

AWS 백서

# Amazon EC2 스팟 인스턴스 개요



## Amazon EC2 스팟 인스턴스 개요: AWS 백서

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

Amazon의 상표 및 브랜드 디자인은 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계 여부에 관계없이 해당 소유자의 자산입니다.

# Table of Contents

요약 및 소개 .....	1
요약 .....	1
소개 .....	1
스팟 인스턴스를 사용하는 시기 .....	2
스팟 인스턴스를 시작하는 방법 .....	3
스팟 인스턴스 작업 방식 .....	4
스팟 인스턴스 중단 관리 .....	5
스팟 인스턴스 제한 .....	6
스팟 인스턴스 모범 사례 .....	7
다른 AWS 서비스와 스팟 통합 .....	8
Amazon EMR 통합 .....	8
EC2 Auto Scaling 통합 .....	8
Amazon EKS 통합 .....	8
Amazon ECS 통합 .....	8
AWS Fargate Spot을 지원하는 Amazon ECS 통합 .....	8
Amazon Batch 통합 .....	9
Amazon SageMaker 통합 .....	9
Amazon Gamelift 통합 .....	9
AWS Elastic Beanstalk 통합 .....	9
결론 .....	10
리소스 .....	11
문서 기록 및 기여자 .....	12
문서 기록 .....	12
기여자 .....	12

# Amazon EC2 스팟 인스턴스 개요

게시 날짜: 2021년 3월 5일([문서 기록 및 기여자](#))

## 요약

본 백서에서는 투자로 얻을 수 있는 가치를 극대화하고 예측 정확성과 비용 예측 가능성을 향상하고, 소유권과 비용 투명성이 존중되는 문화를 조성하며 최적화 상태를 지속적으로 측정할 수 있도록 지원합니다.

본 백서에서는 Amazon EC2 스팟 인스턴스의 개요와 더불어, 이를 효과적으로 사용하는 모범 사례를 제공합니다.

## 소개

[온디맨드](#), [예약 인스턴스](#) 및 [Savings Plans](#)에 이어 네 번째 [Amazon Elastic Compute Cloud](#)(Amazon EC2) 요금 모델은 [스팟 인스턴스](#)입니다.

스팟 인스턴스를 사용하면 온디맨드 요금과 비교하여 최대 90% 할인된 가격으로 예비 Amazon EC2 컴퓨팅 용량을 사용할 수 있습니다. 즉, 애플리케이션 실행 비용을 대폭 절감하거나 동일한 예산으로 애플리케이션의 컴퓨팅 용량과 처리량을 늘릴 수 있습니다. 온디맨드 인스턴스와 스팟 인스턴스의 유일한 차이점은 EC2에 해당 용량이 필요하면 2분 동안 알림과 함께 스팟 인스턴스가 EC2에 의해 중단될 수 있다는 것입니다.

예약 인스턴스 또는 Savings Plans와 달리 스팟 인스턴스는 온디맨드 가격 대비 비용 절감을 위해 약정이 필요하지 않습니다. 그러나 스팟 인스턴스는 실행 중인 용량 풀(인스턴스 유형과 가용성 영역의 조합)에 사용 가능한 용량이 없으면 EC2에서 종료할 수 있으므로 유연한 워크로드에 가장 적합합니다.

## 스팟 인스턴스를 사용하는 시기

내결함성이 있고 유연한 여러 애플리케이션에 스팟 인스턴스를 사용할 수 있습니다. 예를 들면 상태 비저장 웹 서버, API 엔드포인트, 빅 데이터 및 분석 애플리케이션, 컨테이너화된 워크로드, CI/CD 고성능 및 고처리량 컴퓨팅(HPC/HTC), 렌더링 워크로드 및 기타 유연한 워크로드가 있습니다.

스팟 인스턴스는 유연성이 없거나 상태 저장이거나 내결함성이 없거나 인스턴스 노드 간에 밀접하게 연결된 워크로드에는 적합하지 않습니다. 또한 목표 용량을 완전히 사용할 수 없는 경우가 종종 발생하는 워크로드에는 스팟 인스턴스가 권장되지 않습니다. 이러한 워크로드에는 스팟 인스턴스를 사용하거나 종단을 처리하기 위해 온디맨드 인스턴스로 장애 조치를 시도하지 않도록 강력히 경고합니다.

## 스팟 인스턴스를 시작하는 방법

스팟 인스턴스를 시작할 때 가장 권장되는 서비스는 [Amazon EC2 Auto Scaling](#)으로, 이를 사용하여 원하는 용량을 시작 및 유지하고 중단되거나 수동으로 종료된 용량을 교체할 리소스를 자동으로 요청할 수 있습니다. Auto Scaling 그룹을 구성할 때 애플리케이션 요구 사항에 따라 인스턴스 유형과 원하는 용량만 지정하면 됩니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [Auto Scaling 그룹](#)을 참조하세요.

유연성이 더 필요하거나, 자체 인스턴스 시작 워크플로를 구축했거나, 인스턴스 시작 또는 확장 메커니즘의 개별 측면을 제어하려는 경우, EC2 Auto Scaling의 대안으로 인스턴트 모드에서 [EC2 플릿](#)의 사용을 평가하는 것이 좋습니다. 이 동기식 API를 사용하면 인스턴스 유형 및 시작 요구 사항 목록을 지정할 수 있으며, 스팟 인스턴스 또는 온디맨드 인스턴스를 시작하기 위한 EC2 [RunInstances](#) API 호출보다 더 유연한 기능을 제공합니다.

AWS 서비스를 사용하여 클라우드 워크로드를 실행하는 경우, 스팟 인스턴스를 시작하는 데도 사용할 수 있습니다. 예를 들면 [Amazon EMR](#), [Amazon EKS](#), [Amazon ECS](#), [AWS Batch](#), [AWS Elastic Beanstalk](#)가 있습니다. AWS 클라우드와 통합되는 서드 파티 도구를 사용하여 스팟 인스턴스를 시작할 수도 있습니다.

코드형 인프라 도구([AWS CloudFormation](#), [AWS CDK](#)) 또는 AWS API, CLI 또는 SDK로 사용하여 스팟 인스턴스 시작을 자동화할 수 있습니다. [Spot Blueprints](#)는 스팟 모범 사례를 준수하는 AWS Cloudformation 및 Hashicorp Terraform용 코드형 인프라 템플릿을 생성할 수 있는 안내 마법사를 제공합니다.

## 스팟 인스턴스 작업 방식

스팟 인스턴스는 실행 중일 때 다른 EC2 인스턴스와 동일하게 작동합니다. 그러나 EC2에 해당 용량이 필요하면 Amazon EC2에서 중단시킬 수 있습니다.

EC2에서 스팟 인스턴스를 중단하면 사용자가 선택한 중단 동작에 따라 인스턴스를 종료, 중지 또는 최대 절전 모드로 전환합니다.

EC2에서 전체 실행 시간 전에 처음 한 시간 동안 스팟 인스턴스를 중단하는 경우, 사용한 부분 시간에 대해서는 요금이 부과되지 않습니다. 그러나 스팟 인스턴스를 중지하거나 종료하는 경우, 사용한 부분 시간에 대한 비용을 지불하게 됩니다 (온디맨드 또는 예약 인스턴스의 경우와 동일). 다른 운영 체제에서 실행되는 중단된 스팟 인스턴스에 대한 요금 청구 방법에 대한 자세한 내용은 EC2 사용 설명서의 [중단된 스팟 인스턴스에 대한 요금 청구](#)를 참조하세요.

가용 영역별로 각 인스턴스 유형에 대한 스팟 요금은 EC2 예비 용량의 장기 공급 및 수요 추세에 따라 결정됩니다. 유효한 스팟 요금이 가장 가까운 초 단위로 청구됩니다.

원할 경우, 스팟 인스턴스의 최대 요금을 지정할 수 있습니다. 최대 요금을 지정하지 않는 경우, 기본 최대 요금은 온디맨드 요금입니다. 스팟 인스턴스가 실행 중일 때 유효한 스팟 요금보다 더 많이 지불하게 되는 일은 없습니다. 최대 요금을 지정하지 않고 최대 요금의 기본값을 온디맨드 요금으로 설정하는 것이 좋습니다. 최대 요금이 높다고 해서 스팟 인스턴스를 시작할 가능성이 높아지지 않으며 스팟 인스턴스가 중단될 가능성도 줄어들지 않습니다 (EC2에서 용량이 다시 필요하면 스팟 인스턴스를 중단할 수 있기 때문).

가용 영역의 인스턴스 유형에 대한 스팟 요금은 언제든지 변경될 수 있지만 일반적으로 자주 변경되지는 않습니다. AWS는 스팟 인스턴스의 현재 스팟 요금과 이전 요금을 [DescribeSpotPriceHistory](#) API와 API의 데이터를 반영하는 AWS Management Console을 통해 게시합니다. 이를 통해 스팟 요금의 변동 수준과 시기를 추정할 수 있습니다.

## 스팟 인스턴스 중단 관리

스팟 인스턴스 중단을 원활하게 처리하고 성능이나 가용성에 미치는 영향을 최소화하는 가장 좋은 방법은 내결함성을 갖춘 애플리케이션을 설계하는 것입니다. EC2 인스턴스 재분배 권고 및 스팟 인스턴스 중단 공지를 활용하여 이를 달성할 수 있습니다.

EC2 인스턴스 재분배 권고는 스팟 인스턴스의 중단 위험이 높아지면 알림을 보내는 신호입니다. 이 신호를 통해 스팟 인스턴스가 중단되기 2분 전 공지에 앞서 스팟 인스턴스를 사전에 관리할 수 있습니다. 중단 위험이 높지 않은 신규 또는 기존 스팟 인스턴스에 대한 워크로드를 재분배하도록 결정할 수 있습니다. EC2 Auto Scaling 그룹 및 의 용량 및 재분배 기능을 통해 이 새로운 신호를 손쉽게 사용할 수 있습니다. 자세한 내용은 [Amazon EC2 Auto Scaling 용량 재분배](#)를 참조하십시오.

스팟 인스턴스 중단 알림은 Amazon EC2에서 스팟 인스턴스를 중단하기 2분 전에 발생하는 경고입니다. 워크로드가 "시간 유연성"인 경우, 스팟 인스턴스가 중단될 때 종료되지 않고 중지 또는 최대 절전 모드로 전환되도록 구성할 수 있습니다. Amazon EC2는 중단 시 자동으로 스팟 인스턴스를 중지하거나 최대 절전 모드로 전환하고, 사용 가능한 용량이 확보되면 인스턴스를 자동으로 재개합니다.

EC2 인스턴스 재조정 권고 및/또는 스팟 인스턴스 중단 알림을 사용하여 내결함성을 염두에 두고 워크로드를 설계하여 알림을 캡처하고 작업 상태를 스토리지(예: Amazon S3, Amazon EFS 또는 Amazon FSx)에 저장하고 인스턴스에서 로그 파일을 유지하거나(또는 내결함성 접근 방식을 위한 지속적인 스트리밍) 로드 밸런서에서 연결을 드레이닝하는 등의 작업을 수행할 수 있습니다.

일부 AWS 및 서드 파티 서비스는 이미 스팟 중단을 처리하여 애플리케이션에 미치는 영향을 줄입니다. 예를 들어, [스팟 인스턴스와 함께 관리형 노드 그룹](#)을 실행하는 Amazon EKS는 기존 노드에 대한 재분배 권고 또는 중단 공지가 전달되면 자동으로 대체 Kubernetes 노드가 시작됩니다.



# 스팟 인스턴스 제한

리전별 AWS 계정당 실행 및 요청된 스팟 인스턴스의 수에는 제한이 있습니다. 스팟 인스턴스 제한은 실행 중인 스팟 인스턴스가 사용 중이거나 미결제 스팟 인스턴스 요청의 이행 보류 중 사용할 vCPU(가상 중앙 처리 장치) 수를 기준으로 관리됩니다. 스팟 인스턴스를 종료하지만 스팟 인스턴스 요청을 취소하지 않으면 Amazon EC2에서 스팟 인스턴스 종료를 감지하여 요청을 닫을 때까지 스팟 인스턴스 vCPU 제한에 대한 요청이 계산됩니다.

스팟 인스턴스에는 다음과 같은 6가지 제한이 있습니다.

- 모든 표준(A, C, D, H, I, M, R, T, Z) 스팟 인스턴스 요청
- 모든 F 스팟 인스턴스 요청
- 모든 G 스팟 인스턴스 요청
- 모든 Inf 스팟 인스턴스 요청
- 모든 P 스팟 인스턴스 요청
- 모든 X 스팟 인스턴스 요청

각 제한은 하나 이상의 인스턴스 패밀리에 vCPU 제한을 지정합니다. 다른 인스턴스 패밀리, 세대 및 크기에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

vCPU 제한을 사용하면 변화하는 애플리케이션 요구 사항을 충족하는 모든 인스턴스 유형 조합을 시작하는 데 필요한 vCPU 수에 대한 제한을 사용할 수 있습니다. 예를 들어 모든 표준 스팟 인스턴스 요청 제한이 256개의 vCPU라고 가정하면 m5.2xlarge 스팟 인스턴스 32개(vCPU 32 x 8) 또는 c5.4xlarge 스팟 인스턴스 16개(vCPU 16 x 16)를 요청하거나 총 256개의 vCPU에 해당하는 표준 스팟 인스턴스 유형 및 크기의 조합을 요청할 수 있습니다.

자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [스팟 인스턴스 제한 및 사용량 모니터링](#) 및 [스팟 인스턴스 제한 증가 요청](#)을 참조하십시오.

## 스팟 인스턴스 모범 사례

애플리케이션의 다음 모범 사례 적용 방법은 인스턴스 유형 요구 사항, 예산 요구 사항 및 애플리케이션 설계에 따라 결정됩니다.

- 인스턴스 유형에 대한 유연성을 갖습니다. 스팟 인스턴스 풀은 인스턴스 유형(예: m5.large)과 가용 영역(예: us-east-1a)이 동일한 미사용 EC2 인스턴스의 집합입니다. 요청하는 인스턴스 유형과 워크로드를 배포할 수 있는 가용 영역에 대한 유연성이 있어야 합니다. 그러면 스팟에서 필요한 컴퓨팅 용량을 찾고 할당할 가능성이 높아집니다. 예를 들어 c4, m5 및 m4 제품군의 large를 사용하려는 경우, c5.large만 요청하지 마십시오.
- 용량 최적화된 할당 전략을 사용합니다. EC2 Auto Scaling 그룹의 할당 전략은 예비 용량이 있는 스팟 인스턴스 풀을 수동으로 찾을 필요 없이 목표 용량을 프로비저닝하는 데 도움이 됩니다. 용량 최적화된 이 전략은 가장 사용 가능한 스팟 인스턴스 풀에서 인스턴스를 자동으로 프로비저닝하므로 이 전략을 사용하는 것이 좋습니다. 스팟 인스턴스 용량은 최적의 용량을 가진 풀에서 소싱되므로 스팟 인스턴스가 중단될 가능성이 줄어듭니다. 할당 전략에 대한 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [스팟 인스턴스](#)를 참조하십시오.
- 사전 예방적 용량 리밸런싱을 사용합니다. 용량 리밸런싱을 사용하면 실행 중인 스팟 인스턴스에 중단 2분 전 공지가 수신되기 전에 미리 새 스팟 인스턴스로 Auto Scaling 그룹을 보강할 수 있으므로 워크로드 가용성을 유지하는 데 도움이 됩니다. 용량 리밸런싱이 활성화된 경우, Auto Scaling은 리밸런싱 권고를 수신하는 스팟 인스턴스의 사전 교체를 시도하여 중단 위험이 높지 않은 새 스팟 인스턴스로 워크로드를 리밸런싱할 기회를 제공합니다.
- 통합된 AWS 서비스를 사용하여 스팟 인스턴스를 관리합니다. 다른 AWS 서비스가 스팟과 통합되어 개별 인스턴스 또는 플릿을 관리할 필요 없이 전체 컴퓨팅 비용을 절감할 수 있습니다. 적용 가능한 워크로드에는 Amazon EMR, Amazon ECS, AWS Batch, Amazon EKS, SageMaker, AWS Elastic Beanstalk, Amazon GameLift 등의 솔루션을 고려하는 것이 좋습니다. 이러한 서비스의 스팟 모범 사례에 대해 자세히 알아보려면 [Amazon EC2 스팟 인스턴스 워크숍 웹 사이트](#)를 참조하십시오.
- 스팟 인스턴스에 적합한 최신 시작 도구를 선택합니다. AWS 통합 서비스 중 하나가 워크로드에 적합하지 않고 스팟 인스턴스 시작을 제어할 수 있는 애플리케이션을 구축해야 하는 경우, 올바른 도구를 사용합니다. ELB 지원 애플리케이션, 컨테이너식 워크로드, 대기열 처리 작업 등 다양한 워크로드에 대해 보다 포괄적인 기능 집합을 제공하기 때문에 대부분의 워크로드에서는 EC2 Auto Scaling을 사용해야 합니다. 개별 요청에 대한 더 많은 제어가 필요하고 “시작 전용” 도구를 찾고 있는 경우, 인스턴스 유형 다각화 및 할당 전략과 같은 더 광범위한 기능을 갖춘 RunInstances를 바로 대체할 수 있는 인스턴트 모드의 EC2 플릿을 사용합니다.

## 다른 AWS 서비스와 스팟 통합

Amazon EC2 스팟 인스턴스는 여러 AWS 서비스와 통합됩니다.

### Amazon EMR 통합

스팟 인스턴스에서 Amazon EMR 클러스터를 실행하여 분석 워크로드에 대한 방대한 양의 데이터 처리 비용을 크게 절감할 수 있습니다. [EMR 인스턴스 플릿](#)으로 스팟 인스턴스를 온디맨드 및 예약 인스턴스와 쉽게 혼합하여 EMR 클러스터를 실행할 수 있습니다. [EMR 할당 전략](#)을 사용하여 가용성이 가장 높은 용량 풀에서 스팟 인스턴스를 시작할 수 있습니다.

### EC2 Auto Scaling 통합

[Amazon EC2 Auto Scaling](#) 그룹을 사용하여 스팟 인스턴스를 시작 및 관리하고, 애플리케이션 가용성을 유지하고, 인스턴스 유형 및 구매 옵션(온디맨드/스팟) 선택을 다양화하고 동적, 예약 및 예측 조정 정책을 사용하여 Amazon EC2 용량을 확장할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [내결함성 및 유연한 애플리케이션에 대한 스팟 인스턴스 요청](#)을 참조하십시오.

### Amazon EKS 통합

EKS 관리형 노드 그룹의 스팟 인스턴스를 시작하여 Amazon EKS로 Kubernetes 기반 워크로드의 비용을 최적화할 수 있습니다. EKS 관리형 노드 그룹은 곧 중단될 스팟 인스턴스를 새로 시작된 인스턴스로 대체하여 전체 스팟 인스턴스 수명 주기를 관리해서 스팟 인스턴스 중단 시(EC2에 해당 용량이 필요한 경우) 애플리케이션 성능 또는 가용성에 영향을 미칠 가능성을 줄입니다. 자세한 내용은 Amazon EKS 사용 설명서의 [관리형 노드 그룹](#)을 참조하십시오.

### Amazon ECS 통합

스팟 인스턴스에서 Amazon ECS 클러스터를 실행하여 컨테이너식 애플리케이션을 실행하는 데 드는 운영 비용을 절감할 수 있습니다. Amazon ECS는 곧 중단될 스팟 인스턴스의 자동 트레이닝을 지원합니다. 자세한 내용은 Amazon Elastic Container Service 개발자 안내서의 [스팟 인스턴스 사용](#)을 참조하십시오.

### AWS Fargate Spot을 지원하는 Amazon ECS 통합

컨테이너식 작업이 중단 가능하고 유연하다면 AWS Fargate Spot 용량 공급자를 통해 ECS 작업을 실행하도록 선택할 수 있고, 작업이 서버리스 컨테이너 플랫폼인 AWS Fargate에서 실행되며, Fargate

Spot을 통해 비용을 절감할 수 있습니다. 자세한 내용은 Amazon Elastic Container Service 개발자 안내서의 [AWS Fargate 용량 공급자](#)를 참조하십시오.

## Amazon Batch 통합

[AWS Batch](#)는 AWS에서 고객 배치 컴퓨팅 워크로드를 계획, 예약 및 실행합니다. AWS Batch는 사용자를 대신하여 스팟 인스턴스를 동적으로 요청하므로 배치 작업 실행 비용이 절감됩니다.

## Amazon SageMaker 통합

Amazon SageMaker를 사용하여 관리형 스팟 인스턴스로 기계 학습 모델을 쉽게 교육할 수 있습니다. 관리형 스팟 교육은 온디맨드 인스턴스에 비해 모델 교육 비용을 최대 90% 까지 최적화할 수 있습니다. SageMaker는 사용자를 대신하여 스팟 중단을 관리합니다. 자세한 내용은 Amazon SageMaker 개발자 안내서의 [Amazon SageMaker의 관리형 스팟 교육](#)을 참조하십시오.

## Amazon GameLift 통합

Amazon GameLift는 멀티플레이어 게임용 클라우드 서버를 배포 및 운영하고 확장하는 게임 서버 호스팅 솔루션입니다. Amazon GameLift에서 스팟 인스턴스를 지원하므로 호스팅 비용을 대폭 절감할 수 있습니다. 호스팅 리소스 풀릿을 생성할 때 온디맨드 인스턴스 또는 스팟 인스턴스 중에서 선택할 수 있습니다. 2분 동안 알림을 받으면 스팟 인스턴스가 중단될 수 있으나 Amazon GameLift의 FleetIQ가 중단 가능성을 최소화합니다. 자세한 내용은 Amazon GameLift 개발자 안내서의 [GameLift와 함께 스팟 인스턴스 사용](#)을 참조하십시오.

## AWS Elastic Beanstalk 통합

AWS Elastic Beanstalk는 Java, .NET, PHP, Node.js, Python, Ruby, Go 및 Docker를 사용하여 Apache, Nginx, Passenger, IIS와 같은 친숙한 서버에서 개발된 웹 애플리케이션 및 서비스를 간편하게 배포하고 조정할 수 있는 서비스입니다. 코드를 업로드하기만 하면 Elastic Beanstalk가 용량 프로비저닝, 로드 밸런싱, Auto Scaling부터 애플리케이션 상태 모니터링에 이르기까지 배포를 자동으로 처리합니다. Elastic Beanstalk 환경에서 스팟 인스턴스를 사용하여 웹 애플리케이션의 기본 인프라 비용을 최적화할 수 있습니다. Elastic Beanstalk와 함께 스팟 인스턴스를 사용하는 방법에 대한 자세한 내용은 AWS Elastic Beanstalk 개발자 안내서의 [스팟 인스턴스 지원](#)을 참조하십시오.

## 결론

유연한 컴퓨팅 요구 사항이 있거나 예산을 늘리지 않고 용량을 늘릴지 여부와 관계없이 스팟 인스턴스는 AWS 비용을 최적화하거나 규모를 엄두에 두고 구축할 수 있습니다. 워크로드를 적절하게 설계하여 다양한 요구 사항에 맞는 스팟 인스턴스를 활용할 수 있습니다. 자세한 내용은 [Amazon EC2 스팟 인스턴스](#)를 참조하세요.

## 리소스

- [AWS 아키텍처 센터](#)
- [AWS 백서](#)
- [AWS Architecture Monthly](#)
- [AWS 아키텍처 블로그](#)
- [This Is My Architecture 동영상](#)
- [AWS 설명서](#)

# 문서 기록 및 기여자

## 문서 기록

이 백서의 업데이트에 대한 알림을 받으려면 RSS 피드를 구독하세요.

업데이트 기록-변경	update-history-description	update-history-date
<a href="#">마이너 업데이트</a>	조정된 페이지 레이아웃.	2021년 4월 30일
<a href="#">마이너 업데이트</a>	콘텐츠가 최신 모범 사례를 반영하도록 업데이트되었습니다. 콘텐츠를 더 잘 반영하기 위해 백서 이름이 "대규모 Amazon EC2 스팟 인스턴스 활용"에서 "Amazon EC2 스팟 인스턴스 개요"로 변경되었습니다.	2021년 3월 5일
<a href="#">마이너 업데이트</a>	스팟 인스턴스 제한이 업데이트되었습니다.	2021년 2월 3일
<a href="#">첫 게시</a>	대규모로 Amazon EC2 스팟 인스턴스 활용.	2018년 3월 1일

### Note

RSS 업데이트에 가입하려면 사용 중인 브라우저에 대해 RSS 플러그인이 활성화되어 있어야 합니다.

## 기여자

다음은 본 문서를 작성하는 데 도움을 준 개인 및 조직입니다.

- Amilcar Alfaro, Sr. AWS 제품 마케팅 관리자
- Erin Carlson, AWS 마케팅 관리자

- Keith Jarrett, WW BD 책임자 - 비용 최적화, AWS 비즈니스 개발
- Ran Sheinberg, AWS 수석 솔루션스 아키텍트