

Riferimento rapido SSH

Connessioni, chiavi, configurazione, tunnel, SCP, SFTP

Connessione

Connessione base

```
ssh user@host # connect to host
ssh -p 2222 user@host # custom port
ssh user@host command # run remote command
ssh -t user@host "top" # force TTY allocation
```

Flag di connessione

```
-p porta Connessione a una porta specifica
-i chiave Usa una specifica identità (chiave privata)
-t Forza l'allocazione di uno pseudo-terminale
-v / -vv / -vvv Debug dettagliato (livello crescente)
-q Modalità silenziosa (sopprime avvisi)
-N Nessun comando remoto (per i tunnel)
-f Vai in background prima del comando
-J jump Host di salto (ProxyJump)
```

Gestione delle chiavi

Generazione di chiavi

```
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 # change passphrase
```

Distribuzione della chiave pubblica

```
ssh-copy-id user@host
ssh-copy-id -i ~/.ssh/mykey.pub user@host
# Manual: append .pub to remote authorized_keys
cat ~/.ssh/id_ed25519.pub | ssh user@host \
"mkdir -p ~/.ssh && cat >> ~/.ssh/authorized_keys"
```

File delle chiavi

```
~/.ssh/id_ed25519 Chiave privata (da tenere segreta)
~/.ssh/id_ed25519.pub Chiave pubblica (da condividere liberamente)
~/.ssh/authorized_keys Remoto: chiavi pubbliche accettate
~/.ssh/known_hosts Fingerprint degli host conosciuti
```

File di configurazione

Nozioni di base su ~/.ssh/config

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

# Then connect with just:
# ssh myserver
```

Opzioni di configurazione utili

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

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

Direttive di configurazione

```
Host Pattern alias per la voce
HostName Hostname o IP reale
User Nome utente di accesso
Port Porta remota (default 22)
IdentityFile Percorso alla chiave privata
ProxyJump Transita attraverso un altro host
ServerAliveInterval Intervallo keep-alive (secondi)
IdentitiesOnly Usa solo le chiavi specificate
```

Port forwarding

Forwarding locale (-L)

```
# Access remote port 5432 via local port 5432
ssh -L 5432:localhost:5432 user@host
# Access remote-db:3306 through ssh host
ssh -L 3306:remote-db:3306 user@host
# Bind to all interfaces
ssh -L 0.0.0.0:8080:localhost:80 user@host
```

Forwarding remoto (-R)

```
# Expose local port 3000 on remote port 8080
ssh -R 8080:localhost:3000 user@host
# Allow remote connections from any interface
ssh -R 0.0.0.0:8080:localhost:3000 user@host
```

Forwarding dinamico (-D)

```
# SOCKS proxy on local port 1080
ssh -D 1080 user@host
# Background SOCKS proxy
ssh -D 1080 -fN user@host
```

SCP e SFTP

SCP (copia sicura)

```
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 (trasferimento interattivo)

```
sftp user@host
# Inside sftp session:
# put local.txt - upload file
# get remote.txt - download file
# ls / lcd / cd - list / change directory
```

Flag di trasferimento

```
-r Ricorsivo (copia directory)
-P porta Specifica la porta (SCP usa -P, non -p)
-C Abilita la compressione
-l limite Limite di banda in Kbit/s
-i chiave Usa un file identità specifico
```

Agent forwarding

SSH Agent

```
eval "$(ssh-agent -s)" # start agent
ssh-add ~/.ssh/id_ed25519 # add key to agent
ssh-add -l # list loaded keys
ssh-add -D # remove all keys
```

Forwarding dell'agent

```
ssh -A user@host # forward agent
# Or in ~/.ssh/config:
# Host myserver
# ForwardAgent yes
```

Note sull'agent

L'agent forwarding permette all'host remoto di usare le chiavi locali senza copiarle. Usare solo con host fidati. Preferire ProxyJump rispetto all'agent forwarding quando possibile.

Tunnel

Tunnel persistente

```
# Background tunnel that stays open
ssh -fNT -L 5432:localhost:5432 user@host
# Auto-reconnecting tunnel (with autossh)
autossh -M 0 -fNT -L 5432:localhost:5432 user@host
```

Jump host / Bastion

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

Gestione dei tunnel

```
-f Vai in background dopo l'autenticazione
-N Nessun comando remoto
-T Disabilita lo pseudo-terminale
~ Termina la sessione SSH bloccata (escape)
~C Apre la riga di comando per il forwarding
~# Elenca le connessioni inoltrate
```

Risoluzione dei problemi

Debug della connessione

```
ssh -vvv user@host # max verbosity
ssh -G user@host # dump config (dry run)
ssh-keyscan host # fetch host keys
ssh-keygen -R host # remove from known_hosts
```

Problemi comuni

```
Permission denied Chiave, utente o permessi ~/.ssh errati (700/600)
Host key changed Esegui ssh-keygen -R host, poi riconnetti
Connection timed out Controlla firewall, porta e raggiungibilità dell'host
Too many auth failures Usa -i per specificare la chiave o IdentitiesOnly
Broken pipe Aggiungi ServerAliveInterval alla configurazione
```

Permessi dei file

```
chmod 700 ~/.ssh
chmod 600 ~/.ssh/id_ed25519 # private key
chmod 644 ~/.ssh/id_ed25519.pub # public key
chmod 600 ~/.ssh/authorized_keys
chmod 644 ~/.ssh/known_hosts
```

Riferimento rapido SSH

Buone pratiche di sicurezza

Hardening del server

PasswordAuthentication no	Disabilita l'accesso con password
PermitRootLogin no	Disabilita l'accesso SSH come root
AllowUsers deploy	Whitelist degli utenti autorizzati
Port 2222	Porta non predefinita (evita scanner)
MaxAuthTries 3	Limita i tentativi di autenticazione

Buone pratiche per le chiavi

Preferire le chiavi Ed25519 (più piccole, veloci e sicure).
Impostare sempre una passphrase sulle chiavi private.
Usare ssh-agent per evitare di reinserire la passphrase.
Ruotare periodicamente le chiavi; revocare quelle inutilizzate.
Usare IdentitiesOnly per controllare quale chiave viene offerta.

Multiplexing

Condivisione della connessione

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

# Create socket directory
mkdir -p ~/.ssh/sockets
```

Vantaggi del multiplexing

Riutilizza una singola connessione TCP per più sessioni SSH verso lo stesso host. Elimina l'handshake ripetuto — connessioni più veloci e overhead ridotto.
ControlPersist mantiene il master attivo (secondi).

Sequenze di escape

Comandi di escape SSH

~.	Termina la connessione (chiude sessioni bloccate)
~^Z	Sospende la sessione SSH
~C	Apri la riga di comando (aggiunge forwarding)
~#	Elenca le connessioni inoltrate
~&	Mette SSH in background (in attesa di connessioni)
~?	Mostra la guida degli escape
~~	Invia una tilde letterale