
AWS(Amazon Web Service)



2015040017 정택민
2017035208 곽나경



INDEX

▲ 1 클라우드 컴퓨팅이란?

▲ 2 AWS 소개

▲ 3 AWS 특징

▲ 4 AWS 시작하기

▲ 5 AWS 에 docker 환경 구축하기

1.1 클라우드 컴퓨팅의 정의



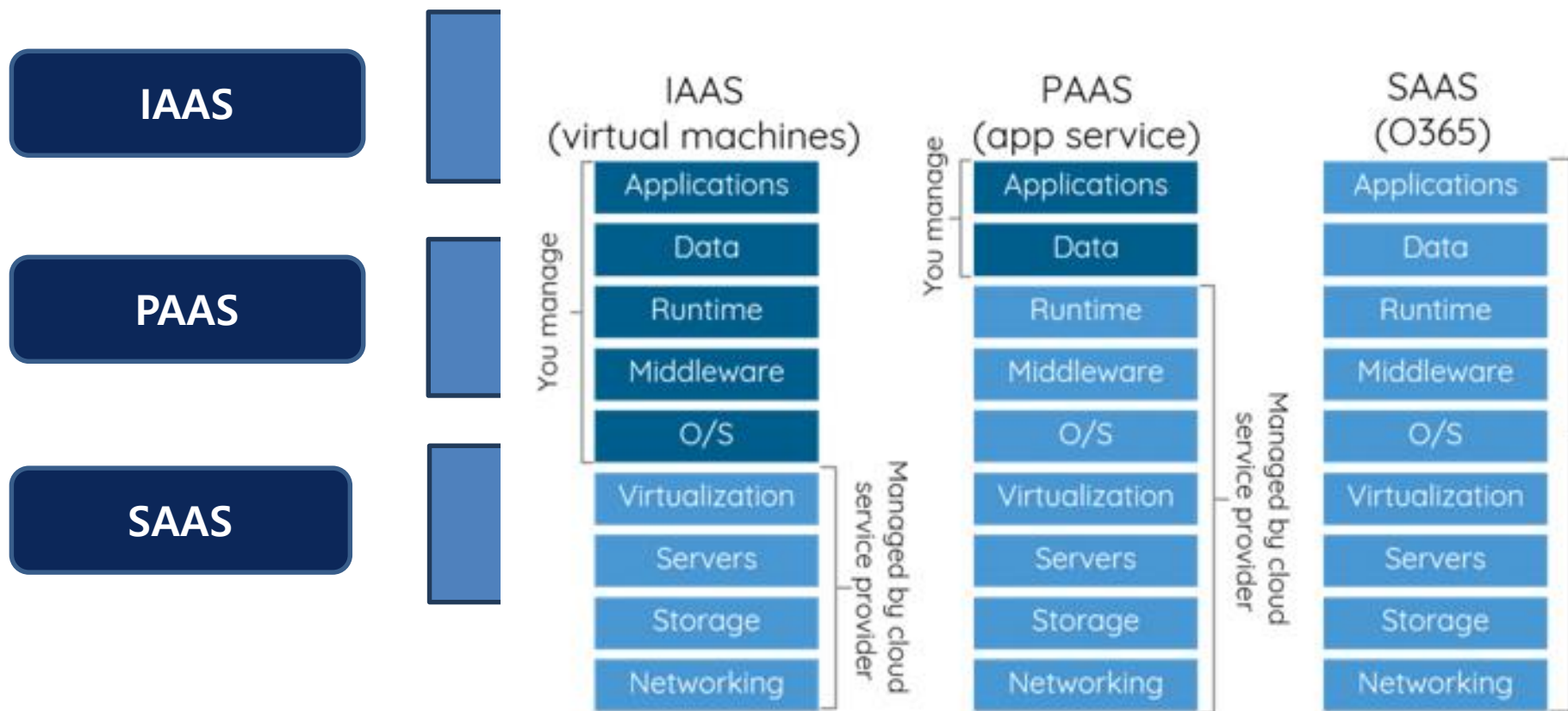
클라우드 컴퓨팅

정보가 인터넷 상의 서버에 영구적으로 저장되고, 클라이언트에는 일시적으로 보관되는 컴퓨터 환경

1.2 클라우드 컴퓨팅의 등장 배경



1.3 클라우드 컴퓨팅의 유형



1.4 클라우드 컴퓨팅의 이점

초기 선투자 비용 없음

고정비용을 가변비용으로 대체
미리 서버를 구매할 필요 없음



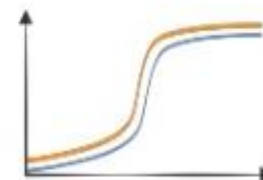
운영 비용 절감

사용한 만큼만 지불하며 규모의
경제로 인한 지속적인 비용



탄력적인 운영 및 확장

필요 용량에 대한 예측 불필요
수요에 맞춘 유연한 확장



속도 및 민첩성

수 분 만에 인프라 구축 가능
빠르게 변화에 대응



비즈니스에만 집중 가능

혁신을 위한 다양한 실험 가능
불필요한 인프라 관리 업무
제거



글로벌 확장

빠른 시간 내 글로벌 서비스
구현 가능



2.1 AWS 소개



Amazon Web Service

Amazon에서 제공하는 클라우드 서비스로, 서버, 스토리지, 네트워크 인프라 등 다양한 서비스를 제공

2.2 AWS 글로벌 인프라맵

전 세계 22개의 지리적 리전 내에 69개의 가용 영역을 운영



3.1 AWS의 이점

- 국제적 입지 : 클라우드 업계 1위, 100만명 이상의 사용자
- 가용성 : 클라우드 공급업체 중 가장 뛰어난 네트워크 가용성
- 성능 : 사용자에게 따라 필요한 서버를 몇 분 안에 증설 가능
- 보안 : Public Cloud의 약점인 보안 강화

3.2 AWS 이용 기업들

엔터프라이즈



게임사



스타트업



3.3 AWS 보안 공유 모델



3.4 AWS 서비스 포트폴리오





충북대학교
SW중심대학사업단

4 AWS 시작하기 (회원가입)

12개월 프리 티어 액세스 포함 AWS 계정

Amazon EC2, Amazon S3 및 Amazon DynamoDB 사용 포함
제안 약관 전문은 aws.amazon.com/free 참조

AWS 계정 생성

이메일 주소

jtm0609@naver.com

암호

●●●●●●●●

암호 확인

●●●●●●●●

AWS 계정 이름 ⓘ

TakminJeong

- 수집하는 개인정보의 항목: 이름, 사용자 아이디, 인증 및 보안 자격 증명, 전화 번호, 주소, 이메일 주소, IP주소, 쿠키, 브라우저와 기기의 식별자, 결제 수단 정보(신용카드 정보 또는 은행계좌 정보), 납세자 등록 번호

- 개인정보의 수집·이용 목적: AWS 계정의 생성 및 관리, AWS 서비스의 제공, AWS 서비스의 개선, 대금 결제, 고객의 AWS 계정 또는 서비스에 관한 의사소통 및 고객 요청 대응

- 개인정보의 보유·이용 기간: 고객의 개인정보는 AWS와의 비즈니스 관계가 유지되는 동안 그리고, AWS의 법적 의무, 분쟁 해결 및 계약 이행에 필요한 기간 동안만 보유됩니다. 또한, AWS는 AWS의 데이터 보존 정책 및 관련 법률에 따라 고객 개인 정보를 삭제합니다.

'(필수) 동의하고 계정 만들기' 버튼을 클릭하면 위의 개인 정보 수집 및 이용에 대한 조건을 읽고 동의하는 것을 의미합니다.

(필수) 동의하고 계정 만들기

4 AWS 시작하기 (회원가입)

계정 유형을 선택하고 아래 연락처 세부 정보 필드를 입력하십시오.

계정 유형 ⓘ

☐ 프로페셔널 ☒ 개인

이름

TakminJeong

전화번호

+821012345678

국가/리전

대한민국

주소

28650

68, Yeche-ro

구/군/시

Seowon-gu, Chungcheongbuk-do

시/도 또는 리전

Cheongju-si,

우편 번호

28650

☐ (필수) 여기를 클릭하면 [AWS 고객 동의](#)의 조건을 읽고 동의하는 것을 의미합니다.

☐ (선택) AWS서비스의 프로모션 알림 메일 수신

계정을 만들고 계속 진행

4 AWS 시작하기 (회원가입)

결제 정보

AWS 프리 티어 한도를 초과한 사용량에 대해서만, 결제 정보를 이용하여 귀하의 신원을 확인합니다. AWS 프리 티어 한도를 초과하지 않은 사용량에 대해서는 요금이 발생하지 않습니다. 자세한 내용은 [FAQ](#)를 참조하십시오.



결제 정보를 제출하면 신용카드가 유효한지를 확인하기 위해 귀하의 신용카드에 1 USD/EUR의 확인 요금을 청구합니다. 이 금액은 확인이 완료될 때까지 3~4일간 신용카드 내역서에 대기 중인 항목으로 표시될 수 있습니다. 확인이 완료되면 해당 청구 내역이 삭제됩니다. 확인 요금을 승인하도 록 은행 웹 사이트로 리디렉션될 수 있습니다.

신용/직불 카드 번호

카드 만료일

03 ▼ 2020 ▼

카드 소유자 이름

청구지 주소

☒ 내 연락처 주소 사용

28650 68, Yeche-ro
Seowon-gu, Chungcheongbuk-do
Cheongju-si, 28650
KR

☐ 새 주소 사용

검증 및 추가

자격 증명 확인

AWS 계정을 사용하려면 먼저 전화 번호를 확인해야 합니다. 계속하면 AWS 자동 시스템이 확인 코드를 사용하여 사용자에게 연락합니다.

확인 코드를 어떻게 보내 드릴까요?

☒ 문자 메시지(SMS) ☐ 음성 통화

국가 또는 리전 코드

대한민국 (+82) ▼

휴대전화 번호

* 전화 번호는 필수 필드입니다.

보안 검사

y4w8fn

위에 보이는 문자를 입력하십시오.

SMS 전송



4 AWS 시작하기 (회원가입)

지원 플랜 선택

AWS는 귀하의 요구를 충족할 수 있는 선별된 지원 플랜을 제공합니다. 귀하의 AWS 사용량에 맞는 지원 플랜을 선택하십시오. [자세히 알아보기](#)



기본 플랜

무료

모든 계정에 포함됨

- 포럼 및 리소스에 대한 상시 셀프 서비스 액세스
- 보안 및 성능 개선을 돕는 모범 사례 확인
- 상태 확인 및 알림에 대한 액세스



개발자 플랜

월 \$29부터

- 이온 채택, 테스트 및 개발을 위해 사용됨
- 업무 시간 중 AWS Support로의 이메일 액세스
- 하나의 기본 연락처로 무제한의 지원 사례를 열 수 있음
- 비 프로덕션 시스템에 대한 12시간 이내 응답



비즈니스 플랜

월 \$100부터

- 프로덕션 워크로드 및 비즈니스 크리티컬 의존성에 사용됨
- 채팅, 전화 및 이메일을 통한 AWS Support로의 상시 액세스
- 무제한의 연락처로 무제한의 지원 사례를 열 수 있음
- 프로덕션 시스템에 대한 1시간 이내 응답

엔터프라이즈 수준의 지원이 필요하십니까?

AWS에서 비즈니스와 미션 크리티컬 워크로드 실행(월 15,000 USD부터 시작됨)에 관한 추가 정보는 계정 관리자에게 문의하십시오. [자세히 알아보기](#)

4 AWS 시작하기 (로그인)

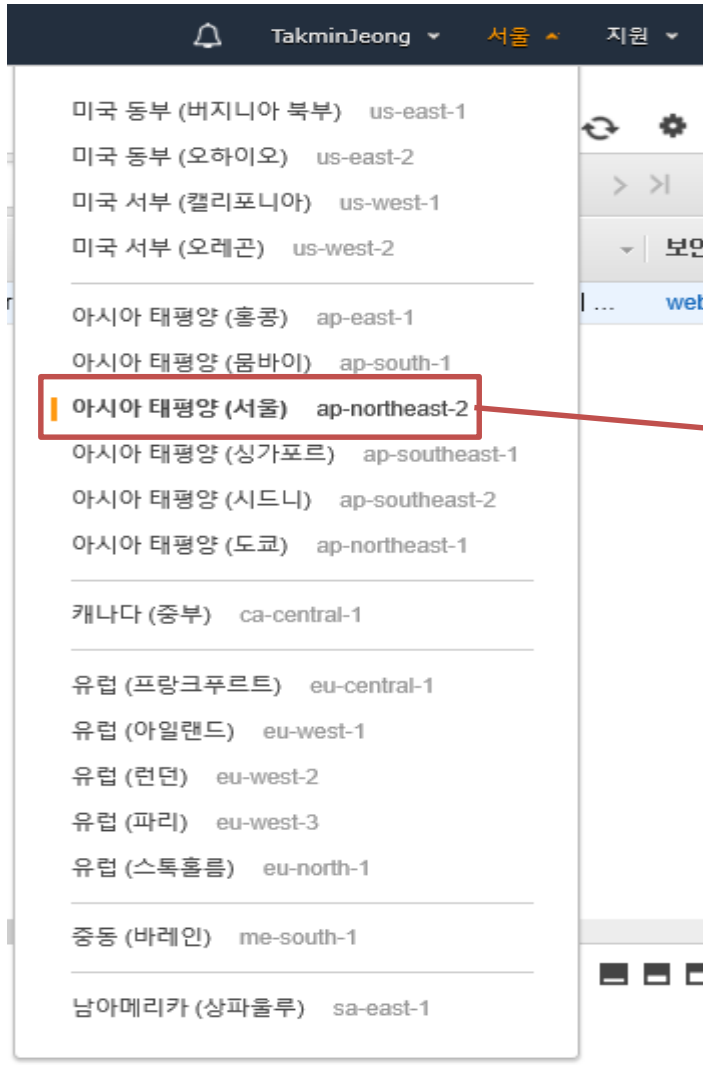
Amazon Web Services에 오신 것을 환영합니다.

Amazon Web Services 계정을 만들어 주셔서 감사합니다. 지금 계정을 활성화하고 있으며 몇 분 이내에 완료됩니다. 활성화가 완료되면 이메일을 받게 됩니다.

콘솔에 로그인


영업 팀 문의

4 EC2 사용하기




리전 선택(서울)

4 EC2 사용하기




서비스


리소스 그룹


TakminJeong서울


내역
EC2
콘솔 홈
Billing


이름 또는 기능(예: EC2, S3 또는 VM, 스토리지)으로 서비스를 찾습니다.


**컴퓨팅**
EC2
Lightsail
Lambda
Batch
Elastic Beanstalk
Serverless Application Repository
AWS Outposts
EC2 Image Builder


**스토리지**
S3
EFS
FSx
S3 Glacier
Storage Gateway
AWS Backup


**블록체인**
Amazon Managed Blockchain


**위성**
Ground Station


**Quantum Technologies**
Amazon Braket

**관리 및 거버넌스**
AWS Organizations
CloudWatch
AWS Auto Scaling
CloudFormation
CloudTrail
Config
OpsWorks

**분석**
Athena
EMR
CloudSearch
Elasticsearch Service
Kinesis
QuickSight
Data Pipeline
AWS Data Exchange
AWS Glue
AWS Lake Formation
MSK

**보안, 자격 증명 및 규정 준수**
IAM
Resource Access Manager
Cognito
Secrets Manager

**최종 사용자 컴퓨팅**
WorkSpaces
AppStream 2.0
WorkDocs
WorkLink

**사물 인터넷**
IoT Core
FreeRTOS
IoT 1-Click
IoT Analytics
IoT Device Defender
IoT Device Management
IoT Events
IoT Greengrass
IoT SiteWise
IoT Things Graph

4 EC2 사용하기

아시아 태평양 (서울) 리전에서 다음 Amazon EC2 리소스를 사용하고 있음:

실행 중인 인스턴스	1	탄력적 IP	0
전용 호스트	0	스냅샷	0
볼륨	1	로드 밸런서	0
키 페어	1	보안 그룹	2
배치 그룹	0		

③ AWS Launch Wizard for SQL Server를 사용하여 AWS에서 Microsoft SQL Server Always On 가용성 그룹을 손쉽게 크기 조정, 구성 및 배포할 수 있습니다. [자세히 알아보기](#)

인스턴스 시작

시작하려면 클라우드의 가상 서버인 Amazon EC2 인스턴스를 시작하십시오.

인스턴스 시작 ▼

참고: 인스턴스는 아시아 태평양 (서울) 리전에서 시작됩니다.

서비스 상태



서비스 상태 대시보드

리전

아시아 태평양 (서울)

상태

이 서비스가 정상적으로 작동 중입니다.

4 EC2 인스턴스 생성(1)

단계1: 가상머신 선택(AMI, Amazon Machine Image)



Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0db78afd3d150fc18

Amazon Linux

프리 티어 사용 가능

Amazon Linux 2는 5년간 지원을 제공합니다. Amazon EC2에 성능 최적화된 Linux kernel 4.14와 systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, 최신 소프트웨어 패키지를 추가적으로 제공합니다.

루트 디바이스 유형: ebs 가상화 유형: hvm ENA 활성화: 예

선택

64비트(x86)



Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-0e5ec6ec0e7672e12

Amazon Linux

프리 티어 사용 가능

Amazon Linux AMI는 EBS 기반의 AWS 지원 이미지입니다. 기본 이미지에는 AWS 명령줄 도구, Python, Ruby, Perl 및 Java가 있습니다. 리포지토리에는 Docker, PHP, MySQL, PostgreSQL 및 기타 패키지가 포함됩니다.

루트 디바이스 유형: ebs 가상화 유형: hvm ENA 활성화: 예

선택

64비트(x86)



Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-0cd7b0de75f5a35d1

프리 티어 사용 가능

Ubuntu Server 18.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

루트 디바이스 유형: ebs 가상화 유형: hvm ENA 활성화: 예

선택

64비트(x86)



Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-0c5a717974f63b04c

프리 티어 사용 가능

Ubuntu Server 16.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

루트 디바이스 유형: ebs 가상화 유형: hvm ENA 활성화: 예

선택

64비트(x86)



SUSE Linux Enterprise Server 15 SP1 (HVM) SSD Volume Type - ami-07d0b6ce73229c4b7

선택

4 EC2 인스턴스 생성(2)

단계2: 인스턴스 타입 선택

Amazon EC2는 각 사용 사례에 맞게 최적화된 다양한 인스턴스 유형을 제공합니다. 인스턴스는 애플리케이션을 실행할 수 있는 가상 서버입니다. 이러한 인스턴스에는 CPU, 메모리, 스토리지 및 네트워킹 용량의 다양한 조합이 있으며, 애플리케이션에 사용할 적절한 리소스 조합을 유연하게 선택할 수 있습니다. 인스턴스 유형과 이 인스턴스 유형이 컴퓨팅 요건을 충족하는 방식에 대해 [자세히 알아보기](#).

필터링 기준: 모든 인스턴스 유형 현재 세대 열 표시/숨기기

현재 선택된 항목: t2.micro (Variable ECU, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB 메모리, EBS 전용)

	그룹	유형	vCPUs ⓘ	메모리 (GiB)	인스턴스 스토리지 (GB) ⓘ	EBS 최적화 사용 가능 ⓘ	네트워크 성능 ⓘ	IPv6 지원 ⓘ
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS 전용	-	낮음에서 중간	예
<input checked="" type="checkbox"/>	General purpose	t2.micro 프리 티어 사용 가능	1	1	EBS 전용	-	낮음에서 중간	예
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS 전용	-	낮음에서 중간	예
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS 전용	-	낮음에서 중간	예
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS 전용	-	낮음에서 중간	예
<input type="checkbox"/>	General purpose	t2.xlarge	4	16	EBS 전용	-	보통	예
<input type="checkbox"/>	General purpose	t2.2xlarge	8	32	EBS 전용	-	보통	예

4 EC2 인스턴스 생성(3)

단계3: 인스턴스 세부 환경 설정

인스턴스 개수 ⓘ	<input type="text" value="1"/>	Auto Scaling 그룹 시작 ⓘ
구매 옵션 ⓘ	<input type="checkbox"/> 스팟 인스턴스 요청	
네트워크 ⓘ	<input type="text" value="vpc-e807d883 (기본값)"/>	↻ 새 VPC 생성
서브넷 ⓘ	<input type="text" value="기본 설정 없음(가용 영역의 기본 서브넷)"/>	↻ 새 서브넷 생성
퍼블릭 IP 자동 할당 ⓘ	<input type="text" value="서브넷 사용 설정(활성화)"/>	
배치 그룹 ⓘ	<input type="checkbox"/> 배치 그룹에 인스턴스 추가	
용량 예약 ⓘ	<input type="text" value="열기"/>	↻ 새 용량 예약 생성
IAM 역할 ⓘ	<input type="text" value="없음"/>	↻ 새 IAM 역할 생성
종료 방식 ⓘ	<input type="text" value="중지"/>	
최대 절전 중지 동작 ⓘ	<input type="checkbox"/> 추가 종료 동작으로 최대 절전 모드를 활성화	
종료 방지 기능 활성화 ⓘ	<input type="checkbox"/> 우발적인 종료로부터 보호	
모니터링 ⓘ	<input type="checkbox"/> CloudWatch 세부 모니터링 활성화	

4 EC2 인스턴스 생성(4)

단계4: 스토리지 추가하기

인스턴스가 다음 스토리지 디바이스 설정으로 시작됩니다. 추가 EBS 볼륨 및 인스턴스 스토어 볼륨을 인스턴스에 연결하거나 루트 볼륨의 설정을 편집할 수 있습니다. 인스턴스를 시작한 후 추가 EBS 볼륨을 연결할 수도 있지만, 인스턴스 스토어 볼륨은 연결할 수 없습니다. Amazon EC2의 스토리지 옵션에 대해 [자세히 알아보십시오](#).

볼륨 유형 ⓘ	디바이스 ⓘ	스냅샷 ⓘ	크기(GiB) ⓘ	볼륨 유형 ⓘ	IOPS ⓘ	처리량(MB/초) ⓘ	종료 시 삭제 ⓘ	암호화 ⓘ
루트	/dev/sda1	snap-038b38aa00713b5d3	8	범용 SSD(gp2) ▼	100/3000	해당 사항 없음	<input checked="" type="checkbox"/>	암호화되지 않음 ▼

새 볼륨 추가

프리 티어 사용 가능 고객은 최대 30GB의 EBS 범용(SSD) 또는 마그네틱 스토리지를 사용할 수 있습니다. 프리 티어 자격 및 사용량 제한에 대해 [자세히 알아보기](#).

4 EC2 인스턴스 생성(5)

단계5: 태그 추가하기

태그는 대소문자를 구별하는 키-값 페어로 이루어져 있습니다. 예를 들어 키가 Name이고 값이 Webserver인 태그를 정의할 수 있습니다.

태그 복사본은 볼륨, 인스턴스 또는 둘 다에 적용될 수 있습니다.

태그는 모든 인스턴스 및 볼륨에 적용됩니다. Amazon EC2 리소스 태그 지정에 대해 [자세히 알아보기](#).

키 (최대 128자)	값 (최대 256자)	인스턴스 ⓘ	볼륨 ⓘ	
webServer	웹서버	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="button" value="X"/>

(최대 50개 태그)

4 EC2 인스턴스 생성(6)

단계6: 보안그룹 설정하기

보안 그룹 할당: ☒ 새 보안 그룹 생성

☐ 기존 보안 그룹 선택

보안 그룹 이름:

설명:

유형 ⓘ	프로토콜 ⓘ	포트 범위 ⓘ	소스 ⓘ	설명 ⓘ	
SSH ▾	TCP	22	내 IP ▾ 121.191.67.175/32	예: SSH for Admin Desktop	✕
HTTP ▾	TCP	80	위치 무관 ▾ 0.0.0.0/0, ::/0	예: SSH for Admin Desktop	✕

규칙 추가

4 EC2 인스턴스 생성(7)

단계7: 인스턴스 비밀번호 생성

기존 키 페어 선택 또는 새 키 페어 생성

키 페어는 AWS에 저장하는 퍼블릭 키와 사용자가 저장하는 프라이빗 키 파일로 구성됩니다. 이 둘을 모두 사용하여 SSH를 통해 인스턴스에 안전하게 접속할 수 있습니다. Windows AMI의 경우 인스턴스에 로그인하는 데 사용되는 암호를 얻으려면 프라이빗 키 파일이 필요합니다. Linux AMI의 경우, 프라이빗 키 파일을 사용하면 인스턴스에 안전하게 SSH로 연결할 수 있습니다.

참고: 선택한 키 페어가 이 인스턴스에 대해 승인된 키 세트에 추가됩니다. 퍼블릭 AMI에서 기존 키 페어 제거에 대해 자세히 알아보십시오.

새 키 페어 생성

키 페어 이름

awspwd

키 페어 다운로드



계속하려면 먼저 프라이빗 키 파일(*.pem 파일)을 다운로드해야 합니다. 액세스할 수 있는 안전한 위치에 저장합니다. 파일은 생성되고 나면 다시 다운로드할 수 없습니다.

취소

인스턴스 시작

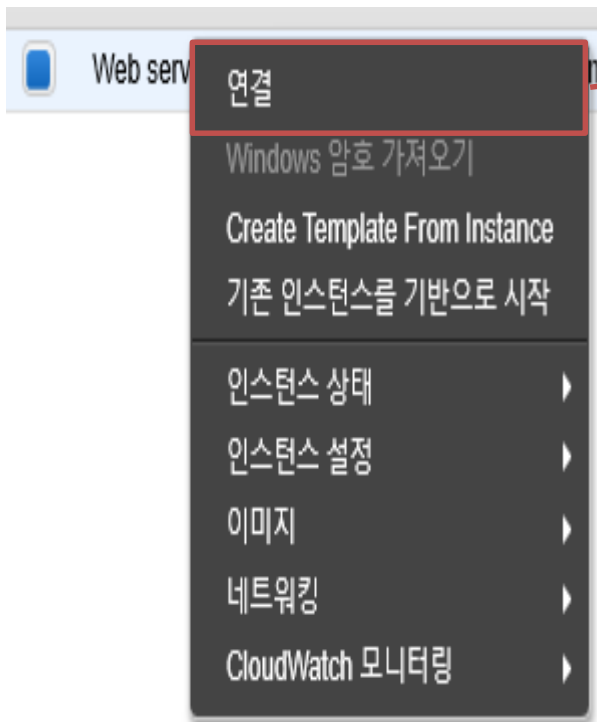
키페어를 컴퓨터 내부
에 안전한 장소에 저장!

4 생성된 EC2 확인

aws콘솔의 서비스 -> EC2 들어간후,
왼쪽 인스턴스 클릭

🔍 태그 및 속성별 필터 또는 키워드별 검색					
<input checked="" type="checkbox"/>	Name ▾	인스턴스 ID ▲	인스턴스 유형 ▾	가용 영역 ▾	인스턴스 상태 ▾
<input checked="" type="checkbox"/>	Web server	i-09f2a17e91bde7e04	t2.micro	ap-northeast-2c	● running

4 AWS 인스턴스 연결하기



인스턴스에 연결

- 연결 방법
- ☒ 독립 실행형 SSH 클라이언트 ⓘ
 - ☐ Session Manager ⓘ
 - ☐ EC2 인스턴스 연결(브라우저 기반 SSH 연결) ⓘ

인스턴스 액세스 방법:

1. SSH 클라이언트를 엽니다([PuTTY를 사용하여 연결](#) 방법 알아보기).
2. 프라이빗 키 파일(aws_password.pem)을 찾습니다. 마법사가 인스턴스를 시작하는 데 사용되는 키를 자동으로 검색합니다.
3. SSH가 작동하려면 키가 공개적으로 표시되지 않아야 합니다. 필요할 경우 이 명령을 사용합니다.

```
chmod 400 aws_password.pem
```

4. 퍼블릭 DNS을(를) 사용하여 인스턴스에 연결:

```
ec2-52-79-242-45.ap-northeast-2.compute.amazonaws.com
```

예:

```
ssh -i "aws_password.pem" ubuntu@ec2-52-79-242-45.ap-northeast-2.compute.amazonaws.com
```

대부분의 경우 위의 사용자 이름이 맞지만, AMI 사용 지침을 숙지하여 AMI 소유자가 기본 AMI 사용자 이름을 변경하지 않도록 하십시오.

인스턴스에 연결하는 데 도움이 필요한 경우 [연결 설명서](#)을(를) 참조하십시오.

생성된 인스턴스 연결클릭

닫기

4 AWS 인스턴스 연결하기(1) [윈도우환경]

XShell 설치

<https://www.netsarang.com/ko/free-for-home-school/>



4 AWS 인스턴스 연결하기(2) [윈도우환경]

파일->열기->새로만들기



새 세션 등록 정보

범주(C):

- 연결
 - 사용자 인증
 - 로그인 프로토포트
 - 로그인 스크립트
 - SSH
 - 보안
 - 터널링
 - SFTP
 - TELNET
 - RLOGIN
 - SERIAL
 - 프록시
 - 연결 유지
 - 터미널
 - 키보드

연결

일반

이름(N): Webserver

프로토콜(P): SSH

호스트(H): ec2-52-79-242-45.ap-north

포트 번호(O): 22

설명(D):

생성한 인스턴스의 퍼블릭 DNS를 입력
(28슬라이드참고)

4 AWS 인스턴스 연결하기(3) [윈도우환경]

방법을 Public키로 변경

ubuntu 입력

생성한 인스턴스의 키페어파일을 등록

새 세션 등록 정보

범주(C):

- 연결
 - 사용자 인증**
 - 로그인 프롬프트
 - 로그인 스크립트
- SSH
 - 보안
 - 터널링
 - SFTP
- TELNET
- RLOGIN
- SERIAL
- 프록시
- 연결 유지
- 터미널
 - 키보드
 - VT 모드

연결 > 사용자 인증

인증 방법과 기타 관련 매개 변수들을 선택하십시오.

이 섹션은 로그인 할 때 시간을 절약하기 위해 사용할 수 있습니다. 그러나 보안을 중요시하는 경우 이 섹션을 비워 두는 것이 좋습니다.

방법(M):

Public Key

설정(S)...

사용자 이름(U):

ubuntu

암호(P):

사용자 키(K):

aws_password

찾아보기(B)...

암호(A):

4 AWS 인스턴스 연결하기(4) [윈도우환경]

```
1 Web server x +
Xshell 6 (Build 0193)
Copyright (c) 2002 NetSarang Computer, Inc. All rights reserved.

Type 'help' to learn how to use Xshell prompt.
[C:\~]$

Host 'ec2-52-79-242-45.ap-northeast-2.compute.amazonaws.com' resolved to 52.79.242.45.
Connecting to 52.79.242.45:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+J'.

Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1057-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Mon Mar 30 03:22:52 UTC 2020

System load:  0.0           Processes:            89
Usage of /:   13.8% of 7.69GB Users logged in:      1
Memory usage: 15%          IP address for eth0: 172.31.33.127
Swap usage:   0%

0 packages can be updated.
0 updates are security updates.

Last login: Mon Mar 30 03:13:07 2020 from 121.191.67.175
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-33-127:~$
```

연결 완료된 모습

5. EC2(ubuntu)에 docker 설치하기(1)

```
$ sudo wget -qO- https://get.docker.com/ | sh
```

Docker는 리눅스 배포판 종류를 자동으로 인식하여 Docker 패키지를 설치해주는 스크립트를 제공합니다.

5. EC2(ubuntu)에 docker 설치하기(2)

```
ubuntu@ip-172-31-33-127:~$ sudo wget -qO- https://get.docker.com/ | sh
# Executing docker install script, commit: 442e66405c304fa92af8aadaald9b31bf4b0ad94
+ sudo -E sh -c apt-get update -qq >/dev/null
+ sudo -E sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq apt-transport-https ca-certificates curl >/dev/null
+ sudo -E sh -c curl -fsSL "https://download.docker.com/linux/ubuntu/gpg" | apt-key add -qq - >/dev/null
Warning: apt-key output should not be parsed (stdout is not a terminal)
+ sudo -E sh -c echo "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable" > /etc/apt/sources.list.d/docker.list
+ sudo -E sh -c apt-get update -qq >/dev/null
+ [ -n ]
+ sudo -E sh -c apt-get install -y -qq --no-install-recommends docker-ce >/dev/null
+ sudo -E sh -c docker version
Client: Docker Engine - Community
 Version:      19.03.8
 API version:  1.40
 Go version:   go1.12.17
 Git commit:   afacb8b7f0
 Built:        Wed Mar 11 01:25:46 2020
 OS/Arch:      linux/amd64
 Experimental: false

Server: Docker Engine - Community
 Engine:
  Version:      19.03.8
  API version:  1.40 (minimum version 1.12)
  Go version:   go1.12.17
  Git commit:   afacb8b7f0
  Built:        Wed Mar 11 01:24:19 2020
  OS/Arch:      linux/amd64
  Experimental: false
 containerd:
  Version:      1.2.13
  GitCommit:    7ad184331fa3e55e52b890ea95e65ba581ae3429
 runc:
  Version:      1.0.0-rc10
  GitCommit:    dc9208a3303feef5b3839f4323d9beb36df0a9dd
 docker-init:
  Version:      0.18.0
  GitCommit:    fec3683
If you would like to use Docker as a non-root user, you should now consider
adding your user to the "docker" group with something like:
```

Docker가 설치된 모습

5. EC2(ubuntu)에 docker 설치하기(3)

Docker는 항상 관리자 권한(sudo)으로 접속 해야하므로, 쉽게 접속을 하기위해, 아이디에 권한을 부여한다.

```
ubuntu@ip-172-31-33-127:~$ sudo usermod -aG docker ubuntu
ubuntu@ip-172-31-33-127:~$
```

5. Pull 명령으로 이미지받기

```
$ sudo docker pull ubuntu:latest
```

Docker pull <이미지이름>:<태그> 형식입니다.
Latest를 입력하면 최신버전을 받습니다.

```
ubuntu@ip-172-31-33-127:~$ docker run -i -t --name hello ubuntu /bin/bash
root@82dbed780cb8:/#
```

Docker 컨테이너로 들어간모습

5. Images 명령으로 이미지 목록 출력하기

```
ubuntu@ip-172-31-33-127:~$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
ubuntu	latest	4e5021d210f6	3 weeks ago	64.2MB

앞서 설치했던 ubuntu 이미지가 잘 설치되었음

5. run 명령으로 컨테이너 생성하기

```
$ sudo docker run -i -t --name hello ubuntu /bin/bash
```

Docker run <옵션> <이미지이름> <실행할 파일> 형식입니다. 여기서는, ubuntu 이미지를 컨테이너로 생성하고 /bin/bash를 실행

```
root@82dbed780cb8:/#
```

Docker 컨테이너로 생성과 동시에들어옴

5. ps명령으로 컨테이너 목록 출력하기

Exit 을하고, 호스트로 돌아간후 , 컨테이너목록을 확인해본다
-a 옵션을 주면 정지된 컨테이너까지 확인가능

```
root@82dbed780cb8:/# exit
exit
ubuntu@ip-172-31-33-127:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
TS	NAMES				
82dbed780cb8	ubuntu	"/bin/bash"	14 minutes ago	Exited (127) 3 seconds ago	
	hello				

```
ubuntu@ip-172-31-33-127:~$
```

5. Docker 컨테이너 시작, 정지

(시작) docker start <컨테이너이름>

```
$ sudo docker start hello
```

(정지) docker stop <컨테이너이름>

```
$ sudo docker stop hello
```

5. Docker 컨테이너 접속

```
docker attach <컨테이너이름>
```

```
ubuntu@ip-172-31-33-127:~$ docker attach hello
root@82dbed780cb8:/#
```