

Elastic Block Store - practice

Sung-Dong Kim,
School of Computer Engineering,
Hansung University

Contents

- ▶ EBS volume 생성
- ▶ 볼륨을 EC2 인스턴스에 연결하고 탑재
- ▶ 사용자 볼륨의 스냅샷 생성
- ▶ 스냅샷에서 새 볼륨 생성
- ▶ 새 볼륨을 EC2 인스턴스에 연결하고 탑재

EBS Volume의 기능

- ▶ **영구 스토리지:** 볼륨 수명은 특정 Amazon EC2 인스턴스와 독립적
- ▶ **범용:** Amazon EBS 볼륨은 모든 운영 체제에서 사용할 수 있는 형식이 지정되지 않은 원시 블록 디바이스
- ▶ **고성능:** Amazon EBS 볼륨은 로컬 Amazon EC2 드라이브와 같거나 더 우수한 성능 제공
- ▶ **높은 안정성:** Amazon EBS 볼륨은 가용 영역 내에서 기본적으로 이중화 제공

EBS Volume의 기능

- ▶ **뛰어난 복원력:** Amazon EBS의 AFR(연간 실패율)은 0.1%에서 1% 사이
- ▶ **가변 크기:** 볼륨 크기는 1GB에서 16TB
- ▶ **사용 편의성:** Amazon EBS 볼륨을 쉽게 생성하고 연결, 백업, 복원 및 삭제할 수 있음



1. EBS 볼륨 생성

- ▶ EC2 → instance (새로 만든 instance, lab)
- ▶ 가용 영역 확인
- ▶ Elastic Block Store → 볼륨
- ▶ 볼륨 생성
 - ▶ type: general purpose SSD (gp2)
 - ▶ size (GB): 1
 - ▶ 가용 영역
 - ▶ Tag: Name = my volume

2. 볼륨을 인스턴스에 연결

- ▶ 볼륨에서 **my volume** 선택
- ▶ 작업 → 볼륨 연결
 - ▶ lab 인스턴스 선택
 - ▶ 디바이스 이름 확인: `/dev/sdf`
- ▶ EC2 instance에서 [스토리지] 탭 확인

3. 파일 시스템 생성 및 구성

- ▶ EC2 접속 후,
- ▶ `df -h`
 - ▶ `/dev/xvda1`: 8GB 디스크 볼륨
 - ▶ 새 볼륨은 보이지 않음
- ▶ 새 볼륨에 ext3 파일 시스템 생성: `sudo mkfs -t ext3 /dev/sdf`
- ▶ 새 스토리지 볼륨을 탑재할 디렉토리 생성: `sudo mkdir /mnt/data-store`

3. 파일 시스템 생성 및 구성

- ▶ 새 볼륨 탑재: `sudo mount /dev/sdf /mnt/data-store`
- ▶ 인스턴스가 시작될 때마다 이 볼륨을 탑재하도록 Linux 인스턴스 구성
 - ▶ `echo “/dev/sdf /mnt/data-store ext3 defaults,noatime 1 2” | sudo tee -a /etc/fstab`
- ▶ 구성의 마지막 행 설정 확인: `cat /etc/fstab`
- ▶ `df -h`

3. 파일 시스템 생성 및 구성

- ▶ 탑재된 볼륨에 파일 생성

- ▶ `sudo sh -c "echo some text has been written > /mnt/data-store/file.txt"`

4. EBS 스냅샷 생성

- ▶ 볼륨 → **my volume** 선택
- ▶ 작업 → 스냅샷 생성
 - ▶ Tag: Name = **my snapshot**
- ▶ 스냅샷 → 확인
- ▶ file.txt 삭제: `sudo rm /mnt/data-store/file.txt`

5. EBS 스냅샷 복원

- ▶ 스냅샷을 이용한 볼륨 생성
- ▶ 스냅샷 → **my snapshot** 선택
- ▶ 작업 → 볼륨 생성
 - ▶ 동일 가용 영역
 - ▶ Tag: Name = **restored volume**
- ▶ 볼륨 선택 → 작업 → 볼륨 연결
 - ▶ lab 인스턴스
 - ▶ 디바이스 이름: `/dev/sdg`

5. EBS 스냅샷 복원

- ▶ 복원된 볼륨 탑재
 - ▶ 디렉토리 생성: `sudo mkdir /mnt/data-store2`
 - ▶ 볼륨 탑재: `sudo mount /dev/sdg /mnt/data-store2`
 - ▶ 확인: `ls /mnt/data-store2`