SSH 빠른 참조

연결, 키, 설정, 터널, SCP, SFTP

연결

기본 연결

```
ssh user@host                    # 호스트에 연결
ssh -p 2222 user@host            # 커스텀 포트
ssh user@host command             # 원격 명령 실행
ssh -t user@host "top"           # TTY 할당 강제
```

연결 플래그

`-p port`	특정 포트에 연결
`-i key`	특정 신원 (개인키) 사용
`-t`	가상 터미널 할당 강제
`-v / -vv / -vvv`	상세 디버깅 (점점 상세해짐)
`-q`	조용한 모드 (경고 억제)
`-N`	원격 명령 없음 (터널용)
`-f`	명령 전에 백그라운드로 이동
`-J jump`	점프 호스트 (ProxyJump)

키 관리

키 생성

```
ssh-keygen -t ed25519 -C "you@example.com"
ssh-keygen -t rsa -b 4096 -C "you@example.com"
ssh-keygen -t ed25519 -f ~/.ssh/mykey
ssh-keygen -p -f ~/.ssh/id_ed25519  # 패스프레이즈 변경
```

공개키 배포

```
ssh-copy-id user@host
ssh-copy-id -i ~/.ssh/mykey.pub user@host
# 수동: .pub을 원격 authorized_keys에 추가
cat ~/.ssh/id_ed25519.pub | ssh user@host \
  "mkdir -p ~/.ssh && cat >> ~/.ssh/authorized_keys"
```

키 파일

`~/.ssh/id_ed25519`	개인키 (비밀 유지)
`~/.ssh/id_ed25519.pub`	공개키 (자유롭게 공유)
`~/.ssh/authorized_keys`	원격: 허용된 공개키
`~/.ssh/known_hosts`	알려진 호스트 지문

설정 파일

~/.ssh/config 기본

```
Host myserver
  HostName 192.168.1.100
  User deploy
  Port 2222
  IdentityFile ~/.ssh/deploy_key

# 이후 간단히 연결:
# ssh myserver
```

유용한 설정 옵션

```
Host *
  ServerAliveInterval 60
  ServerAliveCountMax 3
  AddKeysToAgent yes
  IdentitiesOnly yes

Host bastion
  HostName bastion.example.com
  User admin
```

설정 지시어

`Host`	항목의 별칭 패턴
`HostName`	실제 호스트명 또는 IP
`User`	로그인 사용자명
`Port`	원격 포트 (기본 22)
`IdentityFile`	개인키 경로
`ProxyJump`	다른 호스트를 통해 점프
`ServerAliveInterval`	킵얼라이브 간격 (초)
`IdentitiesOnly`	지정된 키만 사용

포트 포워딩

로컬 포워딩 (-L)

```
# 원격 포트 5432에 로컬 포트 5432로 접근
ssh -L 5432:localhost:5432 user@host
# ssh 호스트를 통해 remote-db:3306에 접근
ssh -L 3306:remote-db:3306 user@host
# 모든 인터페이스에 바인딩
ssh -L 0.0.0.0:8080:localhost:80 user@host
```

원격 포워딩 (-R)

```
# 로컬 포트 3000을 원격 포트 8080에 노출
ssh -R 8080:localhost:3000 user@host
# 모든 인터페이스에서 원격 연결 허용
ssh -R 0.0.0.0:8080:localhost:3000 user@host
```

동적 포워딩 (-D)

```
# 로컬 포트 1080에 SOCKS5 프록시
ssh -D 1080 user@host
# 백그라운드 SOCKS 프록시
ssh -D 1080 -fN user@host
```

SCP 및 SFTP

SCP (보안 복사)

```
scp file.txt user@host:/remote/path/
scp user@host:/remote/file.txt ./local/
scp -r dir/ user@host:/remote/path/
scp -P 2222 file.txt user@host:/path/
```

SFTP (대화형 전송)

```
sftp user@host
# sftp 세션 내:
# put local.txt    — 파일 업로드
# get remote.txt   — 파일 다운로드
# ls / lcd / cd    — 목록 / 디렉토리 변경
```

전송 플래그

`-r`	재귀 (디렉토리 복사)
`-P port`	포트 지정 (SCP는 -P, -p 아님)
`-C`	압축 활성화
`-l limit`	대역폭 제한 Kbit/s
`-i key`	특정 신원 파일 사용

에이전트 포워딩

SSH 에이전트

```
eval "$(ssh-agent -s)"           # 에이전트 시작
ssh-add ~/.ssh/id_ed25519        # 에이전트에 키 추가
ssh-add -l                       # 로드된 키 목록
ssh-add -D                       # 모든 키 제거
```

에이전트 포워딩

```
ssh -A user@host                 # 에이전트 포워딩
# 또는 ~/.ssh/config에서:
# Host myserver
#   ForwardAgent yes
```

에이전트 참고

에이전트 포워딩을 통해 원격 호스트가 키를 복사하지 않고 로컬 키를 사용할 수 있습니다. 신뢰할 수 있는 호스트에만 사용하세요. 가능하면 에이전트 포워딩보다 ProxyJump를 사용하세요.

터널

영구 터널

```
# 계속 열려 있는 백그라운드 터널
ssh -fNT -L 5432:localhost:5432 user@host
# 자동 재연결 터널 (autossh 사용)
autossh -M 0 -fNT -L 5432:localhost:5432 user@host
```

점프 호스트 / 배스천

```
ssh -J bastion user@internal-host
ssh -J user1@hop1,user2@hop2 user@target
# 설정 등등:
# Host internal
#   HostName 10.0.0.5
#   ProxyJump bastion
```

터널 관리

- `-f` 인증 후 백그라운드
- `-N` 원격 명령 없음
- `-T` 가상 터미널 비활성화
- `~.` 멈춘 SSH 세션 종료 (이스케이프)
- `~C` 포워딩을 위한 명령줄 열기
- `~#` 포워딩된 연결 목록

문제 해결

연결 디버깅

```
ssh -vvv user@host               # 최대 상세도
ssh -G user@host                 # 설정 덤프 (드라이런)
ssh-keyscan host                 # 호스트 키 가져오기
ssh-keygen -R host               # known_hosts에서 제거
```

일반 문제

`Permission denied`	잘못된 키, 사용자, 또는 ~/.ssh 권한 (700/600)
`Host key changed`	ssh-keygen -R host 후 재연결
`Connection timed out`	방화벽, 포트, 호스트 접근성 확인
`Too many auth failures`	-i로 키 지정하거나 IdentitiesOnly 사용
`Broken pipe`	설정에 ServerAliveInterval 추가

파일 권한

```
chmod 700 ~/.ssh
chmod 600 ~/.ssh/id_ed25519      # 개인키
chmod 644 ~/.ssh/id_ed25519.pub # 공개키
chmod 600 ~/.ssh/authorized_keys
chmod 644 ~/.ssh/known_hosts
```

보안 모범 사례

서버 강화

`PasswordAuthentication no`	비밀번호 로그인 비활성화
`PermitRootLogin no`	root SSH 접근 비활성화
`AllowUsers deploy`	허용 사용자 화이트리스트
`Port 2222`	비기본 포트 (스캐너 회피)
`MaxAuthTries 3`	인증 시도 횟수 제한

키 관행

Ed25519 키를 사용하세요 (더 작고, 빠르고, 더 안전).
항상 개인키에 패스프레이즈를 설정하세요.
패스프레이즈 재입력을 피하려면 ssh-agent를 사용하세요.
주기적으로 키를 교체하고, 미사용 키를 폐기하세요.
어떤 키를 제공할지 제어하려면 IdentitiesOnly를 사용하세요.

다중화

연결 공유

```
# ~/.ssh/config에서
Host *
  ControlMaster auto
  ControlPath ~/.ssh/sockets/%r@%h-%p
  ControlPersist 600

# 소켓 디렉토리 생성
mkdir -p ~/.ssh/sockets
```

다중화 이점

같은 호스트에 대한 여러 SSH 세션에 단일 TCP 연결을 재사용합니다.
반복적인 핸드셰이크를 제거해 더 빠른 연결과 낮은 오버헤드를 제공합니다.
ControlPersist는 마스터를 활성 상태로 유지합니다 (초 단위).

이스케이프 시퀀스

SSH 이스케이프 명령어

- `~.` 연결 종료 (멈춘 세션 종료)
- `~^Z` SSH 세션 일시 중단
- `~C` 명령줄 열기 (포워딩 추가)
- `~#` 포워딩된 연결 목록
- `~&` SSH 백그라운드 (연결 대기)
- `~?` 이스케이프 도움말 표시
- `~~` 리터럴 물결표 전송