SSH

Kontrolle aus der Ferne

@innaytool [Yannick Bungers]
innay@foss-ag.de

7. Juli 2017

FOSS AG

Was ist SSH?

- Kommando und Tool-Suite
- Sichere Verbindung zum Server (z.B. Fernwartung) über eine Shell
- `telnet` sollte gerade im Internet nicht mehr eingesetzt werden, da unverschlüsselt
- Verschlüsselte Version von `rsh`, `rcp`, `rlogin`, `rexec`

Geschichte

- **1995:** Proprietäres Projekt von SSH Communications Security Corp. (SSH-1) $\rightarrow$ Sicherheitsmängel
- **1996:** Zweite Version (SSH-2) $\rightarrow$ inkompatibel zu SSH-1
- **1999:** Abspaltung der Open Source Variante (OpenSSH)
- **2005:** Neue Version (SSH G3) proprietär
- **2006:** IETF RFC 4250 und andere $\rightarrow$ siehe Man-Page

Einführung

- `ssh <server>`
  Verbindet mit dem aktuellen Benutzernamen zum Server auf dem Standardport
- `ssh -p 22 user@server`
  Verbindet mit dem angegebenen Benutzernamen auf dem angegebenen Port
- Nachfrage nach Key des Servers diesen hinzuzufügen.

Software

- openssh auf den meisten Linux-Distributionen vorinstalliert
- Auch Teil von proprietären Unixoiden
- Alle BSDs
- Auch auf Mac (auch sshd)
- putty für Windows (nicht nur ssh)

SFTP

- `sftp [-p <port>] <server>`
- Benutzung wie ein FTP-client
- `put`, `get`, `cd`, `ls`, ...
- Als Dateibrowsererweiterung
- FileZilla (Windows, Mac OS X)

SCP

- `scp <user>@<host>:<dir> <user2>@<host2>:<dir>`
- `scp -r <user>@<host>:<dir> <user2>@<host2>:<dir>`
- Benutzung wie `cp` nur mit zusätzlichen Host
- Oder eine lokale Datei

SSHFS

- `sshfs <server>:<ordner1> <ordner2>`
- Einbinden eines Ordners als Dateisystem mit ssh-Verbindung
- `sshfs -o reconnect,ServerAliveInterval=15, ServerAliveCountMax=3 <server>:<ordner1> <ordner2>`
- `reconnect`: Verbindet sich nach einer Verbindungsunterbrechung wieder
- `ServerAliveInterval`: Zeit bis ein Aufrechterhaltungspaket gesendet wird
- `ServerAliveCountMax`: Zahl der verloren Pakete

remote execution

- `ssh <server> <Terminal-Anwendung>`
- `ssh -X <server> <Desktop-Anwendungen>`
- Normaler ssh-Befehl mit zusätzlichem Parameter für das auszuführende Programm
- Schlechte Performance $\rightarrow$ X2GO

SOCKS5 Proxy

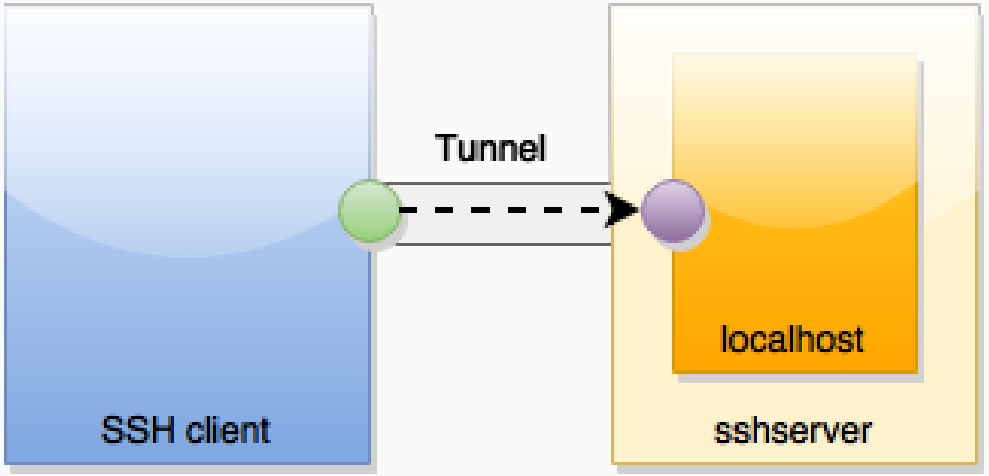
- `ssh -N -D <port> <server>` → SOCKS5 Proxy
- Tunnel zu einem Server für beliebige Ports
- Ähnlich zu einem VPN
- Proxy

Local Single Port Tunnel

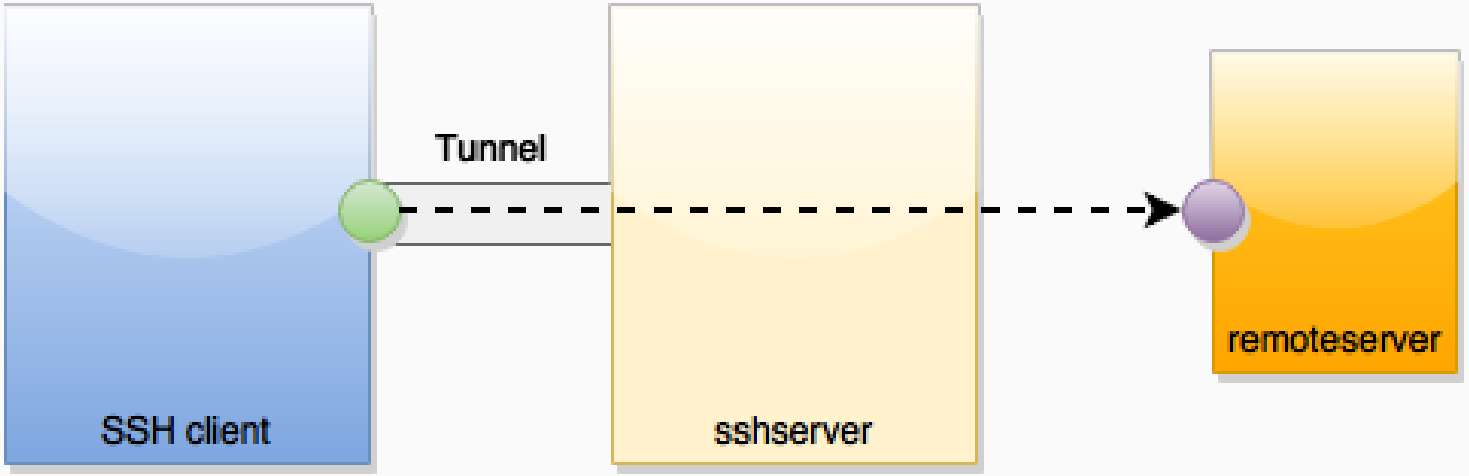
`ssh -N -L <BindAdresse>:<BindPort>:`
`<ZielAdresse>:<ZielPort> <jumpServer>`

Auf dem lokalen Rechner wird der Port und die Adresse gebunden

ssh -L 123:localhost:456 sshserver



ssh -L 123:remoteserver:456 sshserver

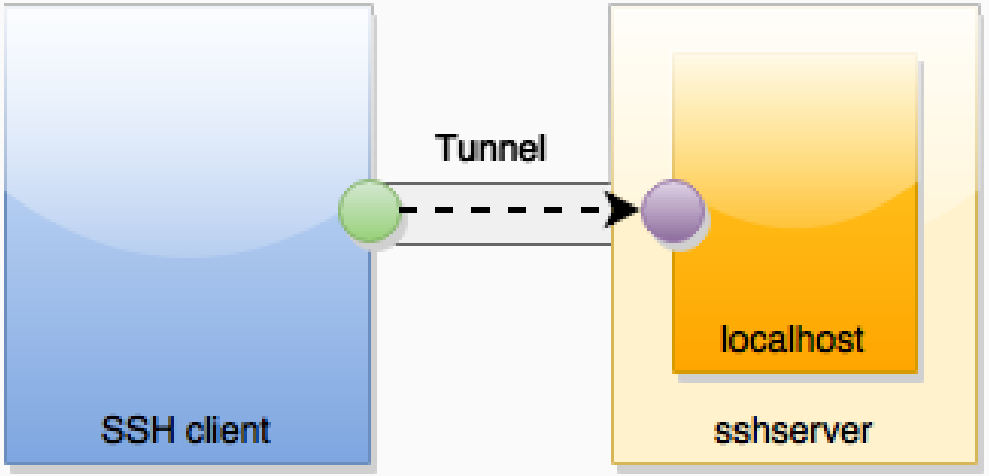


Remote Single Port Tunnel

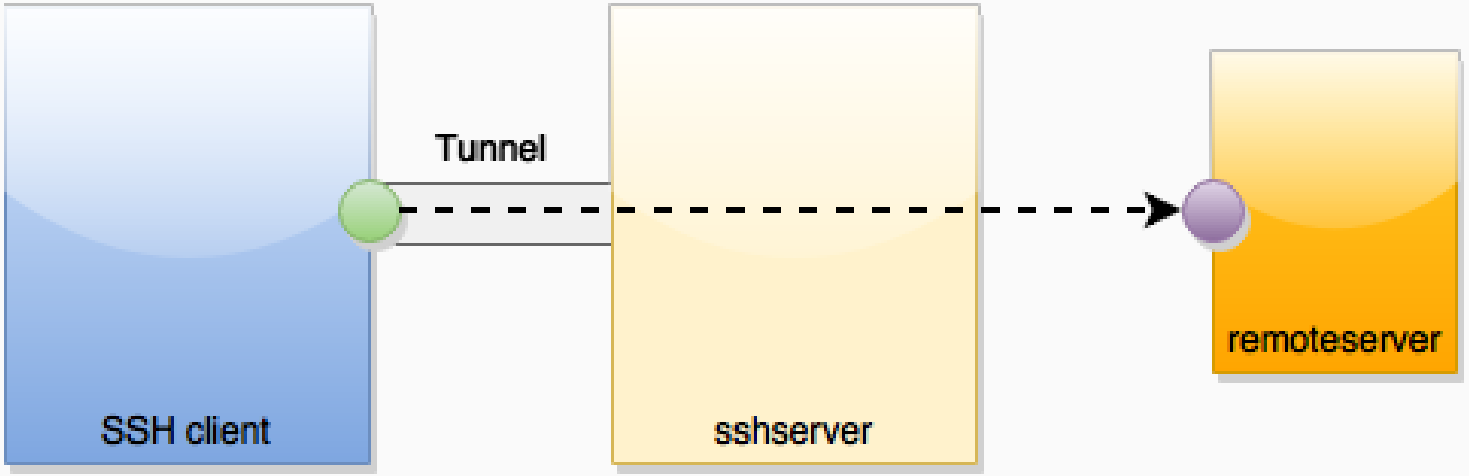
`ssh -N -R <BindAdresse>:<BindPort>:`
`<ZielAdresse>:<ZielPort> <jumpServer>`

Auf dem entfernten Rechner wird die Addresse und der Port gebunden

ssh -L 123:localhost:456 sshserver



ssh -L 123:remoteserver:456 sshserver



Server

- Konfiguration `/etc/ssh/sshd_config`
- Standardport 22
- Root Login (de)aktivieren
- Module (de)aktivieren [`sftp`, Verschlüsselungsalgorithmen, ...]

SSH Keys

- `ssh-keygen`
- `ssh-keygen -t <Typ>`
  Erstellt einen neuen Key vom angegebenen Typ
- `ssh-keygen -l -E <Typ> -f <Datei>`
  Berechnet den Hash-Wert der Key-Datei mit dem angegeben Algorithmus
- `E`: Hash-Typ
- `l`: Fingerprint
- `f`: Datei
- `ssh-keyscan`
- `ssh-copy-id`

Zusammenfassung

- Weite Verbreitung
- `scp`, `sftp`, `sshfs`
- remote-execution (mit und ohne X forwarding)
- Porttunnel
- SSH-Server