



# AWS 기반 컴퓨팅

# 학습 내용

## 강의의 핵심

**배울 내용은 다음과 같습니다.**

- Amazon Elastic Compute Cloud(Amazon EC2) 가상화 설명하기
- EC2 인스턴스를 시작할 때 지정해야 하는 네트워킹 구성 요소 이해하기
- Amazon EC2 사용자 데이터 및 메타데이터 설명하기
- EC2 인스턴스 생성하기

**주요 용어:**

- EC2 인스턴스
- 하이퍼바이저 계층
- 인스턴스 유형
- Amazon Machine Images(AMI)





# AWS 기반 컴퓨팅

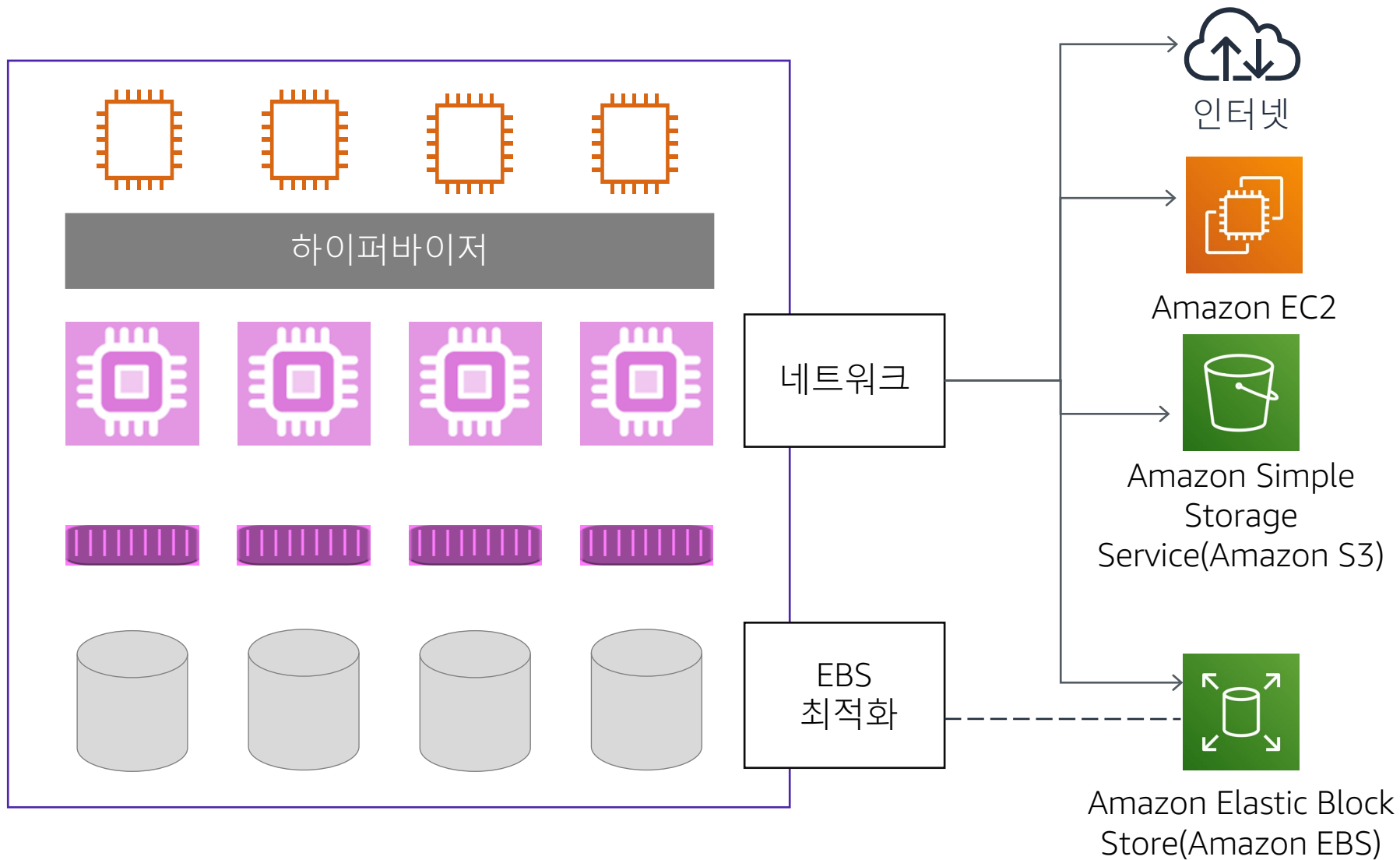
# Amazon EC2 가상화

Amazon EC2 인스턴스

CPU

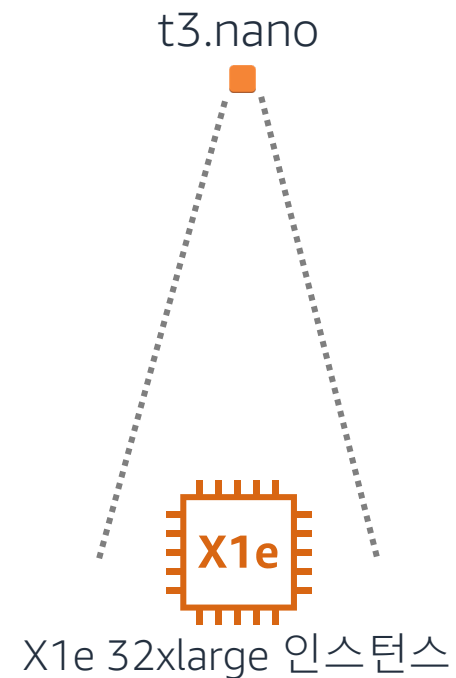
RAM

인스턴스 스토어(임시)

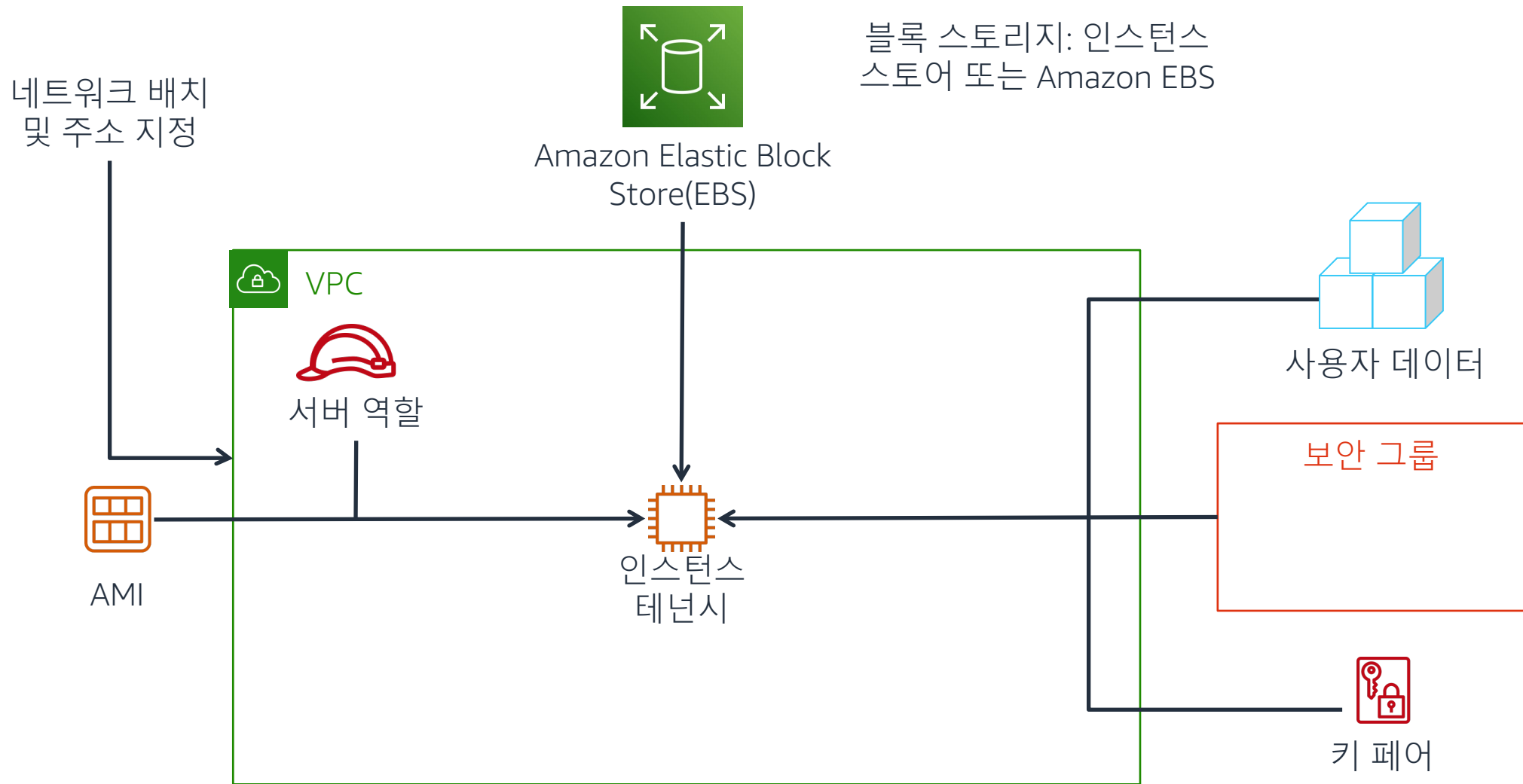


# 인스턴스 유형

변형		
CPU 유형 스토리지 유형	CPU 또는 코어 수 스토리지 양	메모리 양 네트워크 성능
범주		
범용 가속 컴퓨팅	컴퓨팅 최적화 스토리지 최적화	메모리 최적화
패밀리		
범주 내 유형 모음		
유형		
가상 하드웨어 기능 고유 선택		



# 인스턴스 프로비저닝

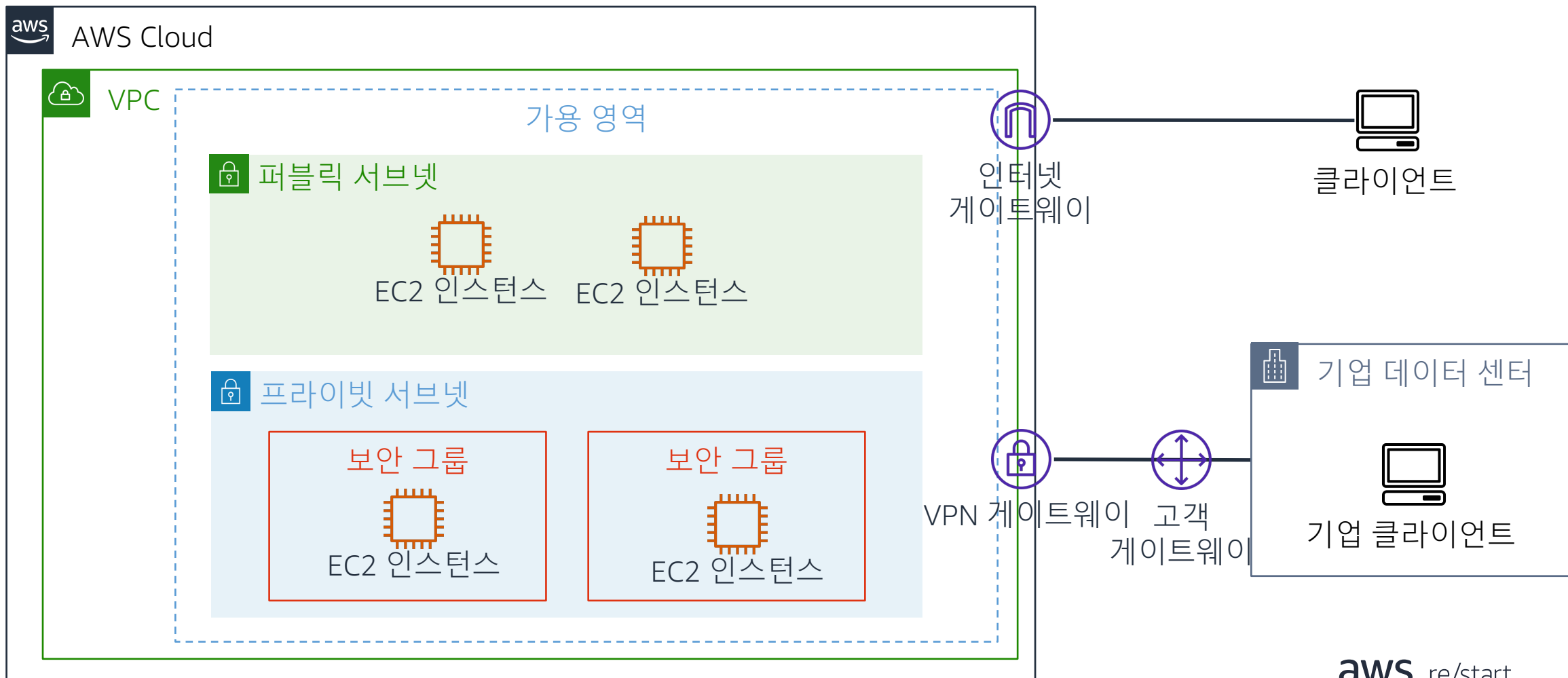


# Amazon Machine Image(AMI)



# Amazon EC2 네트워킹

## Virtual Private Cloud(VPC)

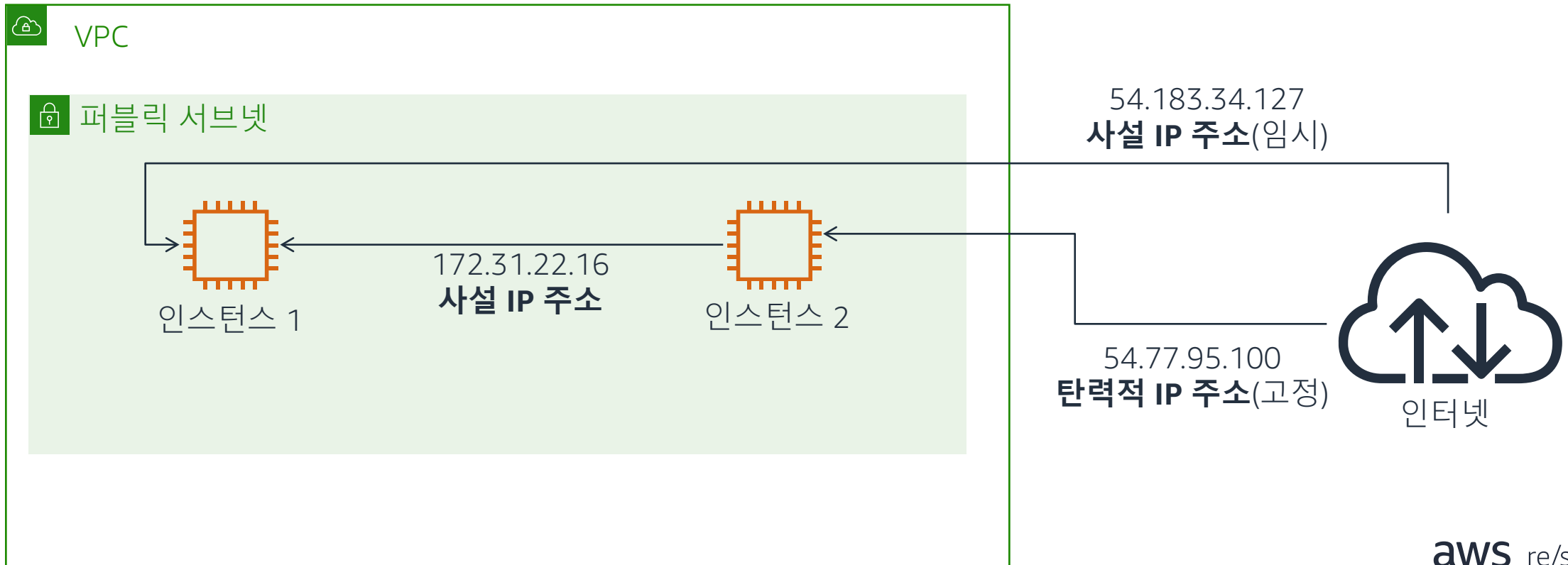




# IP 주소 유형

## 사설, 공인, 탄력적 IP 주소

- 공인 IP 주소는 인스턴스를 중지했다 다시 시작하면 변경됩니다.
- 고정 공인 IP 주소에는 탄력적 IP 주소를 사용합니다.



# 보안 그룹

- 인스턴스 액세스 제한 기준:
  - 포트 범위
  - IP 주소 범위
  - 리소스 ID
- 인스턴스는 여러 보안 그룹과 연결 가능
- 인바운드 및 아웃바운드 데이터 허용
- 시작 후 추가/수정 가능

보안 그룹 설정 예제:

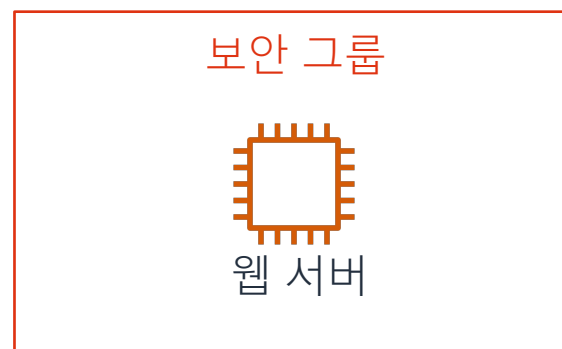
포트 22에 대한 원격 액세스



포트 3306



포트 22 및 웹 트래픽 80에  
대한 원격 액세스



# 보안 그룹: 예제

어디서나 HTTP 포트 액세스 허용:

ID	포트 범위	소스
sg-dfc83cba	80(HTTP)	0.0.0.0/0

특정 컴퓨터에서 Secure Shell(SSH) 액세스 허용:

ID	포트 범위	소스
sg-4ad3712f	22(SSH)	10.50.2.133/32

보안 그룹 멤버의 SSH 액세스 허용:

ID	포트 범위	소스
sg-d1cd6fb4	22(SSH)	sg-4ad3712f

# 인스턴스 프로파일

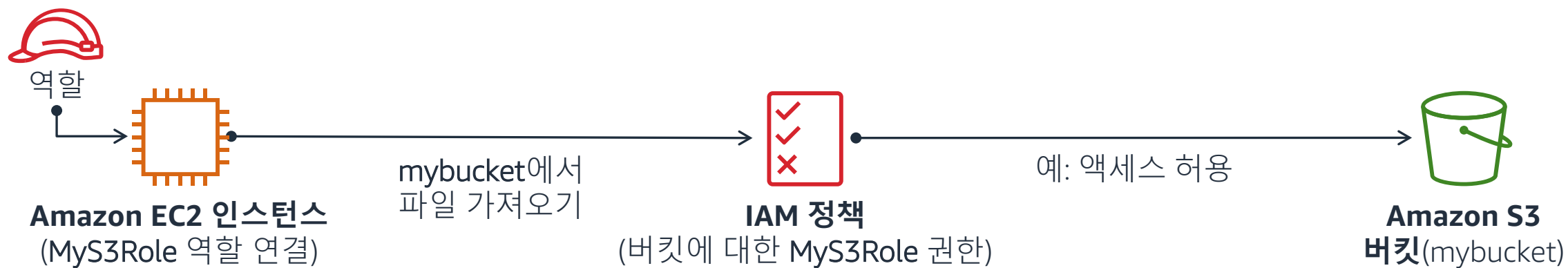
- Amazon EC2에 액세스 키와 비밀 키 로컬로 저장 방지
- AWS Identity 및 Access Management(IAM) 역할을 EC2 인스턴스에 연결할 수 있도록 지원
- 액세스 키를 인스턴스에 자동 전파
  - 하루에 여러 번 액세스 키 자동 교체
  - 여러 인스턴스에서 사용(예: Auto Scaling 그룹)

예제: 애플리케이션 데이터가 저장된 Amazon Simple Storage Service(Amazon S3) 버킷에 액세스할 수 있는 권한을 EC2 인스턴스에 부여합니다.

**모범 실무**

보안 인증 정보가 손상될 수 있으므로 EC2 인스턴스에 영구 액세스 키를 저장하지 마십시오.

# 인스턴스 프로파일 예제



# 사용자 데이터

## 시작 시 실행되는 스크립트:

- Linux 스크립트
- Microsoft Windows 배치 또는 PowerShell 스크립트

## 스크립트 실행 주체:

- Linux의 Cloud-init
- Microsoft Windows의 EC2Launch 서비스

# Linux의 사용자 데이터

## Linux 쉘 스크립트

```
#!/bin/bash
yum update -y
yum groupinstall -y
    "Web Server" "PHP Support"
yum install -y php-mysql
service httpd start
chkconfig httpd on
```

## #cloud-config

```
cloud-config
repo_update: true
repo_upgrade: all
```

```
packages:
    - httpd
    - php
    - php-mysql
```

```
runcmd:
    - chkconfig httpd on
```

# Microsoft Windows의 사용자 데이터

## 배치 파일

```
<script>
winrm quickconfig -q

winrm set winrm/config/winrs
@{MaxMemoryPerShellMB="300"}

winrm set winrm/config
@{MaxTimeoutms="1800000"}
</script>
```

## PowerShell 스크립트

```
<powershell>
# Install IIS and web
# Management Tools.

Import-Module ServerManager

Install-WindowsFeature
    web-server, web-webserver
Install-WindowsFeature
    web-mgmt-tools
</powershell>
```



# EC2 인스턴스 메타데이터

- 실행 중인 인스턴스를 구성 또는 관리하는 데 사용할 수 있는 인스턴스 관련 데이터
- IP 주소 169.254.169.254에서 모든 인스턴스에 제공되는 메타데이터 서비스
- 인스턴스 메타데이터 쿼리

```
export INSTANCE_ID=$(curl http://169.254.169.254/latest/meta-data/instance-id)
```

```
aws ec2 describe-instances --instance-id $INSTANCE_ID
```

**Linux**

```
$INSTANCE_ID=(new-object net.webclient).DownloadString(  
    'http://169.254.159.254/latest/meta-data/instance-id')
```

```
aws ec2 describe-instances --instance-id $INSTANCE_ID
```

**Microsoft Windows**

# 사용자 데이터 및 Amazon EC2 인스턴스 메타데이터

사용자 데이터에서 인스턴스 속성을 참조하는 것이 유용할 수 있습니다.  
인스턴스 메타데이터를 쿼리하고 시작 시 사용자 데이터 스크립트에서 사용합니다.

Amazon Linux AMI의 호스트 이름을 시작 시 instance-id로 변경하는 방법:

## 사용자 데이터 스크립트

```
#!/bin/bash
```

```
#Get hostname from instance-id metadata = $newhost  
newhost=$(curl http://160.254.169.254/latest/meta-data/instance-id/)
```

```
#change hostname in /etc/hosts & /etc/sysconfig/network  
sudo sed -i "s/\<localhost\>/$newhost/g" /etc/hosts  
sudo sed -i "s/\<localhost\>/$newhost/g" /etc/sysconfig/network
```

```
#Reboot instance  
sudo reboot
```

Linux

# EC2 인스턴스 사용자 데이터 검색

## EC2 인스턴스 메타데이터 예제:

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/
```

```
ami-id
```

```
ami-launch-index
```

```
ami-manifest-path
```

```
block-device-mapping/
```

```
events/
```

```
hostname
```

```
iam/
```

```
instance-action
```

# EC2 인스턴스 사용자 데이터

## 예제(결과 포함):

```
[ec2-user ~]$ curl http://169.254.169.254/latest/user-data  
#!/bin/bash  
yum update -y  
service httpd start  
chkconfig httpd on
```

# 추가 메타데이터 예제

퍼블릭 호스트 이름 검색 예제:

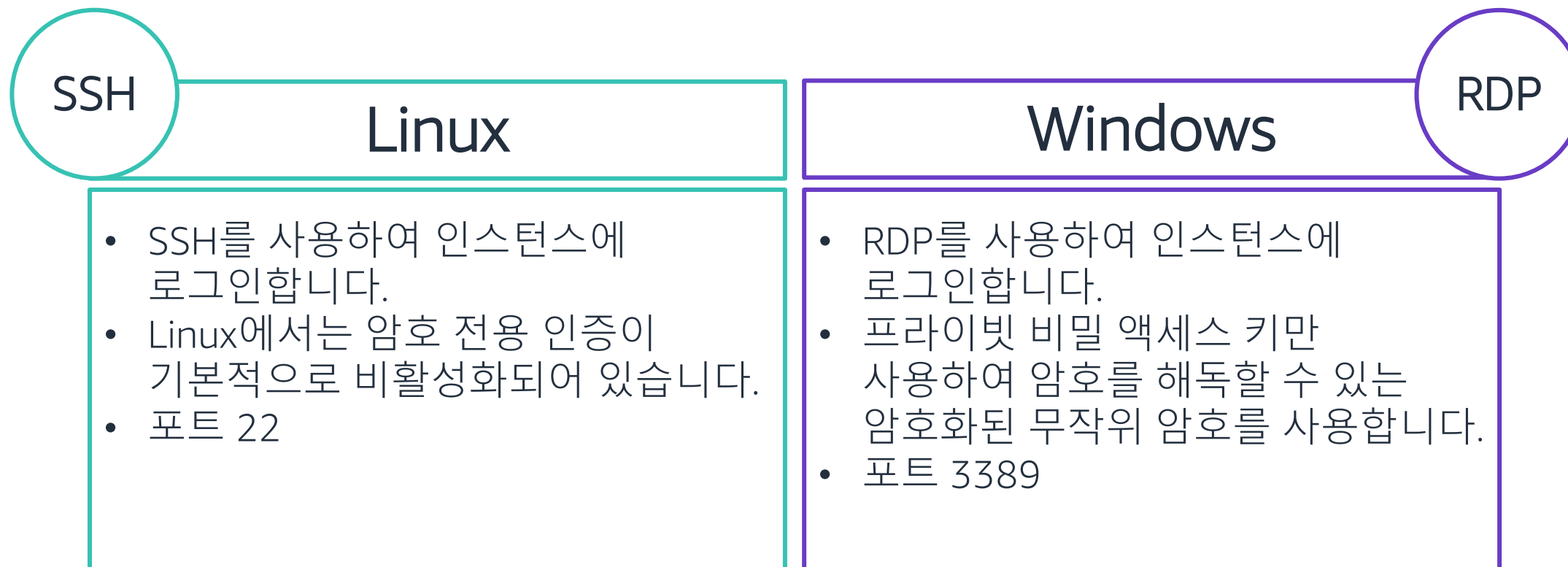
```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-hostname  
ec2-203-0-113-25.compute-1.amazonaws.com
```

인스턴스에 대한 서브넷 ID 검색 예제:

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-  
data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id  
subnet-be9b61d7
```

# 키 페어

## 원격 액세스



# 자체 인스턴스 보안 설정

## 인스턴스를 생성한 후 사용자 및 권한 변경

- Microsoft Windows
  - AWS Directory Service와 함께 사용
- Linux
  - Kerberos 기반 통합 인증(SSO) 사용
  - ssh-keygen을 사용하는 사용자를 위한 퍼블릭 키와 프라이빗 키 페어 생성
  - 사용자의 퍼블릭 키를 `~/.ssh/authorized_keys`에 추가

## 플릿 전체 사용자 관리를 위한 전략 마련:

- AMI 생성
- 구성 소프트웨어 사용

# 구성 관리 애플리케이션

구성 관리 애플리케이션은 다음을 할 수 있습니다.

- 많은 수의 시스템에서 액세스 권한 부여 및 취소 프로세스 간소화
- 몇 개의 명령으로 프로세스 단축

구성 관리 애플리케이션:

- Chef
- Puppet
- Ansible



# 시작 후 구성 옵션

인스턴스 스크린 캡처  
가져오기

종료 방지 기능 활성화

인스턴스가 라우터 또는  
방화벽으로서 네트워크  
주소 변환(NAT) 기능을  
수행하는 경우 소스 및 대상  
확인 비활성화(기본적으로  
활성화됨)

# AWS CLI에서 Amazon EC2 시작

```
aws ec2 run-instances
--image-id ami-0123456789012345
--instance-type t2.micro
--key-name mykeypair
--security-group-ids sg-0123456789012345
--subnet-id subnet-0123456789012345
--iam-instance-profile Name=EC2Admin
--user-data file:///UserData.txt
--tag-specifications
'ResourceType=instance,Tags=[{Key=Name,Value=WebServer}] '
```

# 학습 내용 확인

Amazon Machine Image(AMI)란 무엇입니까?

사용자 데이터 필드의 목적은 무엇입니까?

새 EC2 인스턴스가 구성되고 성공적으로 시작됩니다. 인스턴스는 공인 IP 주소도 가지고 있음을 보여줍니다. 그러나 SSH로 연결하면 작업 시간이 초과되었음을 나타내는 오류가 발생합니다. 어떤 문제가 있을 수 있습니까?

인스턴스에서 실행 중인 스크립트가 메타데이터에 액세스하는 데 사용되는 IP 주소는 무엇입니까?

# 핵심 사항



- EC2 인스턴스 프로파일에 영구 액세스 키를 저장하지 않습니다.
- 애플리케이션 요구 사항에 가장 적합한 인스턴스 유형을 선택합니다. 필요한 경우 나중에 인스턴스 유형을 변경합니다.
- EC2 인스턴스 메타데이터는 프로그래밍 방식으로 액세스하여 인스턴스를 구성하거나 관리할 수 있습니다.