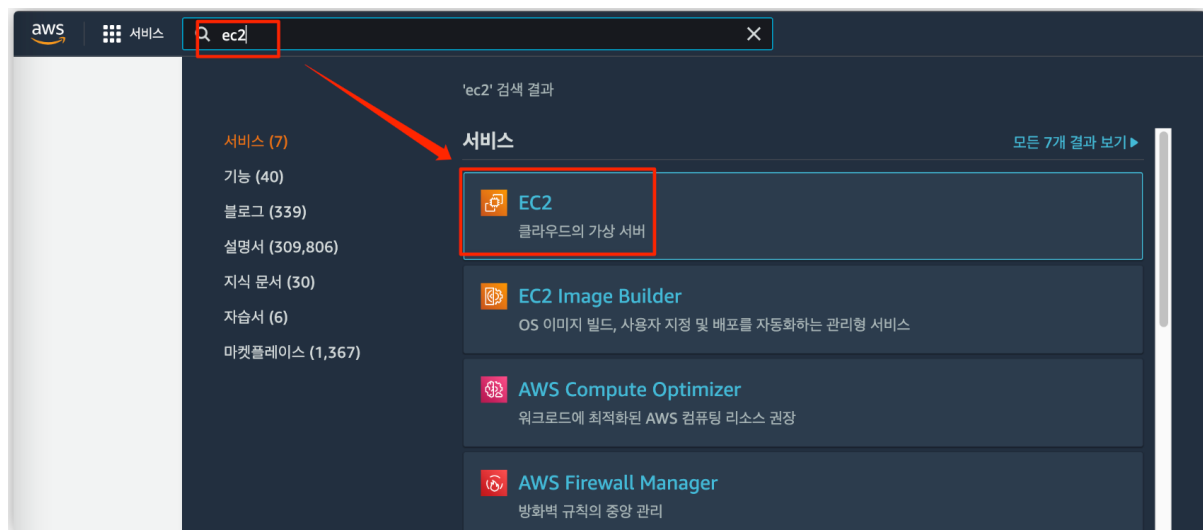


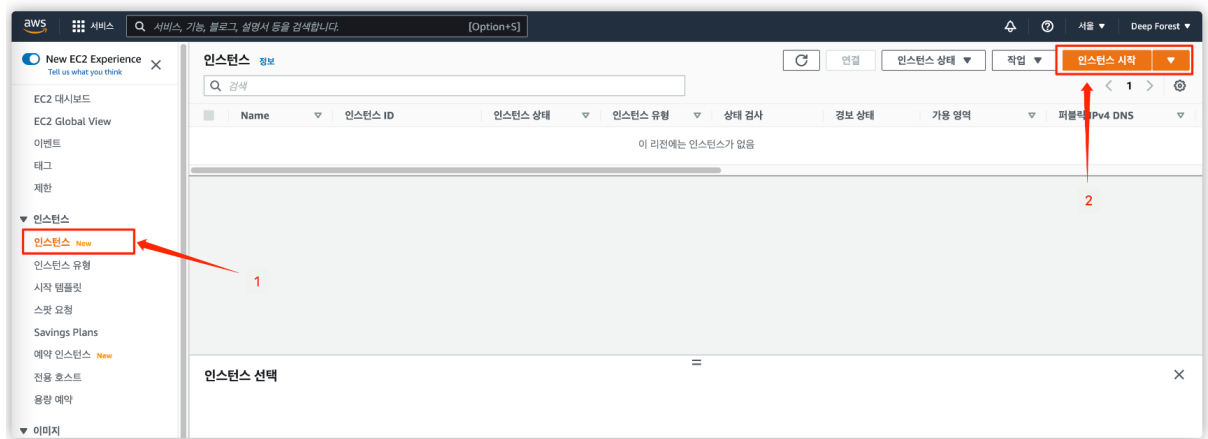
AWS + linux (문서화)

Assignee	
Status	
Priority	진행중 🔥
Due Date	
Date Created	@2022년 12월 22일 오후 5:15
Attachment	
Project	

EC2 메뉴 이동



새 인스턴스 생성



-인스턴스 유형 선택

이름 및 태그

이름

비어있어도 됨

예: 내 웹 서버

추가 태그 추가

▼ 애플리케이션 및 OS 이미지(Amazon Machine Image) 정보

AMI는 인스턴스를 시작하는 데 필요한 소프트웨어 구성(운영 체제, 애플리케이션 서버 및 애플리케이션)이 포함된 템플릿입니다. 아래에서 찾고 있는 항목이 보이지 않으면 AMI를 검색하거나 찾아보십시오.

🔍 수천 개의 애플리케이션 및 OS 이미지를 포함하는 전체 카탈로그 검색

최근 사용

Quick Start

Amazon
Linux



Ubuntu



Windows



Red Hat



SUSE Linux



더 많은 AMI 찾아보기

AWS, Marketplace 및 커뮤니티의 AMI 포함

Amazon Machine Image(AMI)

Ubuntu Server 20.04 LTS (HVM), SSD Volume Type

ami-0454bb2f7de534 (64비트(x86)) / ami-03665e633d23b2df6 (64비트(Arm))

가상화: hvm ENA 활성화됨: true 루트 디바이스 유형: ebs

프리 티어 사용 가능

설명

Canonical, Ubuntu, 20.04 LTS, amd64 focal image build on 2021-11-29

아키텍처

AMI ID

64비트(x86)

ami-0454bb2f7de534

▼ 인스턴스 유형 정보

인스턴스 유형

t2.micro

패밀리: t2 1 vCPU 1 GiB 메모리

온디맨드 Linux 요금: 0.0144 USD 시간당

온디맨드 Windows 요금: 0.019 USD 시간당

프리 티어 사용 가능

인스턴스 유형 비교

키 페어 생성

키 페어 생성

✕

키 페어를 사용하면 인스턴스에 안전하게 연결할 수 있습니다.

아래에 키 페어의 이름을 입력합니다. 메시지가 표시되면 프라이빗 키를 사용자 컴퓨터의 안전하고 액세스 가능한 위치에 저장합니다. 나중에 인스턴스에 연결할 때 필요합니다. [자세히 알아보기](#)

키 페어 이름

my-key

이름은 최대 255개의 ASCII 문자를 포함할 수 있습니다. 선행 또는 후행 공백은 포함할 수 없습니다.

키 페어 유형

☒ RSA
RSA 암호화된 프라이빗 및 퍼블릭 키 페어

☐ ED25519
ED25519 암호화된 프라이빗 및 퍼블릭 키 페어(Windows 인스턴스에는 지원되지 않음)

프라이빗 키 파일 형식

☒ .pem
OpenSSH와 함께 사용

☐ .ppk
PuTTY와 함께 사용

취소

키 페어 생성

-네트워크 및 스토리지 선택

▼ 네트워크 설정

[편집](#)

네트워크

vpc-9d8868f6

서브넷

기본 설정 없음(가용 영역의 기본 서브넷)

퍼블릭 IP 자동 할당

활성화

보안 그룹(방화벽) [정보](#)

다음 규칙을 사용하여 'launch-wizard-1'(이)라는 새 보안 그룹을 생성합니다.

☒ 에서 SSH 트래픽 허용

인스턴스 연결에 도움

내 IP

☐ 인터넷에서 HTTPS 트래픽 허용

예를 들어 웹 서버를 생성할 때 엔드포인트를 설정하려면

☐ 인터넷에서 HTTP 트래픽 허용

예를 들어 웹 서버를 생성할 때 엔드포인트를 설정하려면

▼ 스토리지 구성 [정보](#)

[어드밴스드](#)

1x

30

GiB

gp2

루트 볼륨

[?](#) 프리 티어를 사용할 수 있는 고객은 최대 30GB의 EBS 범용(SSD)또는 마그네틱 스토리지를 사용할 수 있습니다.

×

새 볼륨 추가

선택한 AMI에 인스턴스가 허용하는 것보다 많은 인스턴스 스토어 볼륨이 포함되어 있습니다. AMI에서 처음 0개의 인스턴스 스토어 볼륨에만 액세스할 수 있습니다.

0 x 파일 시스템

[편집](#)

▼ 요약

인스턴스 개수 정보

1

소프트웨어 이미지(AMI)

Canonical, Ubuntu, 20.04 LTS, amd64 focal image
build on 2021-11-29
ami-0454bb2fefc7de534

가상 서버 유형(인스턴스 유형)

t2.micro

방화벽(보안 그룹)

새 보안 그룹

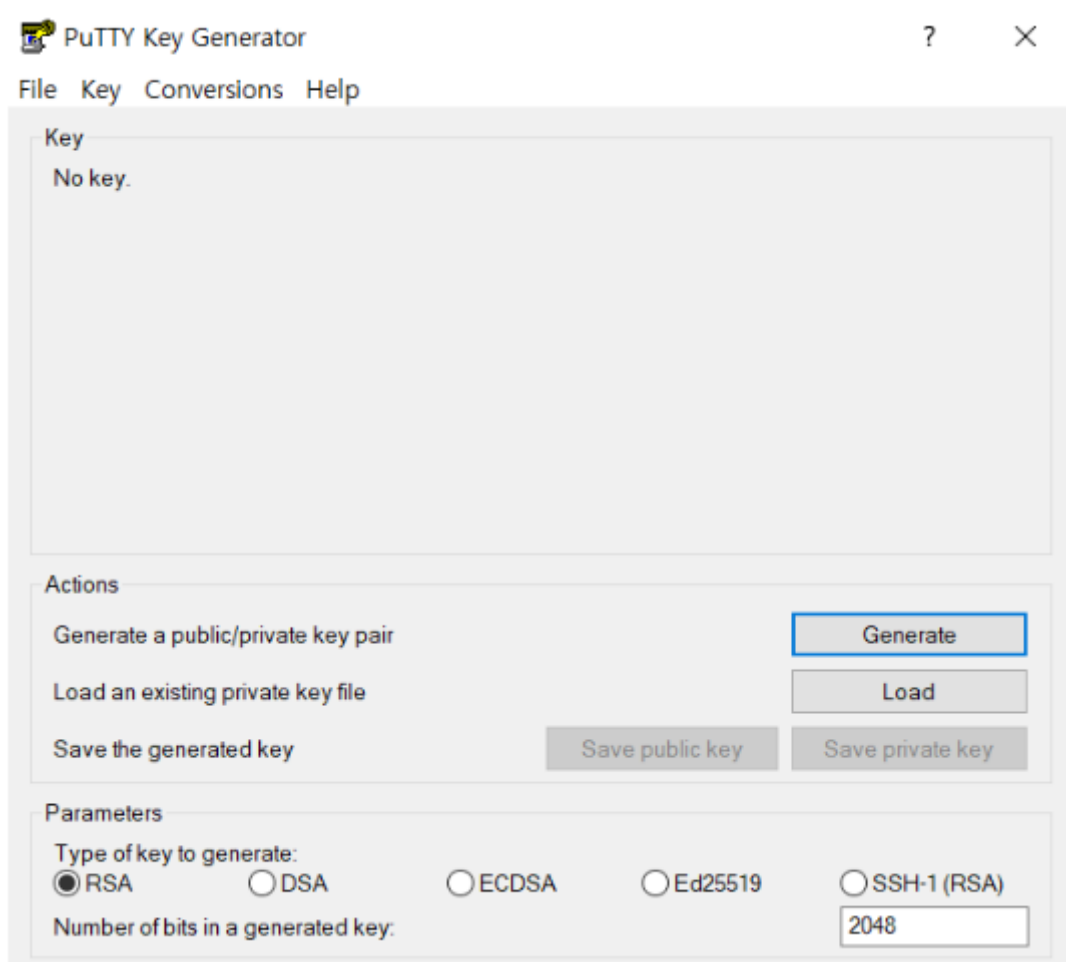
스토리지(볼륨)

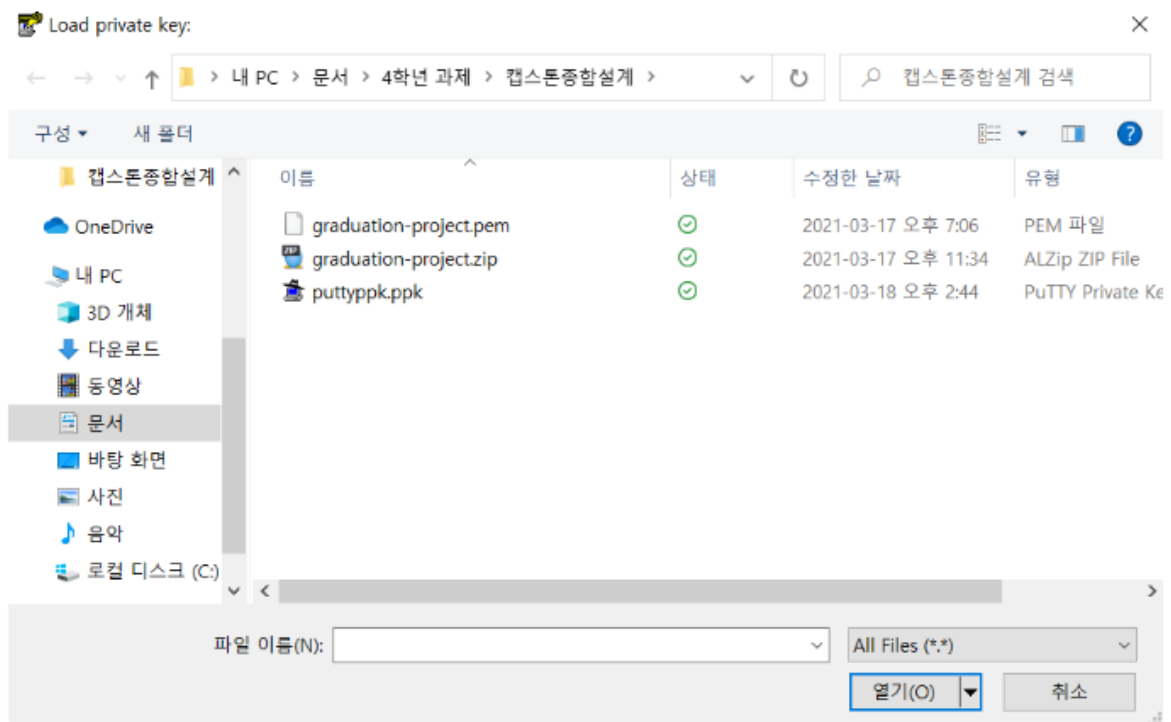
1개의 볼륨 – 30GiB

취소

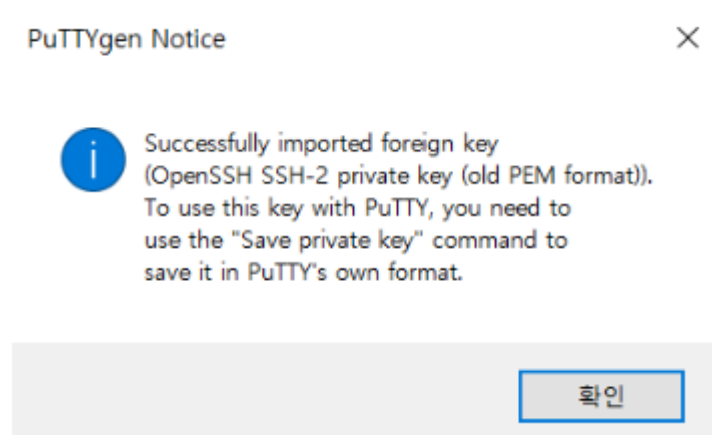
인스턴스 시작

EC2 인스턴스에 PUTTY 이용해서 접속하기
Puttygen을 통해 .ppk 파일 만들기
인스턴스 생성시 만든 .PEM키를 불러와 Load를 눌러 생성.

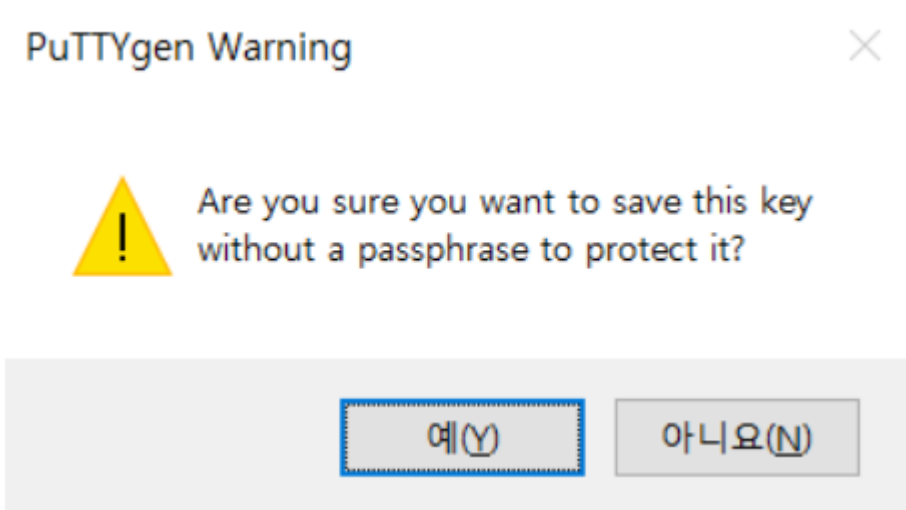




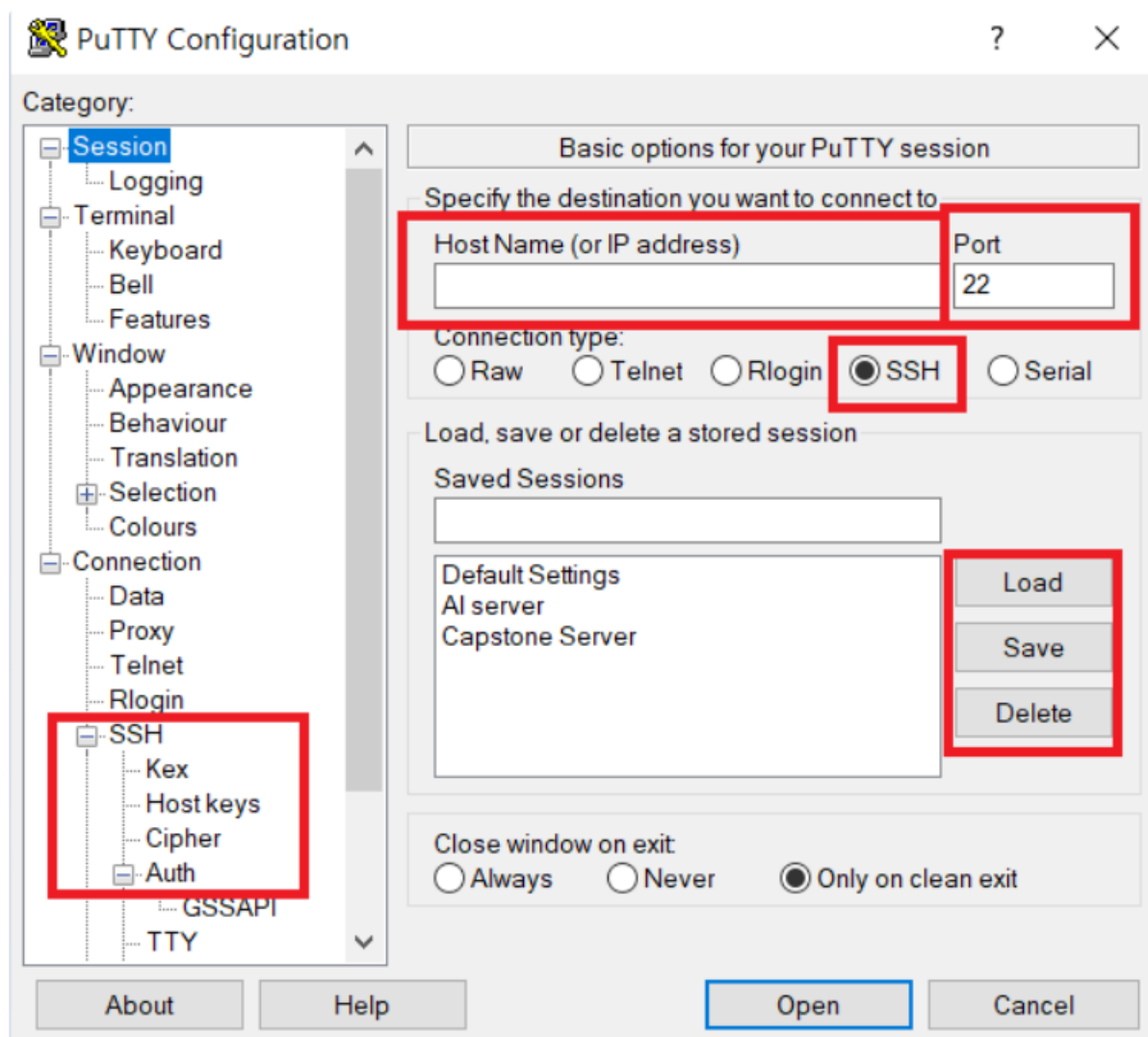
정상적으로 로드된다면 창이 뜰것.



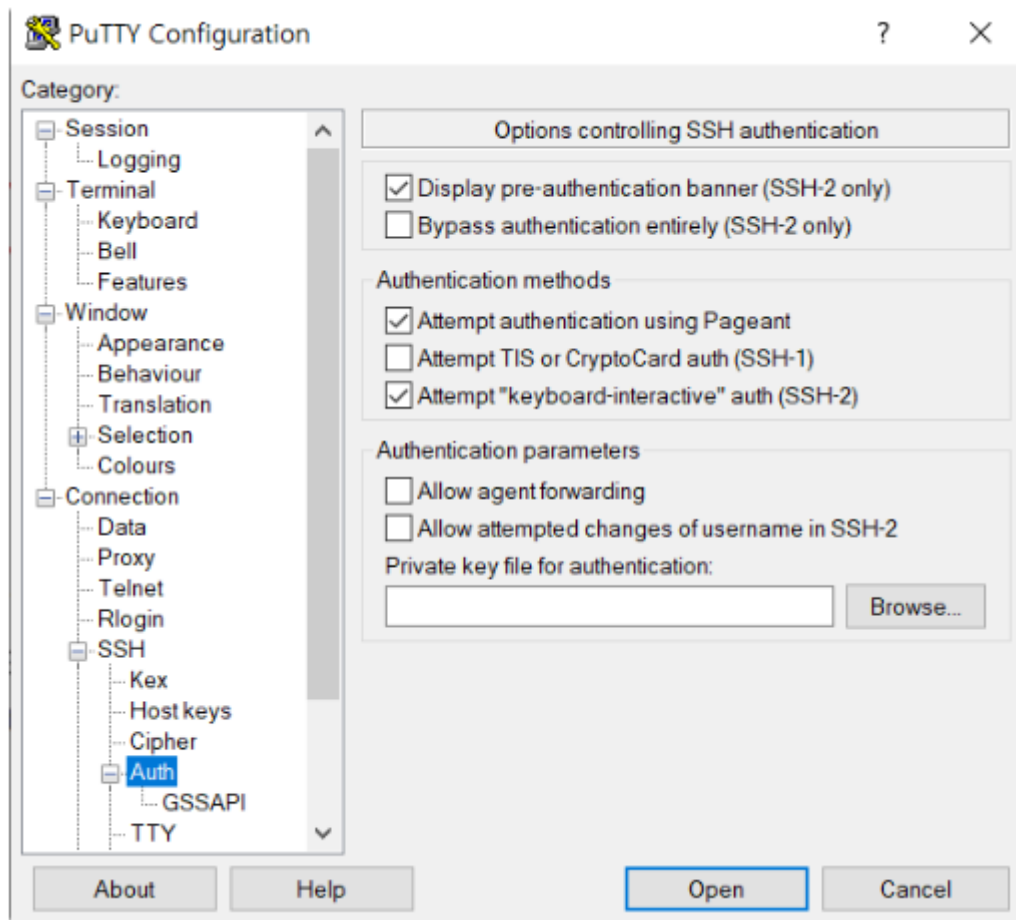
Save Private key 를 통해 최종적으로 키를 생성해줌



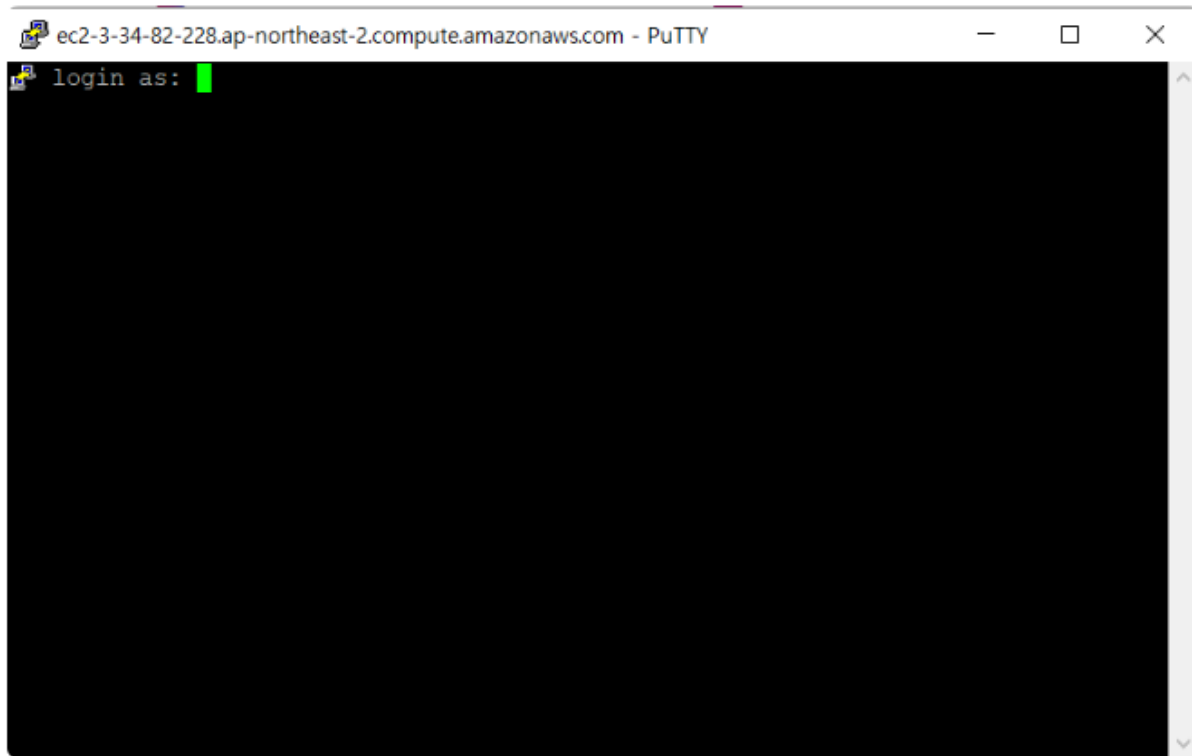
EC2에서 받은 정보들을 각 칸에 넣어준다



만든 private key를 putty 와 연결해준다.



ubuntu 입력 후 엔터로 로그인해줌



1. 패키지 업데이트

`sudo apt-get update`

2. jdk 설치

`sudo apt-get install openjdk-8-jdk`

```
done.  
done.  
Processing triggers for mime-support (3.64ubuntu1) ...  
Processing triggers for libc-bin (2.31-0ubuntu9) ...  
Processing triggers for libgdk-pixbuf2.0-0:amd64 (2.40.0+dfsg-3) ...  
root@ip-172-31-47-244:~#
```

환경변수 설정

which를 통해 java의 경로 가져온 후 해당의 경로를

readlink를 통해 심볼릭 링크가 연결되어 있는 실제 파일의 경로를 가져온다.(readlink를 통해 나온 경로가 환경변수 JAVA_HOME에 등록될 경로이다.

```
ubuntu@ip-172-31-39-121:~$ which java
/usr/bin/java
ubuntu@ip-172-31-39-121:~$ readlink -f /usr/bin/java
/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java
ubuntu@ip-172-31-39-121:~$
```

/etc/profile 파일 편집으로 환경변수를 입력한다. root 권한이 없으므로 sudo 명령어를 사용한다.

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java
export PATH=$PATH:$JAVA_HOME/bin:$PATH
export CLASS_PATH=$JAVA_HOME/lib:$CLASS_PATH
```

위의 내용을

편집모드로 열린 파일에서 시프트+g 키 입력하여 제일 하단에 환경변수로 등록한다.
등록 후 파일 내용 저장 및 편집모드를 나간다.
명령어 : wq (w : 저장, q : 편집모드 종료)

```
ubuntu@ip-172-31-39-121:~$ sudo vi /etc/profile
```

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java
export PATH=$PATH:$JAVA_HOME/bin
export CLASS_PATH=$JAVA_HOME/lib:$CLASS_PATH
```

환경변수 적용 및 확인

```
source /etc/profile
echo $JAVA_HOME
```

```
ubuntu@ip-172-31-39-121:~$ source /etc/profile
ubuntu@ip-172-31-39-121:~$ echo $JAVA_HOME
/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java
```

톰캣 설치

사용할 톰캣 버전의 tar 파일을 서버로 wget을 사용하여 다운로드를 진행한다.

<https://tomcat.apache.org> 해당 사이트에 사용한 톰캣 다운로드 링크를 복사한다.

Download

Which version?
Tomcat 11 (alpha)
Tomcat 10
Tomcat 9
Tomcat 8
Tomcat Migration Tool
for Jakarta EE
Tomcat Connectors
Tomcat Native
Taglibs
Archives

Documentation

Tomcat 11.0 (alpha)
Tomcat 10.1
Tomcat 10.0
Tomcat 9.0
Tomcat 8.5
Tomcat Connectors
Tomcat Native 2
Tomcat Native 1.2
Wiki
Migration Guide
Presentations
Specifications

Problems?

Security Reports
Find help
FAQ
Mailing Lists
Bug Database
IRC

Get Involved

Quick Navigation

[KEYS](#) | [9.0.70](#) | [Browse](#) | [Archives](#)

Release Integrity

You **must** [verify](#) the integrity of the downloaded files. We provide OpenPGP signatures keys of Tomcat's Release Managers. We also provide [SHA-512](#) checksums for every release the same as ours.

Mirrors

You are currently using <https://dlcdn.apache.org/>. If you encounter a problem with the mirrors list) that should be available.

Other mirrors:

9.0.70

Please see the [RELEASE](#)

Binary Distribution

- Core:
 - [zip](#) (pgp, sha512)
 - [tar.gz](#) (pgp, sha512)
 - [32-bit Windows zip](#) (pgp, sha512)
 - [64-bit Windows zip](#) (pgp, sha512)
 - [32-bit/64-bit Windows Service Installer](#) (pgp, sha512)
- Full documentation:
 - [tar.gz](#) (pgp, sha512)
- Deployer:
 - [zip](#) (pgp, sha512)
 - [tar.gz](#) (pgp, sha512)

```

ubuntu@ip-172-31-39-121:~$ sudo wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.70/bin/apache-tomcat-9.0.70.tar.gz
--2022-12-28 11:03:48-- https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.70/bin/apache-tomcat-9.0.70.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11613418 (11M) [application/x-gzip]
Saving to: 'apache-tomcat-9.0.70.tar.gz'

apache-tomcat-9.0.7 100%[=====] 11.08M --.-KB/s in 0.1s

2022-12-28 11:03:49 (77.6 MB/s) - 'apache-tomcat-9.0.70.tar.gz' saved [11613418/11613418]

ubuntu@ip-172-31-39-121:~$ ll
total 11372
drwxr-x--- 4 ubuntu ubuntu 4096 Dec 28 11:03 ./
drwxr-xr-x 3 root root 4096 Dec 22 09:22 ../
-rw-r--r-- 1 ubuntu ubuntu 220 Jan 6 2022 .bash_logout
-rw-r--r-- 1 ubuntu ubuntu 3771 Jan 6 2022 .bashrc
drwx----- 2 ubuntu ubuntu 4096 Dec 23 06:38 .cache/
-rw-r--r-- 1 ubuntu ubuntu 807 Jan 6 2022 .profile
drwx----- 2 ubuntu ubuntu 4096 Dec 22 09:22 .ssh/
-rw-r--r-- 1 ubuntu ubuntu 0 Dec 23 06:41 .sudo_as_admin_successful
-rw-r--r-- 1 root root 11613418 Dec 1 14:20 apache-tomcat-9.0.70.tar.gz
ubuntu@ip-172-31-39-121:~$

```

복사한 톰캣 링크를 사용하여 서버에 다운로드 후 실제로 현 경로에 톰캣 tar 파일이 존재하는지 확인한다.

`sudo wget https://dlcdn.apache.org/tomcat/tomcat-9/v9`

`ll`

다운로드 받은 톰캣을 압축해제 하면 톰캣 설치가 완료된다.

`sudo tar zxvf (톰캣.tar.gz)`

```

ubuntu@ip-172-31-39-121:~$ sudo tar zxvf apache-tomcat-9.0.70.tar.gz

```

`sudo`로 톰캣 다운로드를 진행 하였기 때문에 반드시 `sudo su` 명령어로 root 계정 접속을 하여 bin폴더 안의 톰캣 쉘 파일을 실행해주어야 한다.

`sudo su`

`pwd` (현재 경로가 아파치 톰캣의 bin 경로인지 확인)

`./startup.sh`

```

root@ip-172-31-39-121:/home/ubuntu/apache-tomcat-9.0.70/bin# sudo su
root@ip-172-31-39-121:/home/ubuntu/apache-tomcat-9.0.70/bin# pwd
/home/ubuntu/apache-tomcat-9.0.70/bin
root@ip-172-31-39-121:/home/ubuntu/apache-tomcat-9.0.70/bin# ./
catalina.sh      configtest.sh    digest.sh        setclasspath.sh  startup.sh        version.sh
ciphers.sh       daemon.sh        makebase.sh      shutdown.sh       tool-wrapper.sh
root@ip-172-31-39-121:/home/ubuntu/apache-tomcat-9.0.70/bin# ./startup.sh
Using CATALINA_BASE:   /home/ubuntu/apache-tomcat-9.0.70
Using CATALINA_HOME:   /home/ubuntu/apache-tomcat-9.0.70
Using CATALINA_TMPDIR: /home/ubuntu/apache-tomcat-9.0.70/temp
Using JRE_HOME:        /usr
Using CLASSPATH:        /home/ubuntu/apache-tomcat-9.0.70/bin/bootstrap.jar:/home/ubuntu/apache-tomcat-9.0.70/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.

```

톰캣이 정상적으로 서버에 올라가 졌는지 확인한다.

`ps -ef | grep tomcat`

```

root@ip-172-31-39-121:/home/ubuntu/apache-tomcat-9.0.70/bin# ps -ef | grep tomcat
root      23270      1   11:12 pts/1    00:00:03 /usr/bin/java -Djava.util.logging.config.file=/home/ubuntu/apache-tomcat-9.0.70/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djdk.tls.ephemeralDHKeySize=2048 -Djava.protocol.handler.pkgs=org.apache.catalina.webresources -Dorg.apache.catalina.security.SecurityListener.UMASK=0027 -Dignore.endorsed.dirs= -classpath /home/ubuntu/apache-tomcat-9.0.70/bin/bootstrap.jar:/home/ubuntu/apache-tomcat-9.0.70/bin/tomcat-juli.jar -Dcatalina.base=/home/ubuntu/apache-tomcat-9.0.70 -Dcatalina.home=/home/ubuntu/apache-tomcat-9.0.70 -Djava.io.tmpdir=/home/ubuntu/apache-tomcat-9.0.70/temp org.apache.catalina.startup.Bootstrap start
root      23359      23318   0 11:17 pts/2    00:00:00 grep --color=auto tomcat
root@ip-172-31-39-121:/home/ubuntu/apache-tomcat-9.0.70/bin#

```

톰캣이 정상적으로 실행을 하여도 aws 보안 설정으로 접속이 안될것이며

EC2 → 네트워크 및 보안 → 보안그룹 → 인바운드 규칙 편집에서 8080 포트 허용을 해준다.

보안 그룹 (1/2) 정보

🔄

작업 ▼

보안 그룹을 CSV로 내보내기 ▼

보안 그룹 생성

<

1

>

⚙️

<input type="checkbox"/>	Name	보안 그룹 ID	보안 그룹 이름	VPC ID	설명	소유자
<input type="checkbox"/>	-	sg-078da1660f1afca8d	default	vpc-01febb3a58f27f673	default VPC security gr...	868597909800
<input checked="" type="checkbox"/>	-	sg-0b4f9b6c2ec880f5b	launch-wizard-1	vpc-01febb3a58f27f673	launch-wizard-1 create...	868597909800

sg-0b4f9b6c2ec880f5b - launch-wizard-1

세부 정보

인바운드 규칙

아웃바운드 규칙

태그

📘

이제 Reachability Analyzer를 사용하여 네트워크 연결을 확인할 수 있습니다.

Reachability Analyzer 실행

✕

인바운드 규칙 (2)

🔄

태그 관리

인바운드 규칙 편집

AWS + linux (문서화)

15

EC2 > 보안 그룹 > sg-0b4f9b6c2ec880f5b - launch-wizard-1 > 인바운드 규칙 편집

인바운드 규칙 편집 정보

인바운드 규칙은 인스턴스에 도달하도록 허용된 수신 트래픽을 제어합니다.

보안 그룹 규칙 ID	유형 <small>정보</small>	프로토콜 <small>정보</small>	포트 범위 <small>정보</small>	소스 <small>정보</small>	설명 - 선택 사항 <small>정보</small>	삭제
sg-063d7d6ed444555b1	SSH	TCP	22	사용자 ...		
sg-08ba9c30c2f38d226	사용자 지정 TCP	TCP	8080	사용자 ...		


[규칙 추가](#)

[취소](#) [변경 사항 미리 보기](#) [규칙 저장](#)


아래와 같이 톰캣 화면이 접속이 된다면 성공
<http://접속ip:8080/>

Home Documentation Configuration Examples Wiki Mailing Lists [Find Help](#)

Apache Tomcat/9.0.70

 **APACHE** SOFTWARE FOUNDATION
<http://www.apache.org/>

If you're seeing this, you've successfully installed Tomcat. Congratulations!



Recommended Reading:
[Security Considerations How-To](#)
[Manager Application How-To](#)
[Clustering/Session Replication How-To](#)

[Server Status](#)
[Manager App](#)
[Host Manager](#)

Developer Quick Start

[Tomcat Setup](#)
[First Web Application](#)

[Realms & AAA](#)
[JDBC DataSources](#)

[Examples](#)

[Servlet Specifications](#)
[Tomcat Versions](#)

Managing Tomcat

For security, access to the [manager webapp](#) is restricted. Users are defined in:

```
$CATALINA_HOME/conf/tomcat-users.xml
```

In Tomcat 9.0 access to the manager application is split between different users.
[Read more...](#)

[Release Notes](#)
[Changelog](#)
[Migration Guide](#)
[Security Notices](#)

Documentation

[Tomcat 9.0 Documentation](#)
[Tomcat 9.0 Configuration](#)
[Tomcat Wiki](#)

Find additional important configuration information in:

```
$CATALINA_HOME/RUNNING.txt
```

Developers may be interested in:

[Tomcat 9.0 Bug Database](#)
[Tomcat 9.0 JavaDocs](#)
[Tomcat 9.0 Git Repository at GitHub](#)

Getting Help

[FAQ and Mailing Lists](#)

The following mailing lists are available:

[tomcat-announce](#)
 Important announcements, releases, security vulnerability notifications. (Low volume).

[tomcat-users](#)
 User support and discussion

[taglibs-user](#)
 User support and discussion for [Apache Taglibs](#)

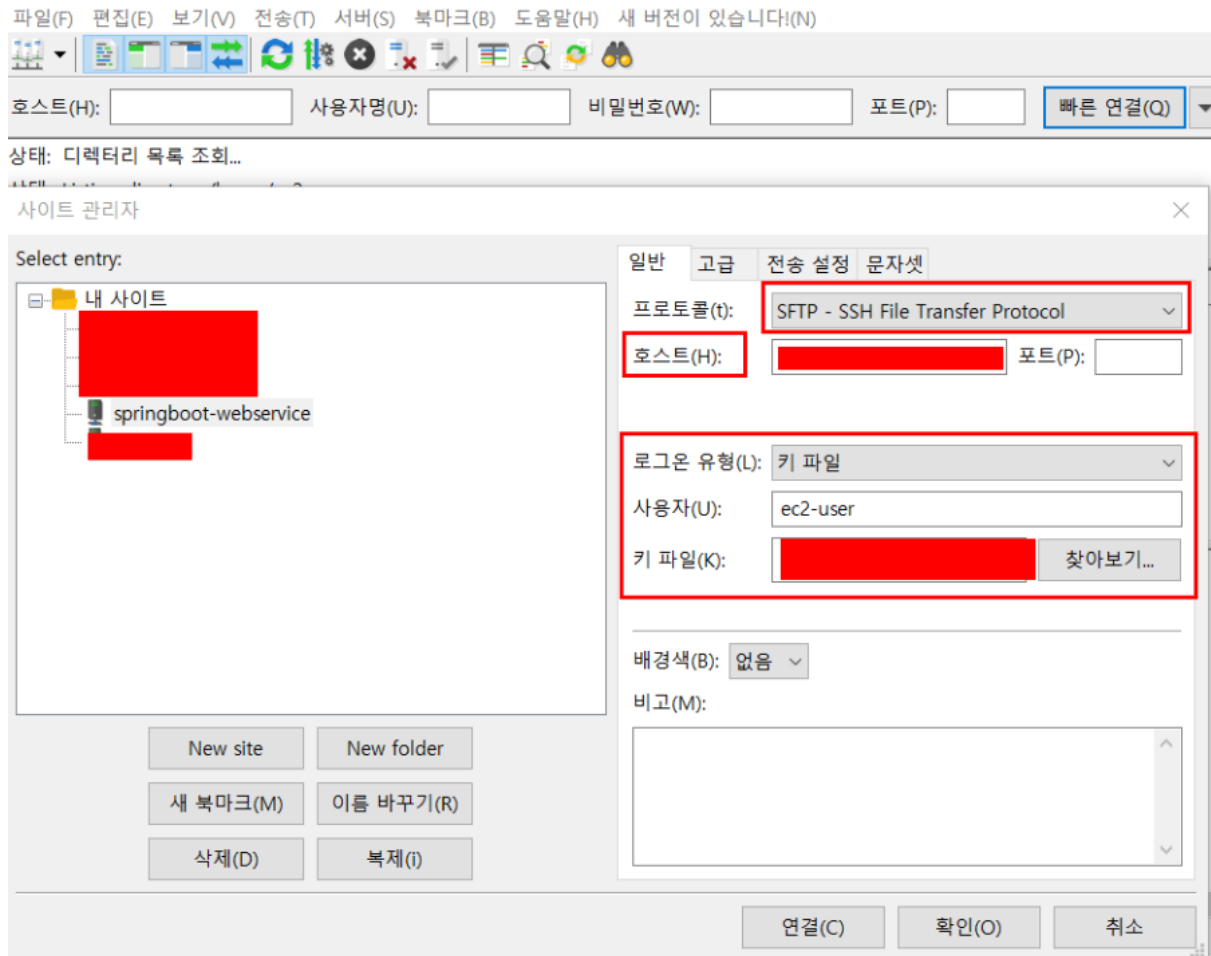
[tomcat-dev](#)
 Development mailing list, including commit messages

Filezilla 설치 및 설정

파일질라 설치 홈페이지 : <https://filezilla-project.org/download.php?type=client>

- 사이트 관리자 추가하기

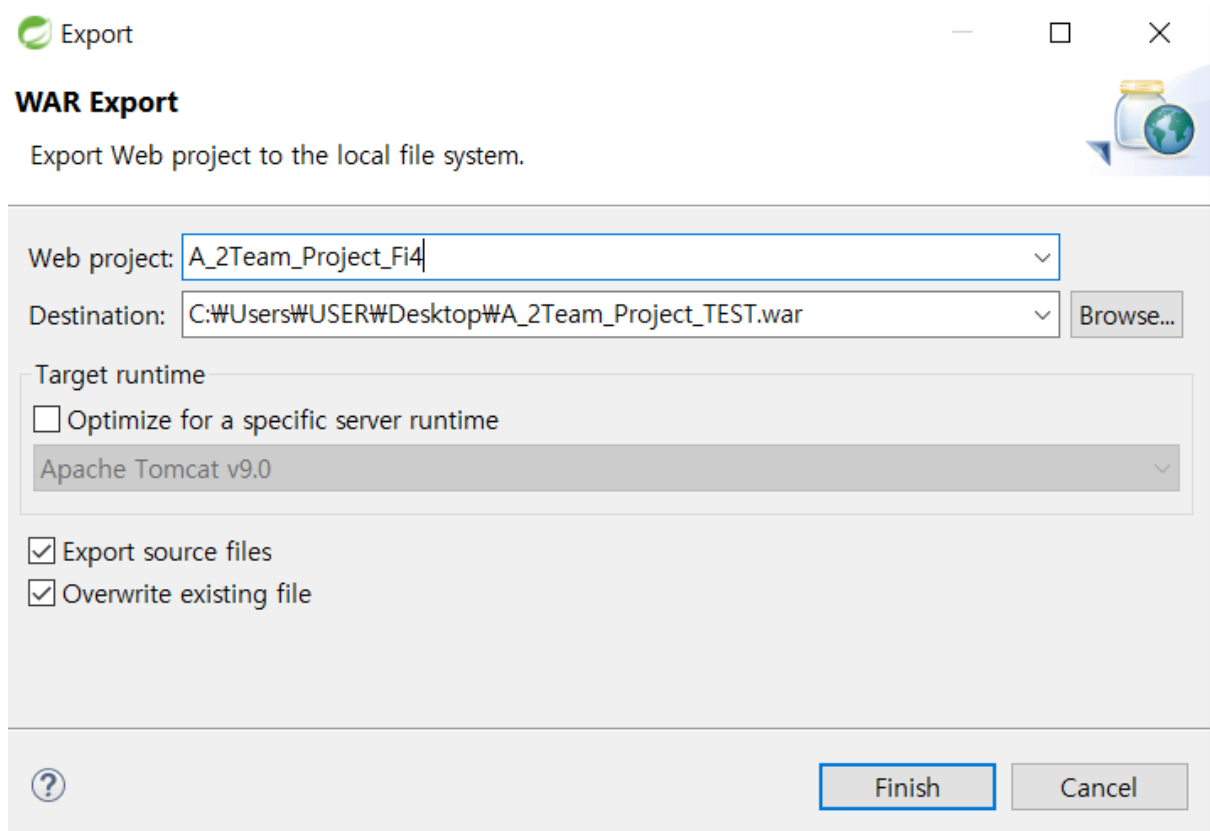
파일 질라 설치 후, 상단 메뉴 탭 중 파일 → 사이트 관리자



war 배포

war 파일 생성을 한다.

프로젝트 Export → WAR file → export 진행



FileZila 는 root 계정 접속이 불가능 하므로 파일 업로드 후 webapp경로로 파일을 옮겨 준다.

webapp에 war파일을 옮기면 톰캣 시작 시 ROOT 경로에 파일이 올라가 서버에 적용을 시킨다.

webapp에 war 파일 옮긴후 아파치/conf/server.xml 에서 <Host name="localhost"

appBase="webapps" 아래에

<Context path="/" docBase="" 프로젝트명'.war' reloadable="true"/>

추가 후 서버 실행을 한다.

Hello World!!|