리눅스 SSH 서버 구축 상세 설명

■ SSH 서버 구축 명령어 및 상세 설명

1. uname -a 입력

uname은 시스템의 정보를 출력하는 명령어입니다. 여기서 -a 옵션은 시스템의 전체 정보를 출력해주는 옵션입니다. uname -a를 입력하여 도출되는 예문과 그에 대한 설명은 다음과 같습니다.

출력 내용(예문)	
Linux edu-b4 3.10.0-1160.42.2.el7.x86_64 #1 SMP Tue Sep 7 14:49:57 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux	
필드 내용	**설명**
Linux	운영체제(커널)의 이름
edu-b4	리눅스 서버의 호스트 이름
3.10.0-1160.42.2.el7.x86_64	운영체제 릴리즈 버전 이름
#1 SMP Tue Sep 7 14:49:57 UTC 2021	운영체제 버전 및 출시 시간
x86_64 x86_64 x86_64	순서대로 다음과 같은 의미를 지님 - 시스템 하드웨어 타입 정보 - 프로세서 정보 - 하드웨어 플랫폼 정보
GNU/Linux	운영 체제에 대한 정보

2. sed 명령어로 sshd_config 내용을 변경

sed란 Stream Editor의 약자로써 원본 텍스트 파일을 편집하는 유용한 명령어다. 일반적으로 텍스트 파일 편집은 vi 편집기를 사용하여 여러 가지 vi 명령어로 편집 창을 통해 추가, 삭제, 변경 등의 편집을 하게 된다. 또한, 작업이 다 끝나게 되면 저장하거나 종료하게 된다. 주목할 점은 파일을 저장할 경우, 원래의 파일을 변경하여 저장하기 때문에 원본이 변경된다는 것이다. 또한, vi 편집기는 실시간으로 저장할 수 있다.

반면에, sed 명령어는 원본을 건드리지 않고 편집하기 때문에 작업이 완료되었어도 기본적으로 원본에는 전혀 영향이 없다. 물론 sed 명령어의 옵션으로 -i 지정하여 원본을 수정할 수는 있다. 즉, 원칙적으로 원본을 수정하지 않기에 sed 명령어를 사용할 경우 원본을 따로 보유할 수 있는 저장 공간이 필요하게 된다. 리눅스는 sed 명령어 수행을 위해 저장 공간으로 패턴 버퍼와 홀드 버퍼라는 2개의 버퍼를 제공하고 있다.

해당 단계에서는 sed 명령어를 사용하여 sshd_config의 내용을 변경하고 있다. 즉, ssh 서버의 환경을 세팅하기 위해 sed 명령어로 sshd_config 파일의 내용을 수정하고 이를 sshd_config.tmp 파일에 저장하는 단계인 것이다. 여기서 sed 명령어의 뒷부분에 /etc/ssh/sshd_config > /etc/ssh/sshd_config.tmp을 작성된 것을 확인할 수 있다. 이는 원본 파일(/etc/ssh/sshd_config)은 그대로 두고, 수정된 내용은 sshd_config.tmp로 저장한다는 것을 의미한다. 해당 단계에서 sed 명령어를 통해 작성한 내용과 설명은 다음과 같다.

사용 코드	
sed -e "s/#Port 22/Port 20022/g" -e "s/#Protocol/Protocol/g" -e "s/PermitRootLogin yes/PermitRootLogin no/g" -e "s/Subsystem sftp₩t₩/user₩/libexec₩/openssh₩/sftp-server/#Subsystem sftp₩t₩/usr₩/libexec₩/openssh₩/sftp-server/g" /etc/ssh/sshd_config > /etc/ssh/sshd_config.tmp	
설명	
1	"s/#Port 22/Port 20022/g"
	문자열 #Port 22를 찾아 Port20022로 변경
2	"s/#Protocol/Protocol/g"
	문자열 #Protocol을 찾아 Protocol로 변경
3	"s/PermitRootLogin yes/PermitRootLogin no/g"
	문자열 PermitRootLogin yes를 찾아 PermitRootLogin no로 변경
4	"s/Subsystem sftp₩t₩/user₩/libexec₩/openssh₩/sftp-server/#Subsystem sftp₩t₩/usr₩/libexec₩/openssh₩/sftp-server/g"
	Subsystem의 경로 문자열을 찾아 앞에 '#'을 입력하여 주석처리
5	/etc/ssh/sshd_config > /etc/ssh/sshd_config.tmp
	/etc/ssh/sshd_config의 내용을 수정하며 저장은 /etc/ssh/sshd_config.tmp의 이름으로 저장
-e 옵션은 여러개의 sed 명령을 사용해야 할 때 사용하는 옵션입니다. 각 구문은 –e "s/이전문자열/바꿀문자열/g'의 형태로 구성돼 있습니다.	

3. mv 명령어를 통해 파일의 이름을 변경

mv 명령어를 활용 기존의 sshd_config 파일을 sshd_config.backup 파일로, sshd_config.tmp 파일을 sshd_config 파일로 변경하는 작업을 수행하는 단계이다. 즉, 기존의 sshd_config 파일은 백업 파일로 쓰고 수정된 내용이 들어간 파일인 sshd_config.tmp 파일은 sshd 설정 파일로 쓰기 위해 수행하는 단계라고 할 수 있다. 이를 위해 작성한 명령어는 아래와 같다.

```
==========================================================================
mv /etc/ssh/sshd_config /etc/ssh/sshd_config.backup
mv /etc/ssh/sshd_config.tmp /etc/ssh/sshd_config
==========================================================================
```

4. 모든 계정에 대해서 접근 금지 설정(/etc/hosts.deny)

SSH 서버에 접근하는 모든 IP 주소를 차단하기 위한 목적으로 수행하는 단계이다. 이때, etc 디렉터리의 hosts.deny 파일을 vi 편집기를 통해 수정하게 되며 내용으로는 'sshd: ALL'을 작성하게 된다.

5. 접속 기록 열람을 통해 서버 접근을 허용할 IP 주소 파악

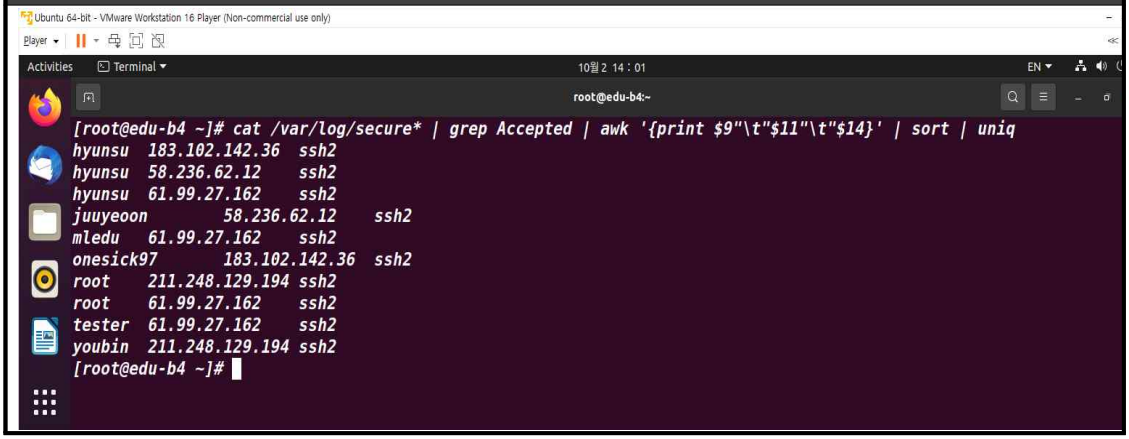
리눅스에서는 ssh 서버에 접근한 IP 주소를 파악할 수 있는 명령어를 보유하고 있다. 즉, 해당 명령어를 통해 접속 기록을 열람하고 허용할 IP 주소를 파악하게 된다. 단, 접근을 허용해야 하는 IP 주소가 접속 기록에

출력되지 않을 수 있으므로 이를 염두에 두고 접근을 허용할 IP 주소를 조사해야 한다. 해당 단계에서 사용한 명령어는 다음과 같다.

로그인에 성공한 IP 주소 목록을 출력하는 구문

```
cat /var/log/secure* | grep Accepted | awk '{print $9"₩t"$11"₩t"$14}' | sort | uniq
```

명령어 실행 시 출력 화면



6. 서버 사용자들의 IP 주소에 대한 접근 허용(/etc/hosts.allow)

앞선 단계에서 우리는 SSH 서버에 접근하는 모든 IP 주소를 차단하였다. 따라서 특정 사용자(팀원)의 서버 접근을 허용하기 위해 별도의 작업을 필요로 하게 된다. 해당 단계를 이를 위한 단계로써 이를 위해 vi 편집기로 /etc/hosts.allow 파일을 수정하게 된다. 이때, 작성되는 내용으로는 'sshd: 아이피 주소'로 작성하게 되며 작성된 IP 주소에 한해서 ssh 접속을 허용해주게 된다.

7. sshd 서버 재시작

서버에 대한 설정이 완료되어 서버를 재가동하는 단계이다. 이때 코드는 systemctl restart sshd의 형태로 작성하게 된다. 만약에 서버를 재시작하지 않는다면 앞서 변경한 sshd 환경 설정 내용이 적용되지 않기 때문에 반드시 서버를 재시작해주어야 한다.

8. 방화벽 설치 및 실행, 그리고 상태 조회

FTP 서버를 구축하는 과정에서 FTP 서버에서 사용하는 포트 번호를 방화벽 설정을 통해 개방해주지 않으면 FTP 서버에 접근할 수가 없게 된다. 따라서 방화벽 설정을 해주어야 하지만 기본적인 리눅스 시스템에서는 방화벽이 존재하지 않는다. 따라서 yum install firewalld -y를 통해 firewalld 패키지를 설치해주어야 한다. 이때 다른 패키지와 마찬가지로 yum을 활용하게 된다. 이후 설치가 완료되면 systemctl start firewalld와 systemctl status firewalld를 차례대로 입력하고 실행과 상태 조회를 수행하여 방화벽 설치가 잘 완료되었는지 확인하는 과정을 거치게 된다.

9. 방화벽 설정을 통해 FTP 서버에 필요한 포트를 개방

방화벽이 잘 실행되는 것을 확인했으면 FTP 서버에 필요한 포트를 개방하는 작업을 수행해야 한다. 이 과정에서 작성되는 코드는 다음과 같으며, 각 코드는 지정한 포트에 대해 영구적으로 개방한다는 의미를 지니고 있다.

```
================================================================================
firewall-cmd --zone=public --permanent --add-port=20020/tcp
================================================================================
```

10. 호스트 이름 설정 및 기타 네트워크 설정(/etc/sysconfig/network)

호스트 이름 설정 및 네트워크에 대한 설정을 진행하는 단계이다. 이때, etc에 있는 sysconfig 디렉터리에 속한 network 파일을 vi 편집기로 내용을 수정하며 설정을 진행하게 된다. 해당 파일에 작성된 내용과 의미

작성된 내용	
NOZEROCONF=yes NETWORKING=yes NETWORKING_IPV6=no HOSTNAME=edu-b4	
코드	**설명**
NOZEROCONF=yes	DHCP 환경이 없는 네트워크에서 Peer To Peer 연결이나 Wireless 환경에서 관리자의 수동적인 설정 없이 네트워킹 작업을 수행하기 위한 프로토콜인 Zero Configuration Networking을 사용하지 않을지를 작성하는 부분이다. yes를 작성할 경우 Zero Configuration Networking 프로토콜을 사용하지 않게 된다.
NETWORKING=yes	네트워킹 연결을 허용할지를 작성하는 부분이다. 우리는 서버 구동을 위해 네트워킹 연결이 필요하므로 yes 작성하였다. (IPV4 기준)
NETWORKING_IPV6=no	IPV6을 통한 네트워킹 연결을 허용할지를 작성하는 부분이다. 우리는 IPv6를 통한 네트워킹 연결을 사용하지 않으므로 no를 작성하였다.
HOSTNAME=edu-b4	서버의 호스트 이름을 설정하는 부분이다. 해당 부분의 값으로는 우리 조에 부여한 서버 이름으로 지정해주었다.

는 다음과 같다.

11. 호스트 이름 설정 - 2

앞선 단계에서 우리는 etc에 있는 sysconfig 디렉터리에 속한 network 파일을 vi 편집기로 내용을 수정하면서 호스트 이름을 기재해주었다. 그러나 CentOS에서는 network 파일을 변경하는 것 외에 hostnamectl이라는 명령어를 사용하여 호스트 이름을 설정하는 별도의 단계를 거쳐야 한다. 즉, 우리는 edu-b4로 호스트 이름을 바꿔줄 것이므로 hostnamectl set-hostname edu-b4라는 구문을 작성하여 호스트 이름을 변경하게 된다. 단, 호스트 이름의 변경은 로그인했을 시점부터 적용되므로 적용된 것을 확인하기 위해서는 별도의 재로그인 과정이 필요하게 된다.