

# Chapter 10. 가상 머신 탄력성 구현

발표자 | 신건우

## Outline

# 목차

- 가상 머신 확장 집합의 개념과 아키텍처
- 가상 머신 확장 집합 배포를 위한 주요 구성요소
- 가상 머신 확장 집합 설정 및 관리 작업



## Practice

# 가상 머신 확장 집합(VMSS, Virtual Machine Scale Sets)

부하 분산 장치로 가상머신이 실패해도 서비스 가용성 제공 가능함

하지만 \*백엔드 풀에 넣은 가상 머신의 수가 트래픽 처리하기 부족하다면??

➔ 일일이 동일한 크기와 환경의 가상 머신 배포를 수동으로 해야함

트래픽 처리량이 다시 감소된다면??

➔ 또 수동으로 중지하고 장치에서 제거해야함

가상 머신 수가 많아지면 너무 번거롭고 휴먼에러 발생 가능성 높아짐

백엔드 풀이란?

- 지정된 부하 분산 규칙에 대한 트래픽을 제공하는 리소스 그룹 정의  
(AWS의 대상그룹)



## Practice

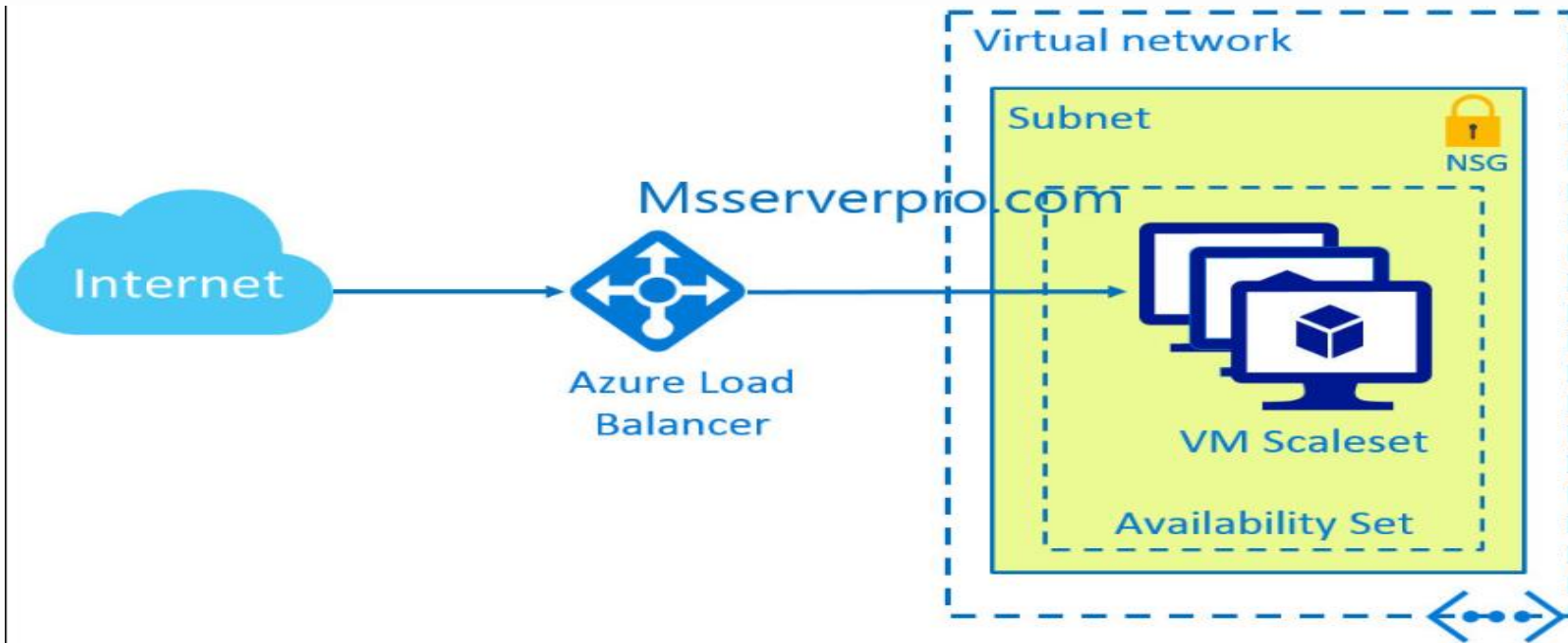
# 가상 머신 확장 집합(VMSS, Virtual Machine Scale Sets)

- 이와 같은 탄력성에 관한 우려를 해결하고자 사용하는 것이 VMSS
- VMSS는 VM 그룹을 만들어 자동/수동으로 크기를 조정하고 VM관리 및 업데이트 가능한 VM관리 그룹
- 대규모 컴퓨팅 환경이나 빅데이터, 컨테이너 워크로드를 처리하는 인프라를 빠르고 쉽게 구현

## Practice

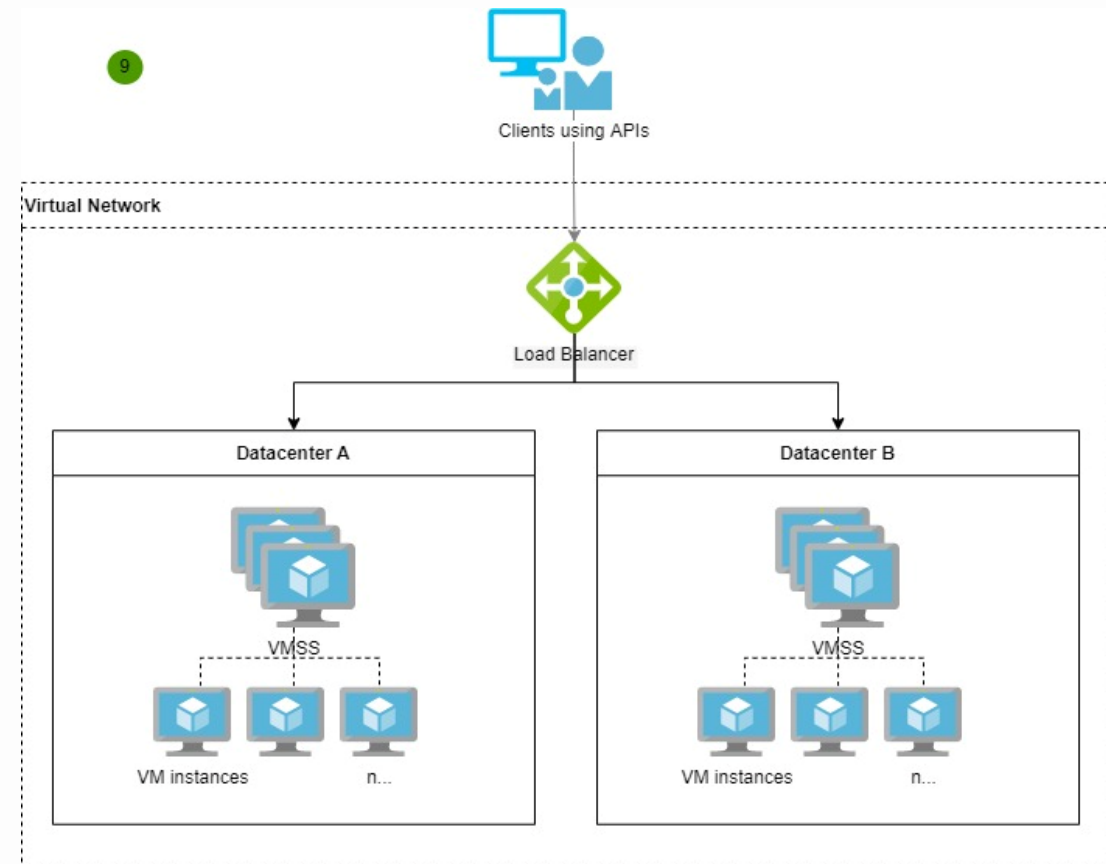
## 주요 이점

- 개별 VM을 관리하는 이전 방식과 비교해 확장 집합의 VM들은 하나의 단위로 관리됨
- 메트릭(성능 지표)를 기반으로 자동 크기 조정
- VMSS를 배포하는 동안 가상 네트워크와 Azure 부하 분산 장치를 한번에 배포 가능
- VMSS는 VM 인스턴스 증감에 따라 부하 분산 장치 구성이나 스크립트 확장, DevOps 연계 등의 작업을 자동으로 수행
- VMSS 서비스 자체에 추가요금x → 기본 컴퓨팅 리소스인 VM 인스턴스와 부하분산 장치, 관리되는 디스크 스토리지, 네트워크 아웃바운드 트래픽에 대한 비용만 지불!! (AWS autoscaling도 추가 비용 x)



## 주요 특징

- VMSS 는 가상네트워크의 단일 서브넷으로 배포(하나의 가상머신을 배포하는 것과 비슷)
- VMSS는 단위로 하나의 배치 그룹을 사용하고 하나의 배치그룹에는 최대 100개의 인스턴스 생성 가능
- 100개 이후 부터 추가 배치 그룹이 생성됨



## Practice

## VMSS 사용에 적절한 상황 4가지

- 서비스 환경의 일관성 유지하며 많은 VM을 실행하는 상황  
VMSS 사용시 기본OS이미지 부터 사용자 지정 이미지까지 구성 작업을 최소화하여 VM인스턴스 빠르게 배포가능
  - VM의 특정 인스턴스가 실패해도 다른 인스턴스를 통해 서비스를 계속 제공해야 하는 상황  
여러 데이터 센터에 VM 인스턴스를 자동으로 배포 가능
  - 서비스 요청 수요의 변동성에 대비해 VM인스턴스를 탄력적으로 운용해야하는 상황  
메트릭 기반으로 크기 조정을 자동으로 수행 또는 특정 날짜나 일정 주기로 크기 조정 가능
  - 대규모 컴퓨팅 작업을 위해 VM 인스턴스를 대량으로 만들어야 하는 상황  
최대 1000개의 VM 인스턴스 범위내 대량으로 신속히 배포 가능
- > 서비스 요청 수요의 변동성이 높고 이를 예측하기 쉽지 않은 경우!!



# VMSS 실습해보기

## 실전 연습 1

Sudo apt-get update

Sudo apt install stress -y

Stress -c 1





## Practice

## 가상 머신 확장 집합 설정 - 기본사항

- 최대 3개의 가용성 영역 지정
- 오케스트레이션 모드

유연한 - 동일한 유형의 가상머신 뿐아니라 서로 다른 유형의 가상 머신을 수동으로 추가 가능(IaaS VM API를 통한 Azure 관리 기능 지원)

균일성 - 동일한 유형의 가상 머신 인스턴스 배포(VMSS VM API를 통한 Azure 관리 기능 사용x)

## 프로젝트 정보

배포된 리소스와 비용을 관리할 구독을 선택합니다. 폴더 같은 리소스 그룹을 사용하여 모든 리소스를 정리 및 관리합니다.

구독 \*

리소스 그룹 \*  [새로 만들기](#)

## 확장 집합 세부 정보

가상 머신 확장 집합 이름 \*  ✓

지역 \*

가용성 영역 ①

## 오케스트레이션

확장 집합에는 가상 머신 인스턴스의 특성(크기, 데이터 디스크 수 등)을 정의하는 "확장 집합 모델"이 있습니다. 확장 집합의 인스턴스 수가 변경되면 확장 집합 모델에 따라 새 인스턴스가 추가됩니다. [확장 집합 모델에 대한 자세한 정보](#)

오케스트레이션 모드 \* ①

- ☒ **유연한:** 같은 가상 머신 유형이나 여러 가상 머신 유형을 사용하여 대규모로 고가용성 달성
- ☐ **균일성:** 같은 인스턴스가 포함된 대규모 상태 비저장 워크로드에 최적화됨

보안 유형 ①

1

인스턴스 정보

이미지 \* ①

2

[모든 이미지 보기](#) | [VM 생성 구성](#)

✓ 이 이미지는 추가 보안 기능과 호환됩니다. [신뢰할 수 있는 시작 보안 유형으로 바꾸려면 여기를 클릭하세요.](#)

VM 아키텍처 ①

☐ Arm64  
☒ x64

Azure Spot 할인으로 실행 ①

☐

크기 \* ①

[모든 크기 보기](#)

최대 절전 모드 사용(미리 보기) ①

☐

❗ 최대 절전 모드를 사용하려면 구독을 등록해야 합니다. [자세한 정보](#)

관리자 계정

인증 형식 ①

☒ 암호  
☐ SSH 공개 키

사용자 이름 \* ①

✓



## Practice

# 가상 머신 확장 집합 설정 - 디스크

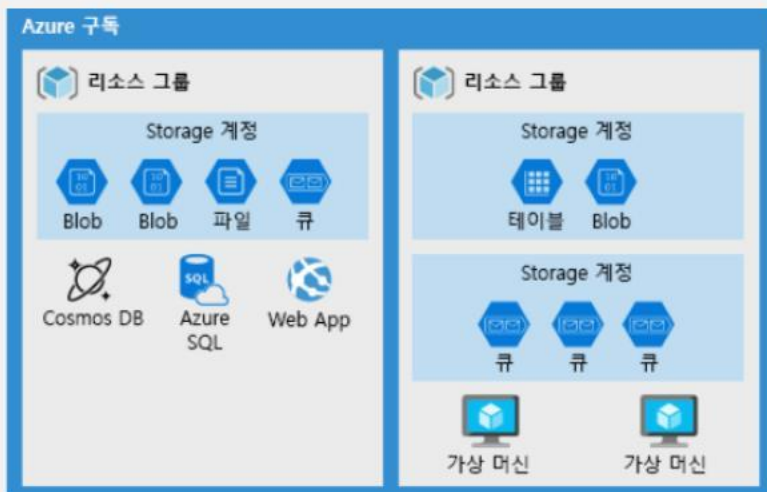
- OS 디스크 유형
- 로컬 중복 스토리지(프리미엄 SSD, 표준 SSD, 표준 HDD) or 영역 중복 스토리지(프리미엄 SSD, 표준 SSD)
- 관리
- 키 관리
  - 플랫폼 관리형 키(관리 디스크와 스냅샷, 이미지, 데이터 사용되지 않을때 자동 암호화)
  - 고객 관리형 키(Azure 자격 증명 모음-Key Vault에 있는 사용자의 키를 사용해 관리 디스크로 암호화 관리)
  - -> 이방식 선택시 플랫폼 관리형키로 변경불가
  - 플랫폼 관리형 키 및 고객 관리형 키
- 데이터 디스크
  - VMSS의 모든 VM 인스턴스에 추가할 데이터 디스크를 설정하거나 기존 디스크를 연결
  - VMSS를 배포한 후에도 추가 가능!!

## Storage Account

# Azure 스토리지는 스토리지 계정을 통해 관리합니다

- 하나의 스토리지 계정으로 여러 개의 Azure 스토리지 서비스를 사용할 수 있도록 인증 및 권한을 부여합니다.
- 예기치 못한 이벤트에 대해 데이터를 보호하는 중복 옵션도 스토리지 계정별로 관리됩니다.

## 스토리지 계정 다이어그램



## 사용 가능한 중복(복제) 옵션 종류

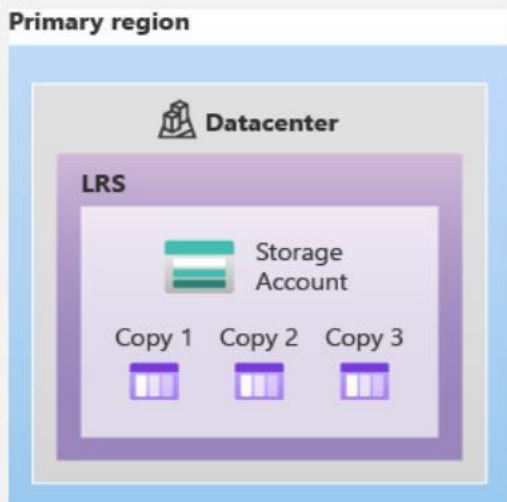
주 지역	LRS (로컬 중복 저장소)
	ZRS (영역 중복 저장소)
보조 지역	GRS (지역 중복 저장소)
	GZRS (지역 영역 중복 저장소)
	RA-GRS (보조 지역에서 읽기 권한이 주어진 지역 중복 저장소)
	RA-GZRS (보조 지역에서 읽기 권한이 주어진 지역 영역 중복 저장소)

## Redundancy Option

# LRS는 주 지역의 데이터 센터에 3개의 복제본을 생성하는 옵션입니다

- LRS(Locally Redundant Storage)는 소프트웨어 오류로부터 데이터를 보호하지만, 하드웨어에 발생한 재해에 대해서는 보호할 수 없습니다.
- 따라서 가장 저렴한 옵션이지만, 고가용성이나 높은 내구성이 필요한 경우엔 권장하지 않습니다.

## LRS 다이어그램



## 상황별 LRS 적용 예시

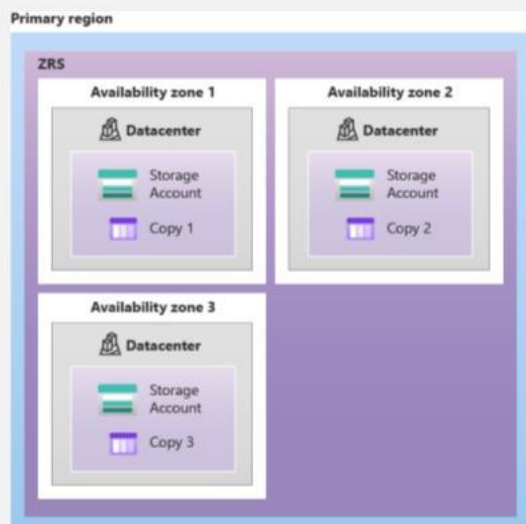
- 데이터 손실이 발생하더라도 쉽게 복구할 수 있는 데이터를 저장하는 경우
- 데이터의 특성 상 특정한 국가 또는 지역 내에서만 사용이 제한되는 경우
- 사용자가 직접 디스크의 복제 및 가용성을 설정하는 Azure 비관리 디스크를 사용하는 경우

## Redundancy Option

# ZRS는 주 지역의 가용성 영역에 3개의 복제본을 생성하는 옵션입니다

- ZRS(Zone Redundant Storage)는 한 가용성 영역에 물리적인 피해가 있더라도 다른 가용성 영역에서 작업이 가능합니다.
- 일시적으로 접근이 불가능 할 때 접근 요청이 들어오면, 지수 백오프와 같은 기법을 사용하여 동기적으로 요청을 처리 및 반환합니다.

## ZRS 다이어그램



## 상황별 ZRS 적용 예시

- 고가용성 및 높은 내구성이 필요한 데이터를 저장하는 경우
- 데이터의 특성 상 특정한 국가 또는 지역 내에서만 사용이 제한되는 경우

## Practice

## 가상 머신 확장 집합 설정 - 확장중

- VM 인스턴스를 수동으로 늘리거나 줄일 수 있음
- 성능 모니터링으로 자동으로 늘거나 줄일 수 있음
- 초기 인스턴스 수
  - VMSS를 배포할 때 초기에 실행할 VM 인스턴스 수
- 크기 조정 정책
  - 수동
  - 사용자 지정(매트릭 기반 자동)
    - 조정의 최소/최대 인스턴스 수와 CPU 임계값을 기반으로 스케일 아웃/스케일 인 설정 가능
    - VMSS 배포 후 다른 메모리나 디스크 관련 메트릭으로 크기 조정 정책 만듦
    - 예측 자동 크기 조정
      - 최소 7일간(15일 권장)의 CPU 사용 패턴 기록으로 기계 학습하여 VMSS에 대한 전체 CPU 부하 예측 이를 바탕으로 요청 처리에 필요할 것으로 예측되는 경우 적시에 스케일 아웃 일으킴
      - > 부하가 많아지고 스케일 아웃 및 인 되는데 시간이 걸리므로 이를 방지
  - 진단 로그
    - 자동 크기 조정을 사용하는 경우 진단 로그 수집 여부 설정 옵션
  - 축소 정책
    - 최신 VM, 가장 오래된 VM, 기본 VM
    - 기본 VM
      - 가용성 영역 간 가상 머신 균형
      - 장애 도메인 간 가상 머신 균형
      - 가장 높은 인스턴스 ID를 가진 가상 머신 제거



## Practice

# 가상 머신 확장 집합 설정 - 관리

- VM 인스턴스의 모니터링 관리 옵션
- 업그레이드 모드
- 게스트 OS모드



## Practice

# 가상 머신 확장 집합 설정 - 상태

- VM 인스턴스의 상태 모니터링
- 애플리케이션 상태 모니터링
  - 애플리케이션 상태 확장 : VM 인스턴스 내에 배포되어 VM 상태를 알려줌, 부하 분산 장치 프로브처럼 외부 프로브를 사용할 수 없는 경우에 사용
  - 부하 분산장치 프로브 : VMSS를 만들면서 부하 분산 장치를 설정한 경우에만 만들 수 있음
- 자동 복구 사용
  - 비정상 인스턴스를 자동으로 제거하고 최신 인스턴스 모델 설정으로 새 인스턴스를 만들 경우 사용
- 유예기간
  - VM 상태 변경을 한 후 잘못된 복구 시도가 일어날 것을 방지하기 위해 자동 복구 일시 중단 시간 설정





## Practice

# 가상 머신 확장 집합 설정 - 고급

### •할당 정책

- 100개 인스턴스가 넘는 크기 조정 사용
  - VMSS는 기본적으로 하나의 배치 그룹에 최대 100개까지 호용됨
  - 여러 배치 그룹을 사용하여 최대 1,000개의 인스턴스를 만들 수 있음
- 영역에서 균등 분산 강제 적용
  - 가용성 영역 2개 이상 선택시 표시됨
  - 각 영역의 VM 수가 다른 영역의 VM 수와 균등하게 배포되도록 강제할지 여부
  - 선택시 영역에서 균형유지를 못하면 확장 집합 크기 조정 실패
- 분산 알고리즘
  - 장애 도메인에서 VM 인스턴스 배포 균형을 결정
  - 최대 분산
    - 각 영역에서 장애 도메인에 최대 분산
    - 대부분의 워크로드 시나리오에서 권장함
    - 최대분산 선택시 100개 인스턴스가 넘는 크기 조정 사용도 디폴트로 적용됨
  - 고정 분산
    - 무조건 5개의 장애 도메인에 분산 시도
    - 5개 미만의 장애 도메인을 사용하면 VMSS 배포 실패



# VMSS 문제풀기

숙제입니다~

[지식 점검 - Training | Microsoft Learn](#)



## 참고

<https://buildgoodhabit.tistory.com/92>

(유연한 균일함 차이)

<https://learn.microsoft.com/ko-kr/azure/virtual-machine-scale-sets/virtual-machine-scale-sets-orchestration-modes>