SSH key를 이용한 로그인 구성 방법 가이드

이 문서는 Linux 환경에서 암호 입력 없이 ssh key를 이용해 원격 서버에 ssh를 연결하는 방법을 정리한다.

배경

서버의 보안이 중요하여 대부분의 조직에서 주기적으로 암호를 변경하고 있다. 그러나 자동화된 시스템의 경우 주기적으로 암호를 변경하는 일은 여간 번거롭지 않다.

특히 수많은 서버에 자동으로 배포하는 Bamboo와 같은 도구 운영시 주기적인 암호 변경은 피하고 싶은 업무가 된다.

이 문서는 암호를 사용하지 않고 보다 안전한 방법으로 ssh를 접속하는 방법을 정리한다.

ssh key 등록

1. Client와 remote server에 ssh 설치
2. 공개키 생성

공개키 생성 명령

```
ssh-keygen -t rsa
```

3. 공개키 파일 저장 위치 지정 (default 경로($HOME/.ssh)에 저장)

```
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): <>
Created directory '/root/.ssh'.
```

4. 비밀번호 지정 (자동 로그인을 원한다면 생략)

```
Enter passphrase (empty for no passphrase): < passphrase >
Enter same passphrase again:< passphrase >
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:tf64eem0ikpekRqymP1+IVRh9iIc3X5MMPrMTuOVV5o root@discourse.i.curvc.com
The key's randomart image is:
+---[RSA 2048]----+
|      ..+oo.     |
|     . +oo...    |
|      o.o.oo    .|
|      .. B..o. + |
|    ... S B.o E  |
|   + o.o.* o .   |
|  o o o...+ ..   |
|     + ... =o.   |
|     .=o. ==+    |
+----[SHA256]-----+
```

5. 생성 파일 확인

```
ls -al ~/.ssh/
total 8
drwx------. 2 root root   38 Mar 31 21:36 .
dr-xr-x---. 5 root root  185 Mar 31 21:36 ..
-rw-------. 1 root root 1766 Mar 31 21:36 id_rsa
-rw-r--r--. 1 root root  408 Mar 31 21:36 id_rsa.pub
```

6. .ssh 디렉토리 권한설정(권장)

```
chmod 700 ~/.ssh
chmod 600 ~/.ssh/id_rsa
chmod 644 ~/.ssh/id_rsa.pub
```

7. Remote server에 public key (id_rsa.pub) 등록하기
    a. ssh-copy-id 명령어 사용

```
ssh-copy-id -i <ssh public key> <username>@<server ip>
) ssh-copy-id -i .ssh/id_rsa.pub wastest@10.100.1.10
```

    b. scp 명령어 사용
        i. Remote server에 public key (id_rsa.pub) 등록하기

```
scp $HOME/.ssh/id_rsa.pub <ip or domain>:/home/<user id>/id_rsa.pub

#
id_rsa.
pub
100%  408    36.0KB/s   00:00
```

        ii. Remote server로 이동 후 id_rsa.pub 파일을 authorized_keys 파일에 추가

```
cat $HOME/id_rsa.pub >> $HOME/.ssh/authorized_keys
```

        만약, ssh 폴더가 없다면 다음 명령어 실행

```
mkdir ~/.ssh
```

8. Remote server 접속 확인하기

```
ssh <user>@<ip or domain>:/destination/path
```

만약 id_rsa 파일을 $HOME/.ssh/id_rsa에 만들지 않고 다른 디렉토리에 만들었다면 -i 옵션을 사용하여 ssh key 파일 지정

```
ssh -i $HOME/auth <ip or domain>:/destination/path
```

최초 접속 시 yes → passphrase 입력 (private key에 passphrase 입력하지 않았으면 바로 로그인)

```
The authenticity of host '10.0.20.30 (10.0.20.30)' can't be established.
ECDSA key fingerprint is SHA256:tPUIUgPalizfp/Vf9d1O/krJkNGbHC8rDr9Shonj55A.
ECDSA key fingerprint is MD5:ea:43:9f:1b:33:12:5f:4f:c8:0e:7b:7b:63:91:ed:db.
Are you sure you want to continue connecting (yes/no)? yes
```

특정 ip만 접속 허용 구성 방법

1. 특정 ip접속 허용

```
vi /etc/hosts.allow


#  sshd: ip


# hosts.allow   This file contains access rules which are used to
#               allow or deny connections to network services that
#               either use the tcp_wrappers library or that have been
#               started through a tcp_wrappers-enabled xinetd.
#
#               See 'man 5 hosts_options' and 'man 5 hosts_access'
#               for information on rule syntax.
#               See 'man tcpd' for information on tcp_wrappers
#
sshd:10.0.20.30 #  IP
sshd:10.0.20/24 # 10.0.20.*
```

2. 특정 ip제외한 모든 접속 막기

```
vi /etc/hosts.deny

#   sshd:ALL

# hosts.deny    This file contains access rules which are used to
#               deny connections to network services that either use
#               the tcp_wrappers library or that have been
#               started through a tcp_wrappers-enabled xinetd.
#
#               The rules in this file can also be set up in
#               /etc/hosts.allow with a 'deny' option instead.
#
#               See 'man 5 hosts_options' and 'man 5 hosts_access'
#               for information on rule syntax.
#               See 'man tcpd' for information on tcp_wrappers
#
sshd:ALL
```

3. sshd 서비스 재시작

```
systemctl restart sshd
```

4. 접속 차단 테스트

```
[root@localhost ~]# ssh root@10.1.22.0
ssh_exchange_identification: read: Connection reset by peer
```