기반 정책이 없음

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

리소스 기반 정책은 Amazon S3 버킷과 같은 리소스에 연결되는 JSON 정책 문서입니다.

이러한 정책은 지정된 보안 주체가 해당 리소스에 대해 수행할 수 있는 작업 및 관련 조건을 제어합니다. 리소스 기반 정책은 인라인 정책이며, 관리형 리소스 기반 정책은 없습니다.

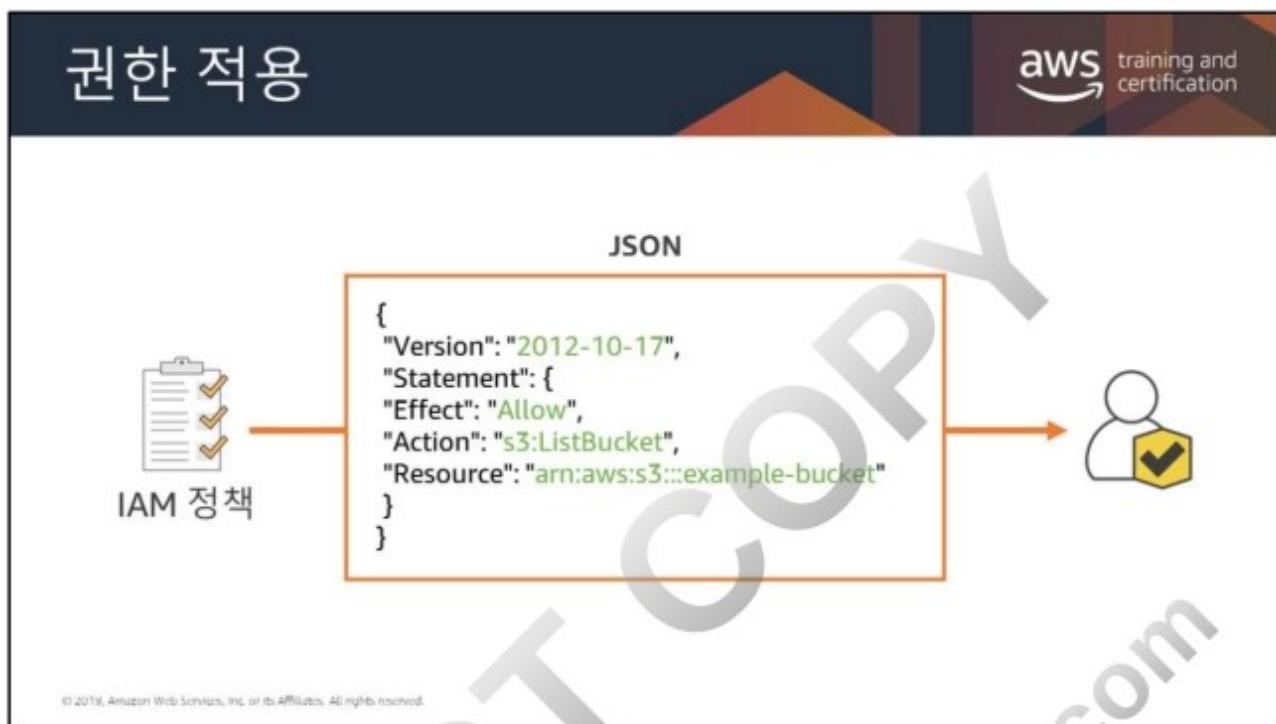
IAM 자격 증명이 기술적으로는 AWS 리소스이지만 리소스 기반 정책을 IAM 자격 증명에 연결할 수는 없습니다.



IAM 사용자는 권한이 연결된 자격 증명(보안 주체)일 뿐입니다.

AWS에 요청을 하기 위해서는 자격 증명이 반드시 필요한 애플리케이션에 사용할 IAM 사용자를 생성할 수 있습니다.

애플리케이션은 계정 내에서 고유한 자격 증명과 AWS 서비스에 액세스할 수 있는 고유한 권한 세트를 가질 수 있습니다. 이는 현대 운영 체제에서 프로세스가 고유한 자격 증명과 권한을 갖는 것과 비슷합니다. 애플리케이션 또는 심지어 EC2 인스턴스가 s3 버킷과 같은 리소스에 액세스할 권리가 있는 경우에는 코드에 자격 증명을 포함할 필요가 없습니다.



IAM 정책은 하나 이상의 권한으로 구성된 공식 문입니다.

- 어떤 IAM 엔터티에도 정책을 연결할 수 있습니다.
- 정책은 엔터티가 수행할 수 있거나 수행할 수 없는 작업에 대한 권한을 부여합니다.
- 단일 정책이 여러 개의 엔터티에 연결될 수 있습니다.
- 단일 엔터티에 여러 개의 정책이 연결될 수 있습니다.

## IAM 정책 예

The screenshot shows a sample IAM policy document with annotations:

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": ["dynamodb:*", "s3:*"],  
            "Resource": ["arn:aws:dynamodb:region:account-number-without-hyphens:table/table-name",  
                        "arn:aws:s3:::bucket-name",  
                        "arn:aws:s3:::bucket-name/*"]  
        },  
        {  
            "Effect": "Deny",  
            "Action": ["dynamodb:*", "s3:*"],  
            "NotResource": ["arn:aws:dynamodb:region:account-number-without-hyphens:table/table-name",  
                           "arn:aws:s3:::bucket-name",  
                           "arn:aws:s3:::bucket-name/*"]  
        }  
    ]  
}
```

Annotations in Korean:

- 사용자에게 액세스 권한 부여(특정 DynamoDB 테이블과...)
- ... 특정 Amazon S3 버킷 및 해당 콘텐츠
- 명시적 거부 문은 보안 주체가 지정된 테이블 및 버킷이 아닌 AWS 작업 또는 리소스를 사용할 수 없도록 합니다.
- 명시적 거부 문은 허용문보다 우선 적용됩니다.

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

**중요:** 버전 구문을 편집하지 마십시오. 이것은 IAM 정책을 처리하는 엔진을 참조합니다.

작업 섹션의 별표는 와일드카드입니다. 이 경우, DynamoDB 및 Amazon S3 서비스에 대한 모든 작업이 허용됩니다. 와일드 카드를 포함하는 이름을 사용할 수도 있습니다. 예를 들어 s3>List\*는 ListAllMyBuckets, ListBucket, ListBucketByTags, ListBucketMultipartUploads, ListBucketVersions, ListMultipartUploadParts와 일치합니다.

거부 섹션에서 NotResource는 이전에 본 적이 없다면 혼동을 줄 수 있습니다. NotResource는 지정된 목록의 리소스를 제외한 모든 리소스와 명시적으로 일치하는 고급 정책 요소입니다.

여기에서 NotResource는 여기에 나열된 것 이외의 모든 작업을 명시적으로 거부하는 것을 의미합니다. 정책의 전반부를 변경하는 경우 거부 문도 적절히 변경해야 합니다.

묵시적 거부가 있는데 이렇게 하는 이유는 무엇일까요? 누군가에게는 의도치 않은 권한 부여를 방지하기 위해 액세스를 잠그는 것이 중요합니다. (악의적 사용자를 상정하기 쉽지만 선의를 가졌지만 제대로 이해하지 못한 사용자도 있을 수 있습니다.) 이로 인해 복잡성이 추가된다는 점을 주의하십시오. 이것이 반드시 나쁜 것은 아니지만 복잡성은 추가 작업을 요구하는 것으로 보일 수 있습니다.

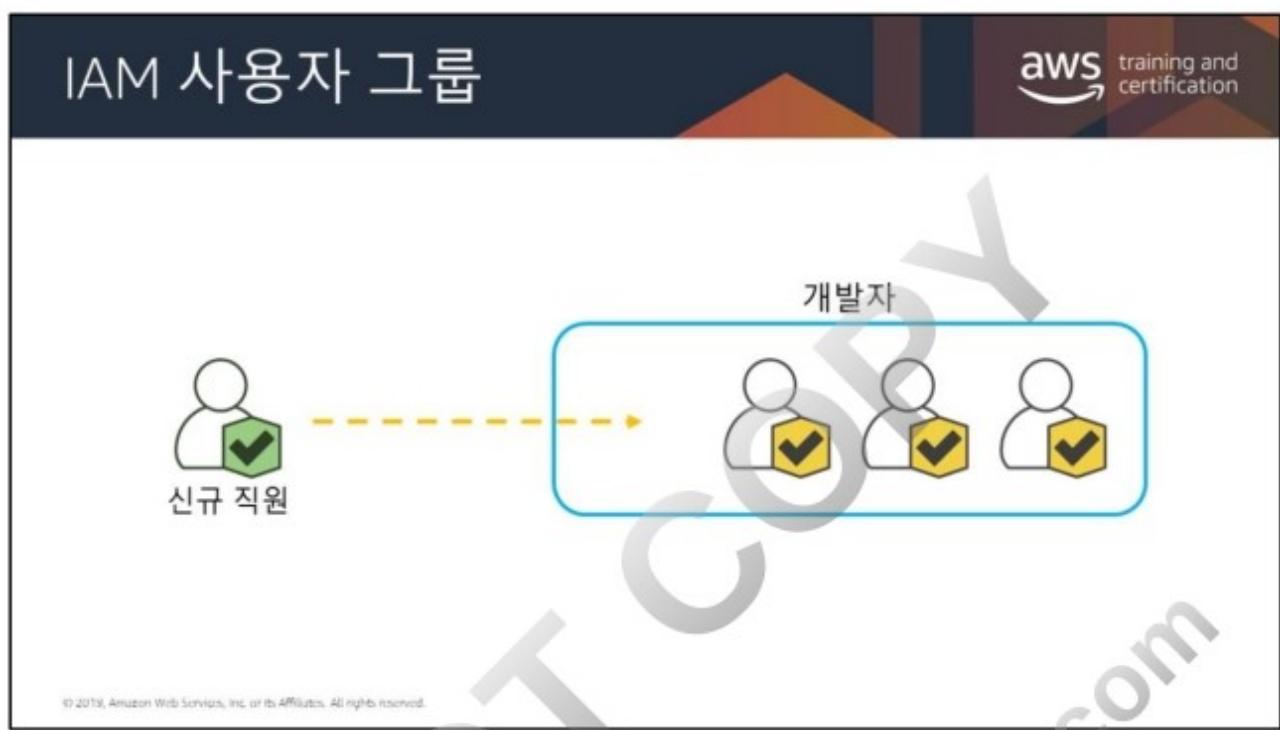
NotResource를 사용하면 일치하는 리소스를 길게 나열하는 대신 일치하면 안 되는 몇 개의 리소스만 나열하면 되므로 정책이 짧아집니다. NotResource를 사용할 때는 이 요소에 지정된 리소스들이 제한되는 유일한 리소스라는 점을 유의해야 합니다.

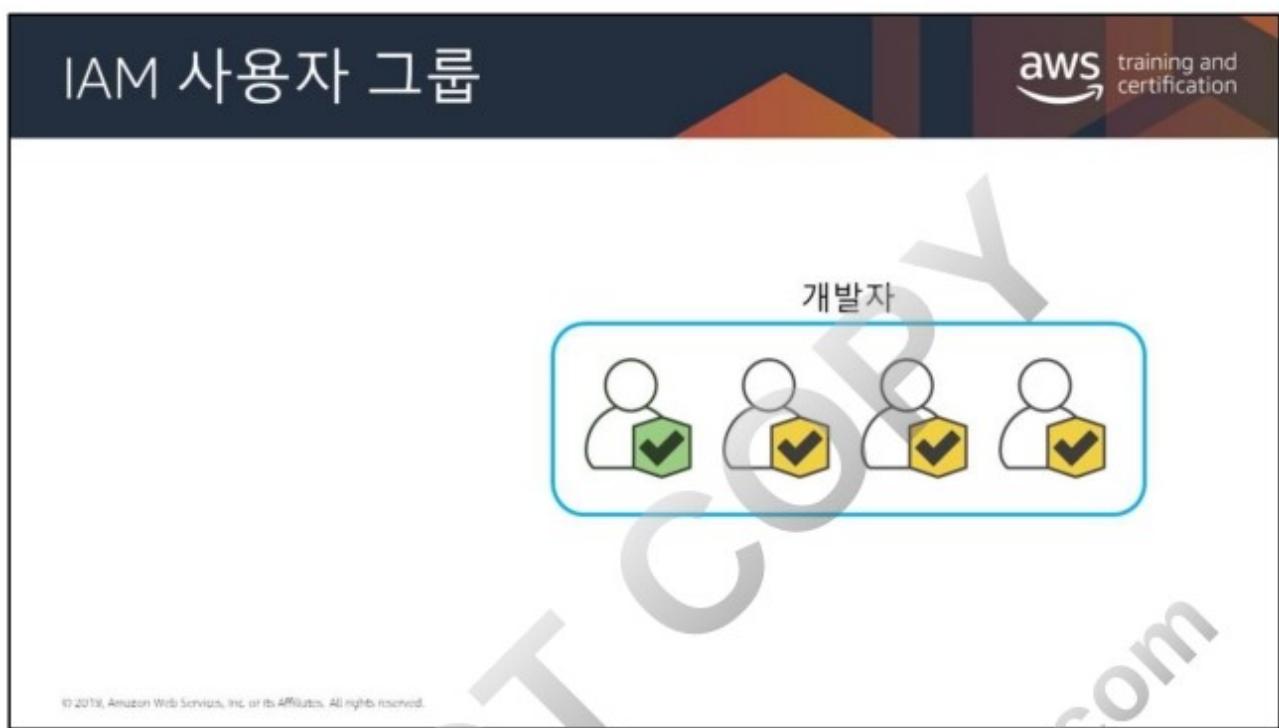


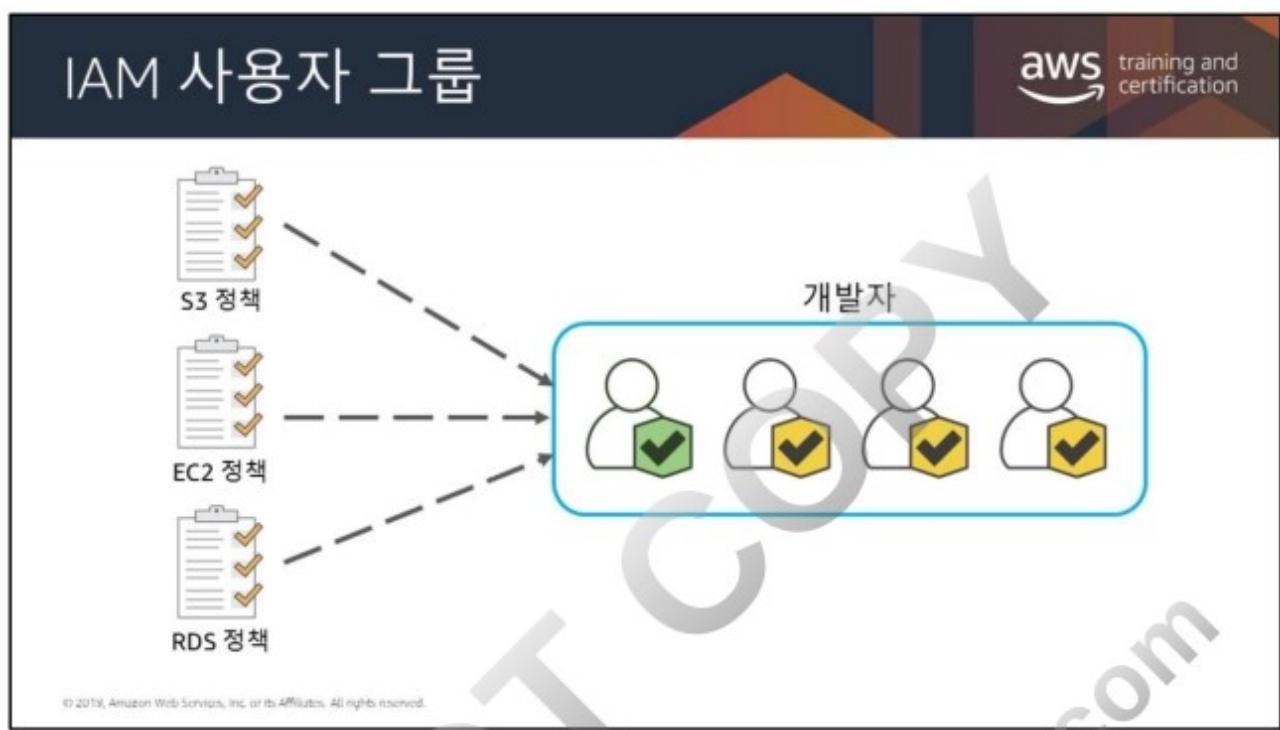
## 사용자 구성

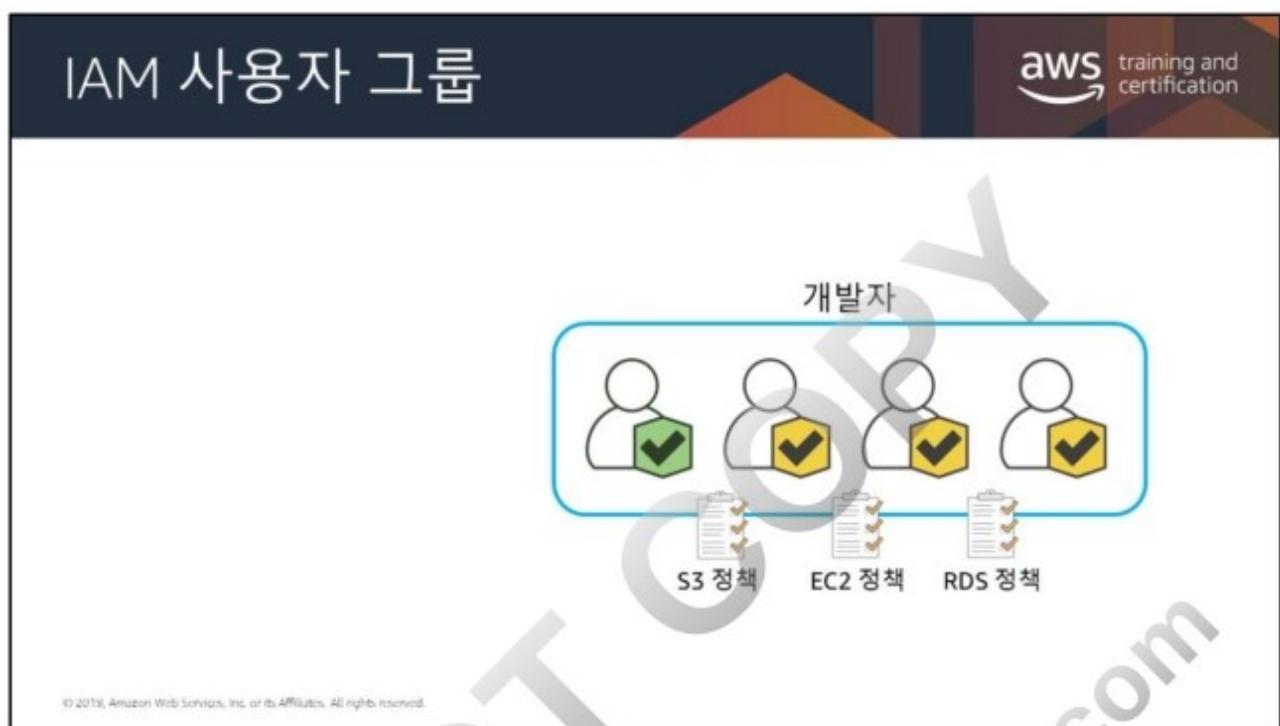
aws training and certification

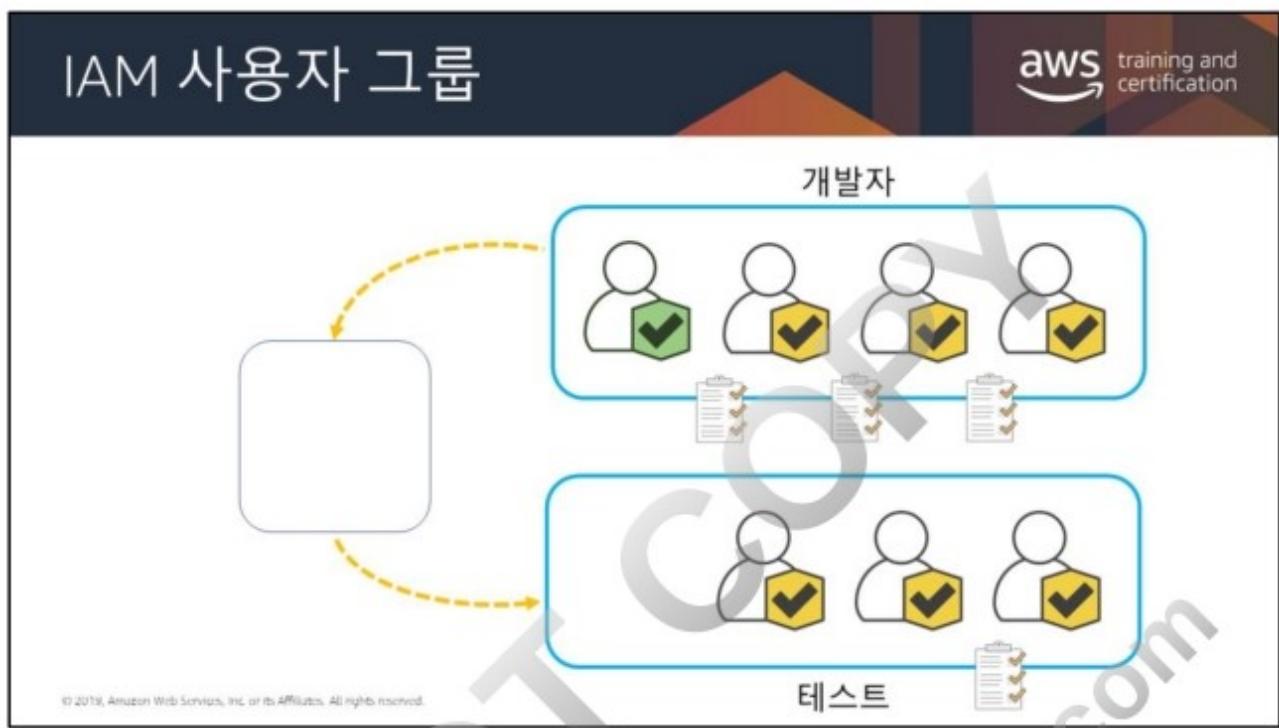
DO NOT COPY  
zlagusdbs@gmail.com



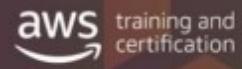








사용자가 하루만 테스트해야 하는  
경우에는 어떻게 합니까?



다른 그룹 간에 사용자를 계속 푸시/풀링하지 않으려는  
경우 어떻게 합니까?

다른 대상에게 영구 자격 증명을 부여하고 싶지 않으면  
어떻게 합니까?

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.