

비즈니스에 Amazon DynamoDB 데이터 스토리지를 활용하는 모바일 채팅 애플리케이션이 있습니다. 사용자는 새로운 메시지를 읽는 동안 가능한 한 짧은 지연을 원합니다. 솔루션 설계자의 목표는 가능한 최소한의 애플리케이션 수정으로 최적의 솔루션을 제공하는 것입니다.

솔루션 아키텍트가 선택해야 하는 기술은 무엇입니까?

- A. 새 메시지 테이블에 대해 Amazon DynamoDB Accelerator(DAX)를 구성합니다. DAX 끝점을 사용하도록 코드를 업데이트합니다.
- B. 증가된 읽기 로드를 처리하기 위해 DynamoDB 읽기 전용 복제본을 추가합니다. 읽기 전용 복제본의 읽기 엔드포인트를 가리키도록 애플리케이션을 업데이트합니다.
- C. DynamoDB의 새 메시지 테이블에 대한 읽기 용량 단위 수를 두 배로 늘립니다. 기존 DynamoDB 엔드포인트를 계속 사용합니다.
- D. Redis용 Amazon ElastiCache 캐시를 애플리케이션 스택에 추가합니다. DynamoDB 대신 Redis 캐시 엔드포인트를 가리키도록 애플리케이션을 업데이트합니다.

주요 키워드 : **Amazon DynamoDB / DAX / ElastiCache**

문제 요구사항 :

DynamoDB를 이용하여 최소한의 애플리케이션 수정으로 짧은 지연의 읽기 작업을 할 수 있는 솔루션 제공

최소한의 수정으로 지연을 줄일 수 있는 솔루션이 어떤 것이 있을까?

### Amazon DynamoDB

NoSQL 기반 서버리스 키-밸류 데이터베이스

#### 특 징

- 배포와 확장이 단순하고 신속함
- 큰 규모에서도 빠르고 일관된 응답시간
- 온디맨드 비용 지불
- DAX(In-Memory-Cache) 지원

#### 사용 목적

- 대규모 DB 구축 비용 절감
- 모바일 게임, 소셜 네트워크 등 짧은 작업
- 스키마 없는 비정형 데이터 저장
- 자동 백업 지원(사용자가 백업할 필요 없음)

### Amazon DynamoDB Accelerator(DAX)

Amazon DynamoDB를 위한 완전관리형 인 메모리 캐시(In-Memory-Cache)

**\*\* In-Memory-Cache란?** 모든 데이터를 메모리에서 처리 - 엄청 빠르지만 전원 종료 시 날아감

### Amazon ElasticCache

실시간 사용 사례를 지원하는 완전관리형 인 메모리 캐싱 서비스

#### 특 징

- 1밀리초 미만의 매우 빠른 속도
- 저렴한 비용
- Redis 및 Memcached 엔진 지원
- 다중 AZ 및 스냅샷 백업으로 가용성 향상

#### 사용 목적

- 내구성 필요없는 실시간 서비스
- 게이밍, 미디어 스트리밍, 소셜 미디어

비즈니스에 Amazon DynamoDB 데이터 스토리지를 활용하는 모바일 채팅 애플리케이션이 있습니다. 사용자는 새로운 메시지를 읽는 동안 가능한 한 짧은 지연을 원합니다. 솔루션 설계자의 목표는 가능한 최소한의 애플리케이션 수정으로 최적의 솔루션을 제공하는 것입니다.

솔루션 아키텍트가 선택해야 하는 기술은 무엇입니까?

A. 새 메시지 테이블에 대해 Amazon DynamoDB Accelerator(DAX)를 구성합니다. DAX 끝점을 사용하도록 코드를 업데이트합니다.

-> DynamoDB의 데이터 스토리지를 활용해야 하므로 선택 제외(DAX – 인 메모리 캐시 사용)

-> 오답 선택 이유 : DAX를 쓰면 빨라지긴 할 것 같아서 선택

B. 증가된 읽기 로드를 처리하기 위해 DynamoDB 읽기 전용 복제본을 추가합니다. 읽기 전용 복제본의 읽기 엔드포인트를 가리키도록 애플리케이션을 업데이트합니다.

-> 읽기 복제본은 가용성을 늘리기 위한 백업 수단이므로 본문의 조건과 별 상관 없어 보여서 선택 제외

C. DynamoDB의 새 메시지 테이블에 대한 읽기 용량 단위 수를 두 배로 늘립니다. 기존 DynamoDB 엔드포인트를 계속 사용합니다.

-> 읽기 용량 단위를 2배로 늘려 짧은 지연을 달성, 기존 엔드포인트를 계속 사용함으로써 최소한으로 수정을 함(정답)

D. Redis용 Amazon ElastiCache 캐시를 애플리케이션 스택에 추가합니다. DynamoDB 대신 Redis 캐시 엔드포인트를 가리키도록 애플리케이션을 업데이트합니다.

-> DynamoDB의 데이터 스토리지를 활용해야 하므로 선택 제외(ElasticCache – 인 메모리 캐시 사용)



### DynamoDB란? 기본 개념설명

<https://interconnection.tistory.com/60#recentComments>

### Amazon DynamoDB 개발자 안내서

[https://docs.aws.amazon.com/ko\\_kr/amazondynamodb/latest/developerguide/Introduction.html](https://docs.aws.amazon.com/ko_kr/amazondynamodb/latest/developerguide/Introduction.html)

기업은 확장성 및 가용성 요구 사항을 충족하기 위해 컨테이너에서 미션 크리티컬 앱을 실행하려고 합니다. 회사는 오히려 주요 애플리케이션 유지 관리에 집중할 것입니다. 회사는 컨테이너화된 워크로드의 기본 인프라 프로비저닝 및 유지 관리에 대한 책임을 원하지 않습니다.

이러한 기준이 충족되도록 솔루션 설계자는 어떤 조치를 취해야 할까요?

- A. Amazon EC2 인스턴스를 사용하고 인스턴스에 Docker를 설치합니다.
- B. Amazon EC2 작업자 노드에서 Amazon Elastic Container Service(Amazon ECS)를 사용합니다.
- C. AWS Fargate에서 Amazon Elastic Container Service(Amazon ECS)를 사용합니다.
- D. Amazon Elastic Container Service(Amazon ECS)에 최적화된 Amazon 머신 이미지(AMI)의 Amazon EC2 인스턴스를 사용합니다.

주요 키워드 : **Amazon ECS / AWS Fargate**

문제 요구사항 :

프로비저닝 및 유지 관리 책임을 지지 않으며 컨테이너를 운영하려고 함

기업이 직접 프로비저닝이나 유지관리를 하지않고 컨테이너를 구동하려면 어떻게해야할까?

### Amazon Elastic Container Service

완전관리형 컨테이너 오케스트레이션 서비스로서 쉽게 컨테이너 워크로드 배포 가능

### Amazon Fargate

컨테이너 서비스를 운영가능한 종량제 서버리스 컴퓨팅 엔진

- ECS, EKS(Elastic Kubernetes Service)와 호환이 가능하여 프로비저닝, 관리 없이 컨테이너 사용 가능
- Amazon CloudWatch Container Insights 등 AWS 서비스와의 통합을 통해 애플리케이션을 모니터링
- 사용한 리소스에 대해서만 비용을 지불

기업은 확장성 및 가용성 요구 사항을 충족하기 위해 컨테이너에서 미션 크리티컬 앱을 실행하려고 합니다. 회사는 오히려 주요 애플리케이션 유지 관리에 집중할 것입니다. 회사는 컨테이너화된 워크로드의 기본 인프라 프로비저닝 및 유지 관리에 대한 책임을 원하지 않습니다.

이러한 기준이 충족되도록 솔루션 설계자는 어떤 조치를 취해야 할까요?

A. Amazon EC2 인스턴스를 사용하고 인스턴스에 Docker를 설치합니다.

-> EC2에서 Docker를 사용할 수 있지만 직접 프로비저닝을 해줘야하므로 선택 제외

B. Amazon EC2 작업자 노드에서 Amazon Elastic Container Service(Amazon ECS)를 사용합니다.

-> EC2에서 ECS를 사용할 수 있지만 직접 프로비저닝을 해줘야하므로 선택 제외

C. AWS Fargate에서 Amazon Elastic Container Service(Amazon ECS)를 사용합니다.

-> AWS Fargate에서 ECS를 실행함으로써 추가 프로비저닝 없이 무한으로 즐길 수 있음 (정답)

D. Amazon Elastic Container Service(Amazon ECS)에 최적화된 Amazon 머신 이미지(AMI)의 Amazon EC2 인스턴스를 사용합니다.

-> EC2에서 ECS를 사용할 수 있지만 직접 프로비저닝을 해줘야하므로 선택 제외





### **AWS Fargate**

<https://aws.amazon.com/ko/fargate/?nc=sn&loc=1>

### **AWS Elastic Container Service**

<https://aws.amazon.com/ko/ecs/features/>

기업의 백업 데이터는 총 700TB이며 데이터 센터의 NAS(Network Attached Storage)에 보관됩니다. 이 백업 데이터는 규제 관련 문의가 있을 경우 사용할 수 있어야 하며 7년 동안 보존해야 합니다. 조직은 백업 데이터를 온프레미스 데이터 센터에서 Amazon Web Services(AWS)로 재배치하기로 결정했습니다. 한 달 이내에 마이그레이션을 완료해야 합니다. 회사의 공용 인터넷 연결은 500Mbps의 데이터 전송 전용 용량을 제공합니다.

가장 낮은 비용으로 데이터를 마이그레이션하고 저장하기 위해 솔루션 설계자는 무엇을 해야 할까요?

- A. AWS Snowball 디바이스에 데이터를 전송하도록 주문합니다. 수명 주기 정책을 사용하여 파일을 Amazon S3 Glacier Deep Archive로 전환합니다.
- B. 데이터 센터와 Amazon VPC 간에 VPN 연결을 배포합니다. AWS CLI를 사용하여 온프레미스에서 Amazon S3 Glacier로 데이터를 복사합니다.
- C. 500Mbps AWS Direct Connect 연결을 프로비저닝하고 데이터를 Amazon S3로 전송합니다. 수명 주기 정책을 사용하여 파일을 Amazon S3 Glacier Deep Archive로 전환합니다.
- D. AWS DataSync를 사용하여 데이터를 전송하고 DataSync 에이전트를 온프레미스에 배포합니다. DataSync 작업을 사용하여 온프레미스 NAS 스토리지에서 Amazon S3 Glacier로 파일을 복사합니다.

주요 키워드 : **Amazon Snowball / AWS S3**

문제 요구사항 :

700TB의 데이터를 AWS로 마이그레이션 해야함. 해당 데이터는 필요한 경우 사용 가능해야하며 7년 장기 보관

가장 낮은 비용으로 대용량의 데이터를 마이그레이션 및 장기보관 스토리징 하기 위한 방법은?

### Amazon Snow Family

물리적 디바이스를 이용하여 네트워크 연결이 어려운 경우에 데이터를 직접 옮길 수 있는 서비스

#### AWS Snowcone

- 8 / 14TB의 물리적 스토리지 디바이스
- IoT의 경우 차량 및 드론 사용 가능
- 2중 암호화 제공

#### AWS Snowball

- 1주 만에 TB규모의 데이터를 이동 가능
- 최대 52개의 vCPU 제공
- 머신러닝 및 풀모션 미디어 분석 가능

#### AWS Snowmobile

- 엑사바이트 규모 초용량 전송 서비스
- 실제 대형 운반 트레일러로 이전

### Amazon S3 Glacier 스토리지 클래스

장기 보관용 데이터 아카이빙 스토리지 – 자주 안쓰고 오래 보관만 하는 데이터를 위한 솔루션

#### Instant Retrieval

- 분기당 한 번 액세스, ms단위의 검색이 필요한 수명이 긴 데이터
- 필요한 경우 즉시 액세스가 가능

#### Flexible Retrieval

- 연간 1~2회 액세스, 비동기식 검색
- 즉각 액세스가 필요없는 백업, 재해 복구 사용 사례

#### Deep Archive

- 연 1회 미만 액세스, 비동기식 검색
- 7~10년 이상 보관

기업의 백업 데이터는 총 700TB이며 데이터 센터의 NAS(Network Attached Storage)에 보관됩니다. 이 백업 데이터는 규제 관련 문의가 있을 경우 사용할 수 있어야 하며 7년 동안 보존해야 합니다. 조직은 백업 데이터를 온프레미스 데이터 센터에서 Amazon Web Services(AWS)로 재배치하기로 결정했습니다. 한 달 이내에 마이그레이션을 완료해야 합니다. 회사의 공용 인터넷 연결은 500Mbps의 데이터 전송 전용 용량을 제공합니다.

가장 낮은 비용으로 데이터를 마이그레이션하고 저장하기 위해 솔루션 설계자는 무엇을 해야 할까요?

A. AWS Snowball 디바이스에 데이터를 전송하도록 주문합니다. 수명 주기 정책을 사용하여 파일을 Amazon S3 Glacier Deep Archive로 전환합니다.

-> 700TB는 네트워크로 마이그레이션하기엔 큼. Snowball로 이전 후 Glacier로 스토리지를 하면 됨(정답)

B. 데이터 센터와 Amazon VPC 간에 VPN 연결을 배포합니다. AWS CLI를 사용하여 온프레미스에서 Amazon S3 Glacier로 데이터를 복사합니다.

-> 네트워크를 통한 마이그레이션을 하기엔 너무 규모가 커서 선택 제외

C. 500Mbps AWS Direct Connect 연결을 프로비저닝하고 데이터를 Amazon S3로 전송합니다. 수명 주기 정책을 사용하여 파일을 Amazon S3 Glacier Deep Archive로 전환합니다.

-> 네트워크를 통한 마이그레이션을 하기엔 너무 규모가 커서 선택 제외

D. AWS DataSync를 사용하여 데이터를 전송하고 DataSync 에이전트를 온프레미스에 배포합니다. DataSync 작업을 사용하여 온프레미스 NAS 스토리지에서 Amazon S3 Glacier로 파일을 복사합니다.

-> 네트워크를 통한 마이그레이션을 하기엔 너무 규모가 커서 선택 제외



### Amazon Snow Family

<https://aws.amazon.com/ko/snow/>

기업은 다양한 가용 영역에 걸쳐 있는 가상 사설 클라우드(VPC)에서 3계층 웹 애플리케이션을 호스팅합니다. 애플리케이션 계층의 경우 Amazon EC2 인스턴스는 Auto Scaling 그룹에 배포됩니다.

조직은 각 리소스에 대한 일일 및 주간 워크로드 패턴을 분석하는 자동화된 확장 전략을 개발해야 합니다. 설정은 소비의 예측 및 실제 변경 모두에 대응하여 리소스를 올바르게 확장해야 합니다.

이러한 요구 사항을 충족하기 위해 솔루션 설계자가 제안해야 하는 확장 접근 방식(있는 경우)은 무엇입니까?

- A. EC2 인스턴스의 평균 CPU 사용률을 기반으로 한 단계적 확장으로 동적 확장을 구현합니다.
- B. 예측 및 확장을 위해 예측 확장을 활성화합니다. 대상 추적으로 동적 확장을 구성합니다.
- C. 웹 애플리케이션의 트래픽 패턴을 기반으로 자동화된 예약된 조정 작업을 생성합니다.
- D. 간단한 확장 정책을 설정합니다. EC2 인스턴스 시작 시간에 따라 휴지 기간을 늘립니다.

주요 키워드 : **Auto-Scaling**

문제 요구사항 :

Multi-AZ VPC에서 자동화된 확장 전략 개발

일정한 주기별로 워크로드 패턴을 분석하여 리소스를 확장할 수 있는 방법은?

### Amazon Auto Scaling

#### 동적 조정 정책 유형

- 대상 추적 조정 : 목표값을 기준으로 그룹의 현재 용량을 늘리거나 줄임
- 단계 조정 : 현재 용량을 일련의 조정 조절에 따라 늘리거나 줄이며 경보 위반의 크기에 따라 달라짐
- 단순 조정 : 단일 조정 설정에 따라 그룹의 현재 용량을 늘리거나 줄임

#### 예측 조정 정책 유형

- 트래픽 흐름의 일일 및 주간 패턴에 앞서 Auto Scaling 그룹의 EC2 인스턴스 수 조절 가능
- 주기적 트래픽이나 반복적인 온/오프 워크로드 패턴의 경우 유리함

기업은 다양한 가용 영역에 걸쳐 있는 가상 사설 클라우드(VPC)에서 3계층 웹 애플리케이션을 호스팅합니다. 애플리케이션 계층의 경우 Amazon EC2 인스턴스는 Auto Scaling 그룹에 배포됩니다.

조직은 각 리소스에 대한 일일 및 주간 워크로드 패턴을 분석하는 자동화된 확장 전략을 개발해야 합니다. 설정은 소비의 예측 및 실제 변경 모두에 대응하여 리소스를 올바르게 확장해야 합니다.

이러한 요구 사항을 충족하기 위해 솔루션 설계자가 제안해야 하는 확장 접근 방식(있는 경우)은 무엇입니까?

A. EC2 인스턴스의 평균 CPU 사용률을 기반으로 한 단계적 확장으로 동적 확장을 구현합니다.

-> 단순 평균 사용률은 예측 조정을 할 수 없어서 선택 제외

B. 예측 및 확장을 위해 예측 확장을 활성화합니다. 대상 추적으로 동적 확장을 구성합니다.

-> 실제 변경치, 예측치 전부 대응을 위해 예측 조정, 동적 조정 실시(정답)

C. 웹 애플리케이션의 트래픽 패턴을 기반으로 자동화된 예약된 조정 작업을 생성합니다.

-> 예측 조정을 할 수 없어서 선택 제외

D. 간단한 확장 정책을 설정합니다. EC2 인스턴스 시작 시간에 따라 휴지 기간을 늘립니다.

-> 너무 간단해서 선택 제외

\*\* 휴지 기간(Scale cooldown) : 직전 조정치가 반영되기 전에 인스턴스 수를 늘이거나 줄이는 것을 잠시 보류하는 것





기업은 이벤트 데이터를 생성하는 서비스를 운영합니다. 회사는 이벤트 데이터를 수신하는 대로 처리하기 위해 AWS를 사용하고자 합니다. 데이터는 처리 중에 보존되어야 하는 특정 순서로 구조화됩니다. 회사는 가능한 가장 낮은 운영 비용으로 솔루션을 배포하기를 원합니다.

솔루션 아키텍트가 이 작업을 어떻게 수행합니까?

- A. 메시지를 보관할 Amazon Simple Queue Service(Amazon SQS) FIFO 대기열을 생성합니다. 대기열의 메시지를 처리하도록 AWS Lambda 함수를 설정합니다.
- B. 처리할 페이로드가 포함된 알림을 전달하기 위해 Amazon Simple Notification Service(Amazon SNS) 주제를 생성합니다. AWS Lambda 함수를 구독자로 구성합니다.
- C. 메시지를 보관할 Amazon Simple Queue Service(Amazon SQS) 표준 대기열을 생성합니다. 대기열의 메시지를 독립적으로 처리하도록 AWS Lambda 함수를 설정합니다.
- D. 처리할 페이로드가 포함된 알림을 전달하기 위해 Amazon Simple Notification Service(Amazon SNS) 주제를 생성합니다. Amazon Simple Queue Service(Amazon SQS) 대기열을 구독자로 구성합니다.

주요 키워드 : AWS SQS

문제 요구사항 :

데이터를 수신하는 순서대로 처리 / 낮은 비용

선입 선출로 데이터를 처리할 수 있는데 값이 싼 솔루션은?

기업은 이벤트 데이터를 생성하는 서비스를 운영합니다. 회사는 이벤트 데이터를 수신하는 대로 처리하기 위해 AWS를 사용하고자 합니다. 데이터는 처리 중에 보존되어야 하는 특정 순서로 구조화됩니다. 회사는 가능한 가장 낮은 운영 비용으로 솔루션을 배포하기를 원합니다.

솔루션 아키텍트가 이 작업을 어떻게 수행합니까?

A. 메시지를 보관할 Amazon Simple Queue Service(Amazon SQS) FIFO 대기열을 생성합니다. 대기열의 메시지를 처리하도록 AWS Lambda 함수를 설정합니다. -> **SQS FIFO(선입선출)로 데이터 처리 / Lambda를 이용하여 서버리스로 저렴하게 배포(정답)**

B. 처리할 페이로드가 포함된 알림을 전달하기 위해 Amazon Simple Notification Service(Amazon SNS) 주제를 생성합니다. AWS Lambda 함수를 구독자로 구성합니다. -> **FIFO 조건이 반영되지않아서 선택 제외**

C. 메시지를 보관할 Amazon Simple Queue Service(Amazon SQS) 표준 대기열을 생성합니다. 대기열의 메시지를 독립적으로 처리하도록 AWS Lambda 함수를 설정합니다. -> **FIFO 조건이 반영되지않아서 선택 제외**

D. 처리할 페이로드가 포함된 알림을 전달하기 위해 Amazon Simple Notification Service(Amazon SNS) 주제를 생성합니다. Amazon Simple Queue Service(Amazon SQS) 대기열을 구독자로 구성합니다.

-> **FIFO 조건이 반영되지않아서 선택 제외**



비즈니스는 두 개의 Amazon EC2 인스턴스를 사용하여 동적 웹 애플리케이션을 실행합니다. 조직에는 각 인스턴스에서 SSL 종료를 완료하는 데 사용되는 자체 SSL 인증서가 있습니다.

최근 트래픽이 증가하고 있으며, 운영팀은 SSL 암호화 및 복호화로 인해 웹 서버의 컴퓨팅 용량이 한도를 초과한다는 결론을 내렸습니다.

솔루션 설계자는 애플리케이션의 성능을 최적화하기 위해 무엇을 해야 할까요?

- A. AWS Certificate Manager(ACM)를 사용하여 새 SSL 인증서를 생성합니다. 각 인스턴스에 ACM 인증서를 설치합니다.
- B. Amazon S3 버킷을 생성합니다. SSL 인증서를 S3 버킷으로 마이그레이션합니다. SSL 종료를 위해 버킷을 참조하도록 EC2 인스턴스를 구성합니다.
- C. 다른 EC2 인스턴스를 프록시 서버로 생성합니다. SSL 인증서를 새 인스턴스로 마이그레이션하고 기존 EC2 인스턴스에 직접 연결하도록 구성합니다.
- D. SSL 인증서를 AWS Certificate Manager(ACM)로 가져옵니다. ACM의 SSL 인증서를 사용하는 HTTPS 리스너로 Application Load Balancer를 생성합니다.

주요 키워드 : **Certificate Manager**

문제 요구사항 :

자체 SSL 인증서를 처리하는데 컴퓨팅 용량이 부족해짐

암호화 관련 어플리케이션을 더욱 최적화 하는 방법?

### AWS Certificate Manager

SSL/TLS X.509 인증서와 키 생성, 저장, 갱신하는 서비스

- 직접 ACM에서 인증서를 발급하거나 다른 곳에서 가져올 수 있음

비즈니스는 두 개의 Amazon EC2 인스턴스를 사용하여 동적 웹 애플리케이션을 실행합니다. 조직에는 각 인스턴스에서 SSL 종료를 완료하는 데 사용되는 자체 SSL 인증서가 있습니다.

최근 트래픽이 증가하고 있으며, 운영팀은 SSL 암호화 및 복호화로 인해 웹 서버의 컴퓨팅 용량이 한도를 초과한다는 결론을 내렸습니다.

솔루션 설계자는 애플리케이션의 성능을 최적화하기 위해 무엇을 해야 할까요?

A. AWS Certificate Manager(ACM)를 사용하여 새 SSL 인증서를 생성합니다. 각 인스턴스에 ACM 인증서를 설치합니다.

-> 인증서로 인한 컴퓨팅 용량이 오히려 줄어들 것 같아서 선택 제외

B. Amazon S3 버킷을 생성합니다. SSL 인증서를 S3 버킷으로 마이그레이션합니다. SSL 종료를 위해 버킷을 참조하도록 EC2 인스턴스를 구성합니다.

C. 다른 EC2 인스턴스를 프록시 서버로 생성합니다. SSL 인증서를 새 인스턴스로 마이그레이션하고 기존 EC2 인스턴스에 직접 연결하도록 구성합니다.

D. SSL 인증서를 AWS Certificate Manager(ACM)로 가져옵니다. ACM의 SSL 인증서를 사용하는 HTTPS 리스너로 Application Load Balancer를 생성합니다.

-> ACM으로 SSL 인증서를 가져온 뒤 ELB로 트래픽을 분산하여 컴퓨팅 용량 확보(정답)



### **Auto Scaling**

[https://docs.aws.amazon.com/ko\\_kr/autoscaling/ec2/userguide/as-scale-based-on-demand.html](https://docs.aws.amazon.com/ko_kr/autoscaling/ec2/userguide/as-scale-based-on-demand.html)

### **Auto Scaling**

[https://docs.aws.amazon.com/ko\\_kr/autoscaling/ec2/userguide/scaling\\_plan.html](https://docs.aws.amazon.com/ko_kr/autoscaling/ec2/userguide/scaling_plan.html)