

AWS 아키텍처 설계

Chapter 08. Elasticity

설계 시나리오

회사가 AWS를 사용하여 전자상거래 서비스를 구축하였고, 수만 명 고객들의 요청이 시간과 요일에 따라 큰 폭으로 변하고 있는 상황이다.

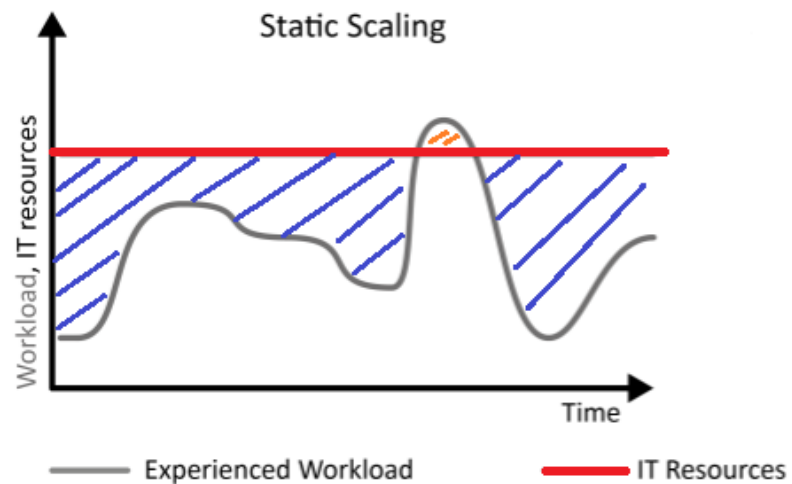
AWS 클라우드 환경에서 이런 상황을 안정적으로 처리할 수 있는 서비스는 무엇이 있을까?

01

탄력성(Elasticity)의 필요성

온프레미스 환경

- 온프레미스 데이터센터 환경의 경우 서버 구매 시 몇 주에서 몇 달이 소요.
- 서버 조달이나 추가 변경에 필요한 시간/비용이 크기 때문에 사전에 피크 타임에 맞춰 리소스를 준비.
- 잘못 계산된 용량 추정에 의해 비용 낭비 혹은 용량 부족 상황이 발생.



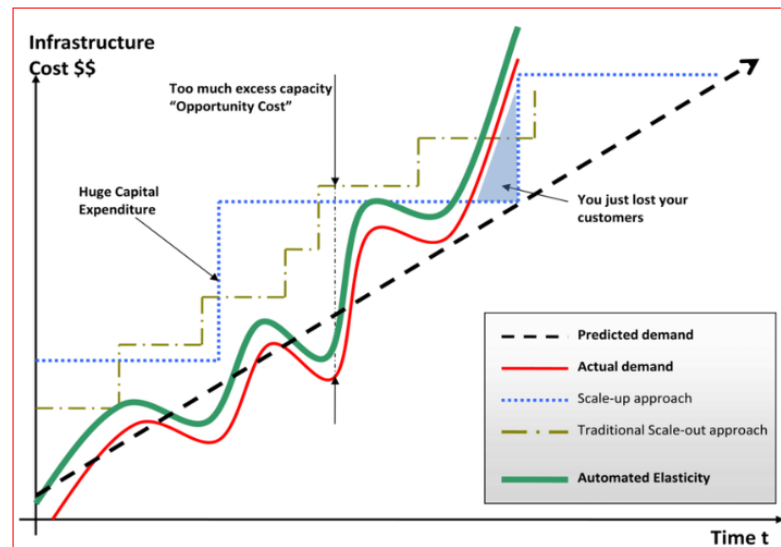
Amazon.com 트래픽 (11월)



11월

탄력성(Elasticity)

- 온프레미스 환경에서 용량 추정을 하는 것은 어렵다.
- AWS 클라우드 환경에서는 **탄력성**을 제공.
- 애플리케이션 설계 변경 없이 용량 요구 변화에 따라 자동으로 확장 및 축소를 수행.

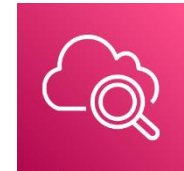


02

AWS 모니터링 서비스

Amazon CloudWatch

- CloudWatch는 로그, 지표 및 이벤트 형태로 모니터링 및 운영 데이터를 수집.
- 로그와 지표 시각화.
- 이상 동작을 감지하며, 경보를 설정하고, 자동화된 작업 수행.

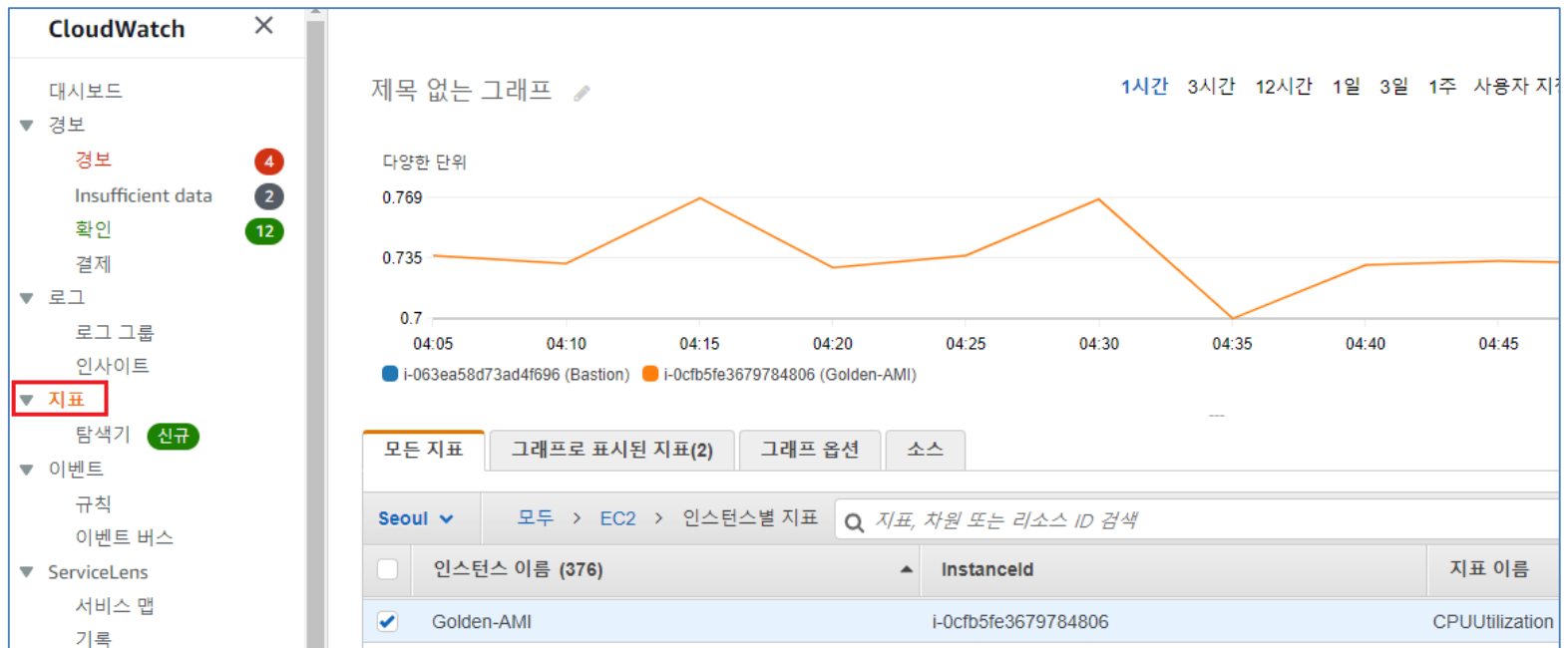


Amazon
CloudWatch



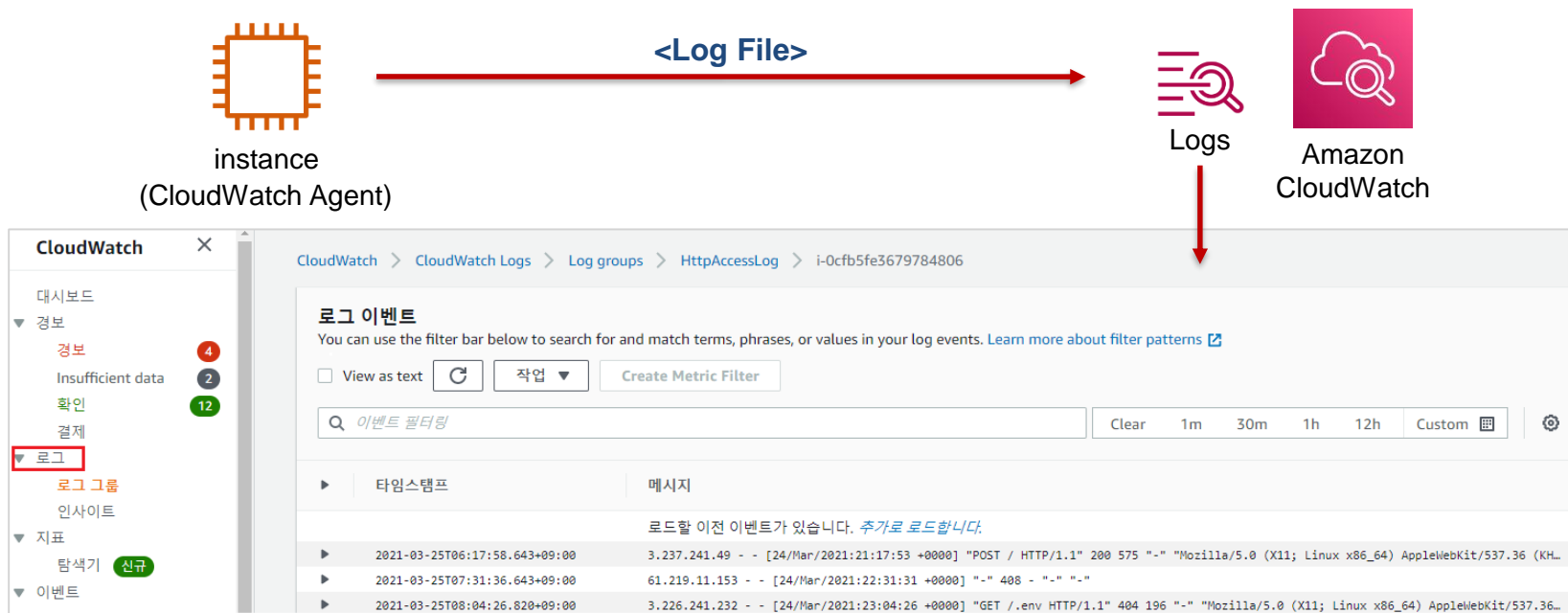
Amazon CloudWatch Metric(지표)

- CloudWatch 지표(metric)는 시스템의 성능에 대한 데이터 값.



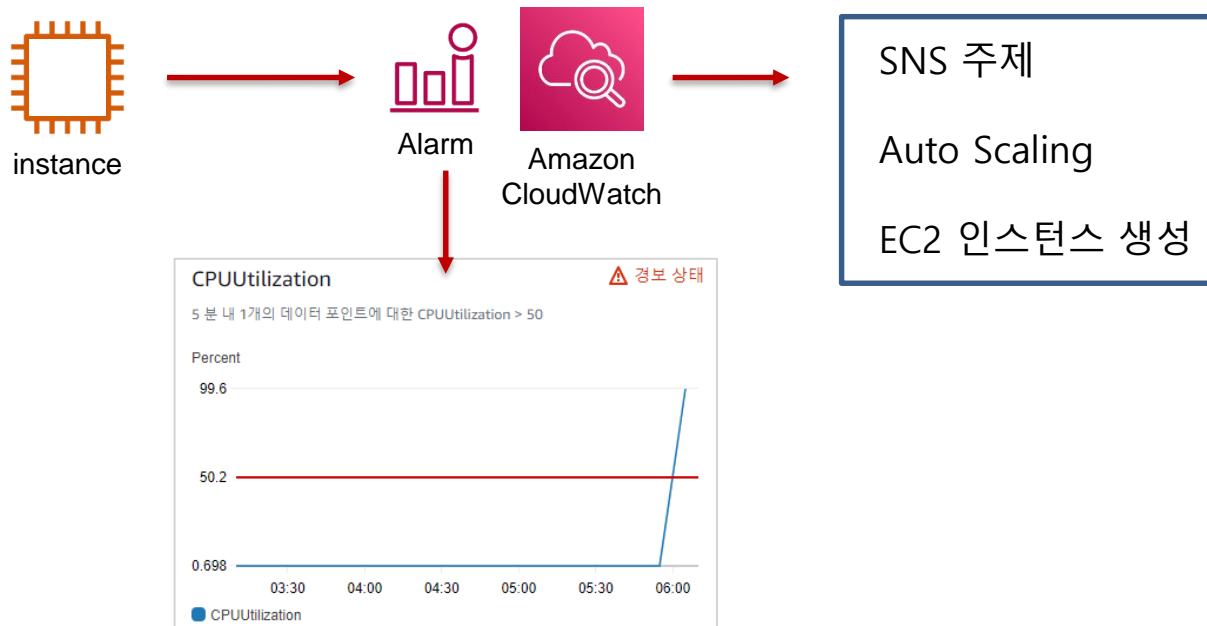
Amazon CloudWatch Logs

- CloudWatch Logs를 사용하면 AWS 서비스에서 로그를 수집, 저장.
- VPC Flow Logs / CloudTrail 로그/ 인스턴스에서 애플리케이션 로그 수집 및 Route 53 DNS 쿼리 기록 등을 수행.



Amazon CloudWatch Alarms (경보)

- 경보를 사용하여 작업을 자동으로 시작할 수 있다
- 지정한 기간에 단일 지표를 감시, 특정 지표에 정의된 임계 값을 기준으로 관리자가 지정한 작업을 하나 이상 수행.
 - Amazon SNS 주제 알림 / Auto Scaling 작업 / EC 작업



Amazon CloudWatch Events (이벤트)

- 이벤트를 감지하고 지정된 규칙에 따라 처리하는 서비스.
- 이벤트 패턴(AWS 리소스 변경 발생) / 스케줄 지정.
- 감지된 이벤트가 규칙에 해당되면 대상으로 지정한 다른 서비스를 호출.

The screenshot displays the Amazon CloudWatch Events console interface. On the left, a navigation pane shows various CloudWatch services, with '이벤트' (Events) highlighted. The main area is titled '이벤트 소스' (Event Source) and contains the following sections:

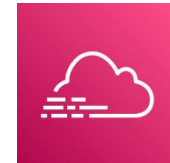
- 이벤트 패턴** (Event Pattern): A section for defining the event pattern. It includes a dropdown for '서비스별 이벤트와 일치시킬 이벤트 패턴을 작성' (Specify an event pattern to match with service events) and a dropdown for '이벤트 유형' (Event type) set to 'EC2 Instance State-change Notification'. Below this, there are radio buttons for '모든 상태' (All states) and '특정 상태' (Specific states), with '특정 상태' selected. A list of states is shown: 'stopped' and 'stopping'. There are also radio buttons for '모든 인스턴스' (All instances) and '특정 인스턴스 ID' (Specific instance ID), with '모든 인스턴스' selected.
- 이벤트 패턴 미리 보기** (Event Pattern Preview): A section showing the JSON representation of the event pattern. The JSON is as follows:

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

On the right side of the console, the '대상' (Target) section is visible. It includes a dropdown for 'SNS 주제' (SNS topic) and a dropdown for '주제*' (Topic*) set to 'Instance_state_has_changed'. There is also a button for '대상 추가*' (Add target*).

AWS CloudTrail

- 보안, 감사 목적으로 AWS 계정에서 수행된 활동의 상세 기록을 보관.
- AWS 계정에서 발생한 API 호출 내역을 기록.
- CloudTrail을 사용하여 AWS 계정의 비정상적인 활동을 탐지.



AWS
CloudWatch

이벤트 기록 (1050+) Info

이벤트 기록에는 지난 90일 동안의 관리 이벤트가 표시됩니다.

읽기 전용

Q false

30m1h3h12hCustom

이벤트 이름

이벤트 시간

사용자 이름

이벤트 소스

리소스 유형

리소스 이름

DeleteLoadBalancer

March 26, 2021, 11:34:44 (UTC+09:00)

root

elasticloadbalancing.amazo

AWS::ElasticLoadBala...

arn:aws:elasticloadbalancing:a...

DeleteLaunchTemplate

March 26, 2021, 11:34:26 (UTC+09:00)

root

ec2.amazonaws.com

AWS::EC2::LaunchTem...

lt-089a6711bbe2d2ada, Lab-T...

TerminateInstances

March 26, 2021, 11:34:05 (UTC+09:00)

AutoScaling

ec2.amazonaws.com

AWS::EC2::Instance

i-04c9ac03b3322f57e

TerminateInstances

March 26, 2021, 11:34:02 (UTC+09:00)

AutoScaling

ec2.amazonaws.com

AWS::EC2::Instance

i-095590edd9bd4fe9a

UpdateInstanceInfor...

March 26, 2021, 11:32:03 (UTC+09:00)

i-04c9ac03b3322f57e

ssm.amazonaws.com

-

-

DeregisterTargets

March 26, 2021, 11:31:58 (UTC+09:00)

AutoScaling

elasticloadbalancing.amazo

AWS::EC2::Instance, A...

i-0a1ab3032aed86870, arn:aw...

DeregisterTargets

March 26, 2021, 11:31:58 (UTC+09:00)

AutoScaling

elasticloadbalancing.amazo

AWS::EC2::Instance, A...

i-011db72d74e4a0cfe, arn:aw...

DeleteAutoScalingGr...

March 26, 2021, 11:31:46 (UTC+09:00)

root

autoscaling.amazonaws.cor

AWS::AutoScaling::Au...

Inventory-ASG

VPC Flow Logs

- VPC의 네트워크 인터페이스에서 전송되고 수신되는 IP 트래픽에 대한 정보를 수집.
- Amazon CloudWatch Logs 또는 Amazon S3에 게시.
- 허용된 트래픽 / 거부된 트래픽 / 모든 트래픽 중 선택.
- VPC / Subnet / ENI에서 활성화 가능.



Flow logs

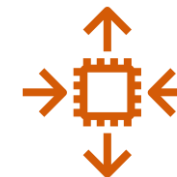
▶	2020-09-03T15:57:16.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	45.148.10.28	172.31.46.22	38871	22	6	2	80	1599116236	1599116288	ACCEPT	OK
▶	2020-09-03T15:57:16.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	172.31.46.22	192.227.147.110	22	33490	6	6	264	1599116236	1599116288	ACCEPT	OK
▶	2020-09-03T15:57:16.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	192.227.147.110	172.31.46.22	33490	22	6	1	40	1599116236	1599116288	ACCEPT	OK
▶	2020-09-03T15:57:16.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	172.31.46.22	45.148.10.28	22	38871	6	1	44	1599116236	1599116288	ACCEPT	OK
▶	2020-09-03T15:58:09.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	172.31.46.22	45.148.10.28	22	58282	6	12	2917	1599116289	1599116348	ACCEPT	OK
▶	2020-09-03T15:58:09.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	172.31.46.22	210.108.48.236	80	1746	6	8	1030	1599116289	1599116348	ACCEPT	OK
▶	2020-09-03T15:58:09.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	45.148.10.28	172.31.46.22	58282	22	6	14	1915	1599116289	1599116348	ACCEPT	OK
▶	2020-09-03T15:58:09.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	172.31.46.22	45.148.10.28	22	34954	6	4	237	1599116289	1599116348	ACCEPT	OK
▶	2020-09-03T15:58:09.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	45.148.10.28	172.31.46.22	34954	22	6	5	256	1599116289	1599116348	ACCEPT	OK
▶	2020-09-03T15:58:09.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	210.108.48.236	172.31.46.22	1746	80	6	12	2908	1599116289	1599116348	ACCEPT	OK
▶	2020-09-03T15:58:09.000+09:00	2	274728859905	eni-0dbca8c1cd71e8c5e	172.31.46.22	210.108.48.236	80	1745	6	4	196	1599116289	1599116408	ACCEPT	OK

03

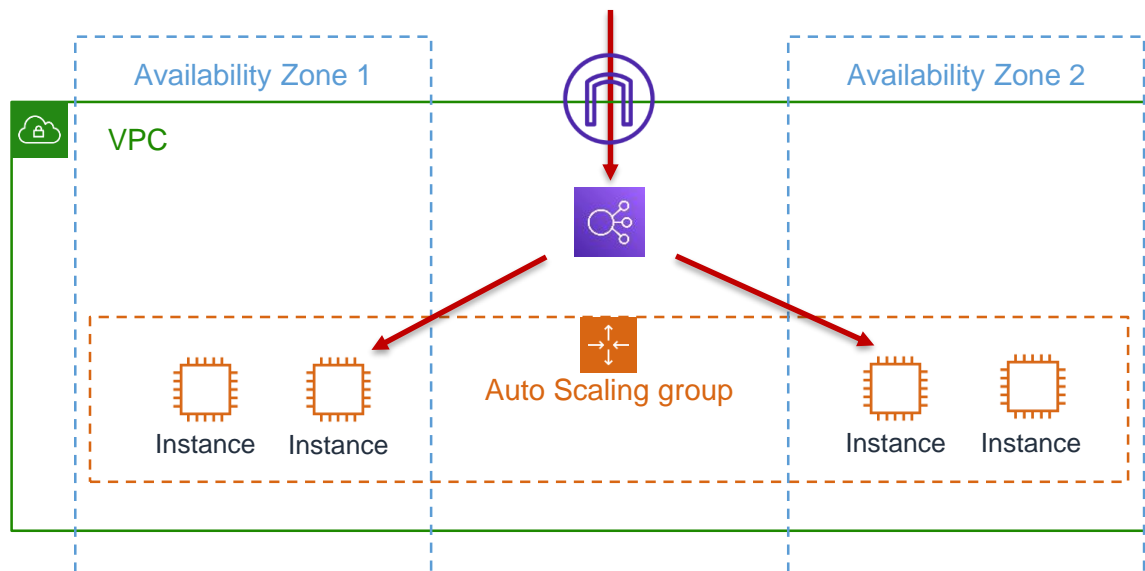
Amazon EC2 Auto Scaling

Amazon EC2 Auto Scaling

- 시작 템플릿으로 자동으로 구동될 EC2 인스턴스 내용 정의.
- 오토스케일링 정책으로 인스턴스가 자동으로 시작/종료될 조건을 정의.
- 자동 생성된 인스턴스를 로드 밸런서 Target Group과 매핑하면 트래픽 분산 대상으로 자동 등록.
- 다수의 가용 영역에 걸쳐서 구성 가능.

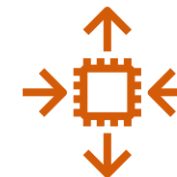


Auto Scaling



Amazon EC2 Auto Scaling

- 내결함성 향상 / 가용성 향상 / 비용 절감.
- 자동 조정 방법 - 예약 / 동적 / 예측
- 동일 Auto Scaling 그룹 안에서 구매 옵션 선택 가능.



Auto Scaling

인스턴스 배포

온디맨드 기본 용량 - 선택 사항
Auto Scaling 그룹이 기본적으로 보유해야 하는 온디맨드 용량을 지정합니다. 최대 그룹 크기는 이 값까지 증가합니다(단, 감소하지는 않음).

온디맨드 인스턴스

기본 용량을 초과하는 온디맨드 비율(%)
기본 용량 부분 이외의 추가 용량을 온디맨드 인스턴스와 스팟 인스턴스로 분할한 비율(%)입니다.

% 온디맨드

% 스팟

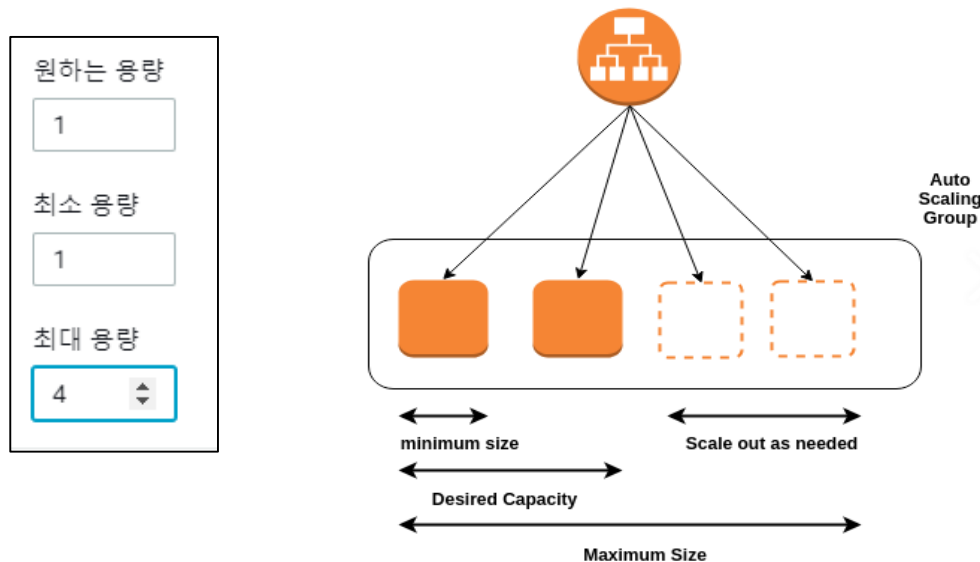
가용 영역당 스팟 할당 전략

☒ 용량 최적화(권장)
사용 가능한 스팟 용량에 기반하여 최적의 상태로 스팟 인스턴스를 시작합니다.

☐ 최저 가격
최저 가격 인스턴스 풀에서 스팟 인스턴스를 시작합니다.

Amazon EC2 Auto Scaling

- Auto Scaling 그룹 설정 시 원하는 용량 / 최소 용량 / 최대 용량을 지정.



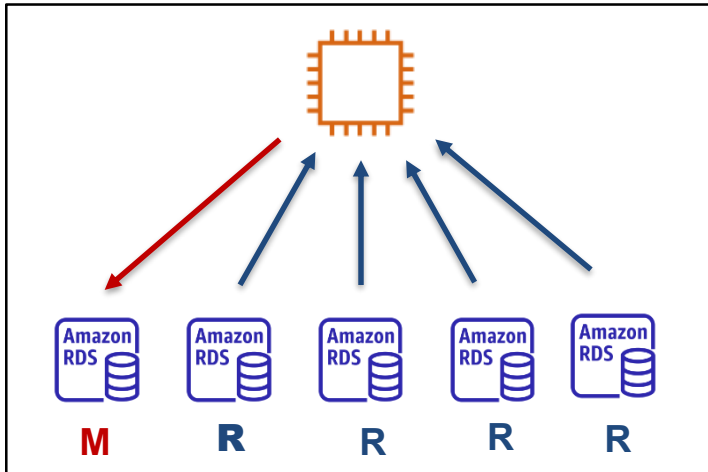
- 적절한 **최소 용량** / **최대 용량**은 어떻게 정해야 하는가?
- 조기에 빠르게 확장 / 필요한 경우 **수명 주기 후크** 사용

04

데이터베이스 용량 조정

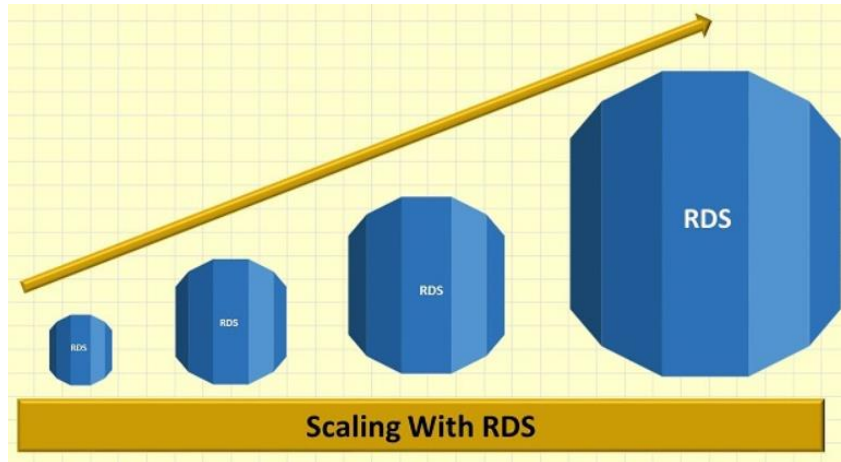
Amazon RDS 수평적 용량 확장

- RDS의 경우 **읽기 전용 복제본**을 사용하여 **수평적 확장** 가능. (비동기식 복제)
- 읽기 요청에 대한 부하를 분산 처리.
- 마스터 DB 장애 발생시 읽기 전용 복제본 중 하나를 후 마스터로 승격.
- Aurora DB 클러스터는 경우 최대 15개의 복제본 생성 가능. (나머지는 5개)



Amazon RDS 수직적 용량 확장

- RDS의 경우 클릭 몇 번으로 DB 인스턴스 용량을 수직적으로 확장 가능.
- 몇 분의 짧은 다운 타임이 발생.
- RDS Multi-AZ 구성인 경우 다운 타임 없이 수직적으로 용량 조정도 가능.



DynamoDB 용량 조정

• DynamoDB에는 두 가지 용량 모드가 존재.

- 프로비저닝 용량 모드 : 초당 읽기 및 쓰기 수를 지정. Auto Scaling으로 자동 조정 가능.
- 온디맨드 용량 모드 : 애플리케이션이 수행하는 모든 읽기 및 쓰기에 대한 요금이 부과.

DynamoDB > 테이블 > hello > 용량 편집

읽기/쓰기 용량 편집

용량 모드 정보

☐ 온디맨드
애플리케이션이 수행하는 실제 읽기 및 쓰기에 대해 요금을 지불하는 방식으로 결제를 간소화합니다.

☒ 프로비저닝됨
읽기/쓰기 용량을 미리 할당하는 방식으로 요금을 관리하고 최적화합니다.

테이블 용량

읽기 용량

Auto Scaling [정보](#)
실제 트래픽 패턴에 따라, 사용자를 대신하여 프로비저닝 처리 용량을 동적으로 조정합니다.

☒ 켜기
☐ 끄기

최소 용량 단위
1

최대 용량 단위
10

목표 사용률(%)
70

쓰기 용량

Auto Scaling [정보](#)
실제 트래픽 패턴에 따라, 사용자를 대신하여 프로비저닝 처리 용량을 동적으로 조정합니다.

☒ 켜기
☐ 끄기

최소 용량 단위
1

최대 용량 단위
10

목표 사용률(%)
70

예상 읽기/쓰기 용량 비용

현재 설정에 따라, 테이블 및 인덱스에 대해 프로비저닝된 읽기 및 쓰기 용량의 총 예상 비용은 다음과 같습니다. 자세한 내용은 프로비저닝된 용량에 대한 [Amazon DynamoDB 요금](#) [리소스](#)를 참조하세요.

총 읽기 용량 단위 5	총 쓰기 용량 단위 5	리전 us-east-1	추정 비용 US\$2.91/월
-----------------	-----------------	-----------------	---------------------