

OpenSSH

OpenSSH - Tipps & Tricks

Andreas Schiermeier

Was erwartet euch?

- Möglichkeiten von OpenSSH
- Protokollüberblick (SSHv2)
- Konfiguration
- Serverlogin
- Public Key Authentisierung
- SSH Agent
- Tunnelbau
- Dateiübertragung mit sftp automatisieren
- Firewalls
- Zeichensatzprobleme

Möglichkeiten von OpenSSH

- gesicherte Netzwerkkommunikation
- Fernverwaltung
 - Streamumleitung
- flexible Benutzerauthentifizierung
- Dateitransfer

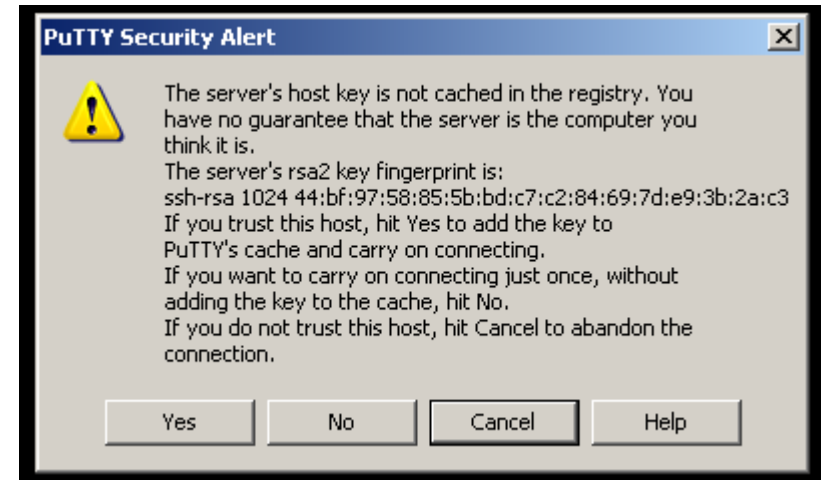
Protokollüberblick (SSHv2)

- Transport Layer
- User Authentication Layer
- Connection Layer

Transport Layer

- Verschlüsselung
- Integritätsprüfung
- Komprimierung

```
as@movingmind:~> ssh 127.0.0.1
The authenticity of host '127.0.0.1 (127.0.0.1)' can't be established.
ECDSA key fingerprint is 70:c1:ae:b1:bb:a3:30:c6:b0:cf:1a:ee:30:d5:d5:35.
Are you sure you want to continue connecting (yes/no)? █
```



User Authentication Layer

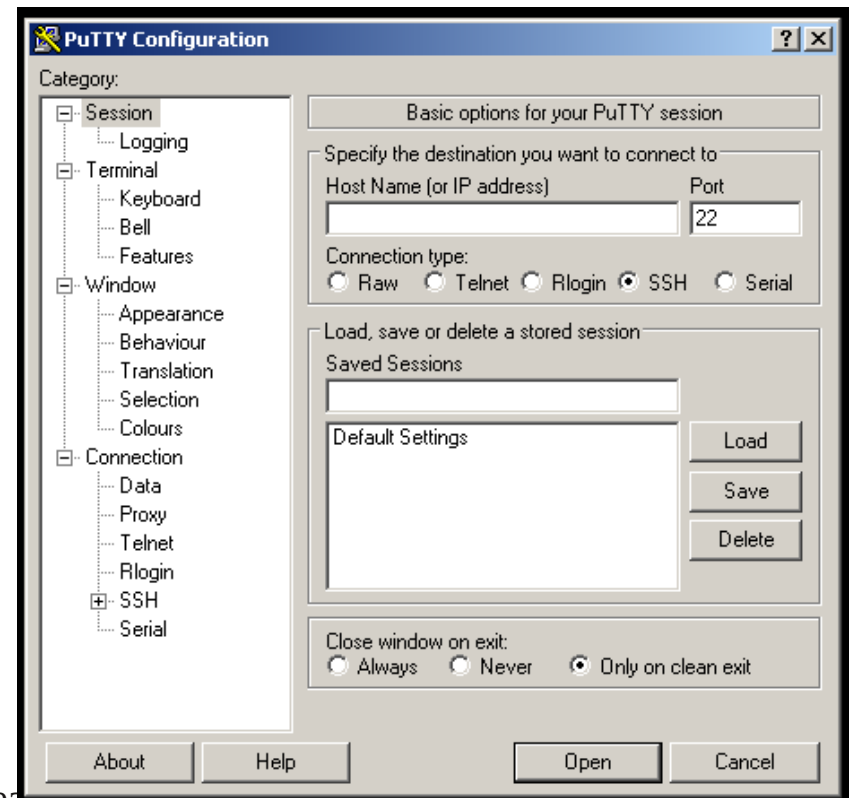
- Benutzerauthentifizierung mit verschiedenen Methoden
 - password
 - Challenge Response (keyboard-interactive)
 - publickey
 - GSSAPI, Kerberos

Connection Layer

- Aufteilung der Verbindung in virtuelle Kanäle (Shell, Portweiterleitungen)
- Kanäle können mehrmals vorhanden sein (auch Shell)
- X11-, Agent- & Tunnelweiterleitungen

Konfiguration

- Server (man sshd_config)
 - /etc/ssh/sshd_config
- Client (man ssh_config)
 - CLI
 - ~/.ssh/config
 - /etc/ssh/ssh_config



Konfiguration

- Match: abweichende Konfiguration nach
 - Systembenutzer (User)
 - Systemgruppe (Group)
 - Clientadresse (Host)
 - angesprochene Serveradresse (Address)

z.B.

```
Match User as
```

```
ForceCommand /bin/date
```

- Port umkonfigurieren: Port 22022

Konfiguration - Client

- Kürzel für Serverzugriff
- Verbindungsspezifische Konfiguration
z.B.

```
Host meinsrv
```

```
Hostname mein.server.de
```

```
Host *.meinefirma.de
```

```
ForwardAgent yes
```

```
Port 22022
```

Serverlogin

- PermitRootLogin

- yes

- no

- without-password

- forced-commands-only

- /root/.ssh/authorized_keys

- command="rdiff-backup --server --restrict-read-only /",no-port-forwarding,no-X11-forwarding,no-agent-forwarding,from="127.6.9.3" ssh-dss AAAAB3Nza...*

- UsePAM

- In Verbindung mit PermitRootLogin

- PasswordAuthentication no

- ChallengeResponseAuthentication no

Serverlogin

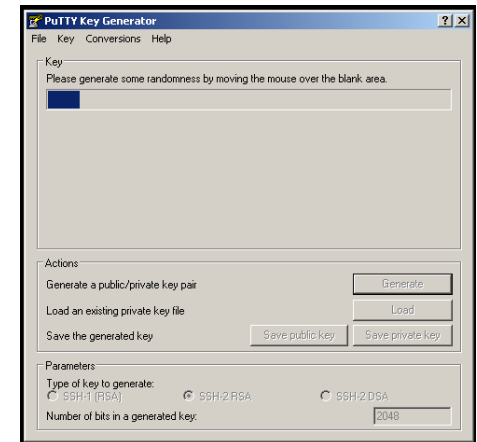
- Abhilfe bei aufgebrauchten Verbindungen im Rahmen von Brute-force-Angriffen

MaxStartups (Standard 10)

LoginGraceTime (Standard 120 Sekunden)

Public Key Authentisierung

- Vorteil: Passwortloser Login auf Serversystemen
- Key erzeugen: `ssh-keygen -b 2048 -t dsa`
 - Key unterhalb von `~/.ssh`
 - Windows: `puttygen`
- Key verteilen: `ssh-copy-id benutzername@mein.server.de`
 - manuell: auf dem Server in `~/.ssh/authorized_keys` (Berechtigungen!)
- Fingerprint auslesen: `ssh-keygen -l -f /pfad/zum/privatekey`



Public Key Authentisierung

- zusätzliche Möglichkeiten in `~/.ssh/authorized_keys`
 - `command`, erzwungenes Kommando, `ForceCommand` in `sshd_config`
 - `from`, Zugriff auf IP- & Adressbereiche einschränken
 - `no-port-forwarding`
 - `permitopen`
 - `tunnel`
- zwischenzeitlich auch Aufbau einer CA-Infrastruktur möglich (aber nicht X.509)

SSH Agent

- Ermöglicht bequemen Umgang mit Passphrase-geschützten Key-Einsatz
- ssh-agent im Hintergrund
 - `SSH_AGENT_PID`, **`SSH_AUTH_SOCK`**
 - Key(s) entsperren und im Agent hinterlegen: `$ ssh-add`
 - Verwendung von Keys nur nach Bestätigung: `$ ssh-add -c`
 - Keys aus Agent entfernen: `$ ssh-add -D`
 - Agentforwarding: `$ ssh -A benutzer@mein.server.de`
 - in `~/.ssh/config`: `ForwardAgent yes`
 - Bei Verdacht auf kompromittiertes System, Agentforwarding deaktivieren: `$ ssh-add -a benutzer@mein.server.de`
- Putty: pageant & Verknüpfung zu PPK im Autostart-Ordner

Tunnelbau

- lokaler Listener, Verbindung zu Port im entfernten Netz

```
$ ssh -L 8080:192.168.1.1:80 benutzer@mein.server.de
```
- entfernter Listener, Verbindung zu Port im lokalen Netz

```
$ ssh -R 3128:proxy.meinefirma.de:3128 benutzer@mein.server.de
```
- SOCKS Proxy

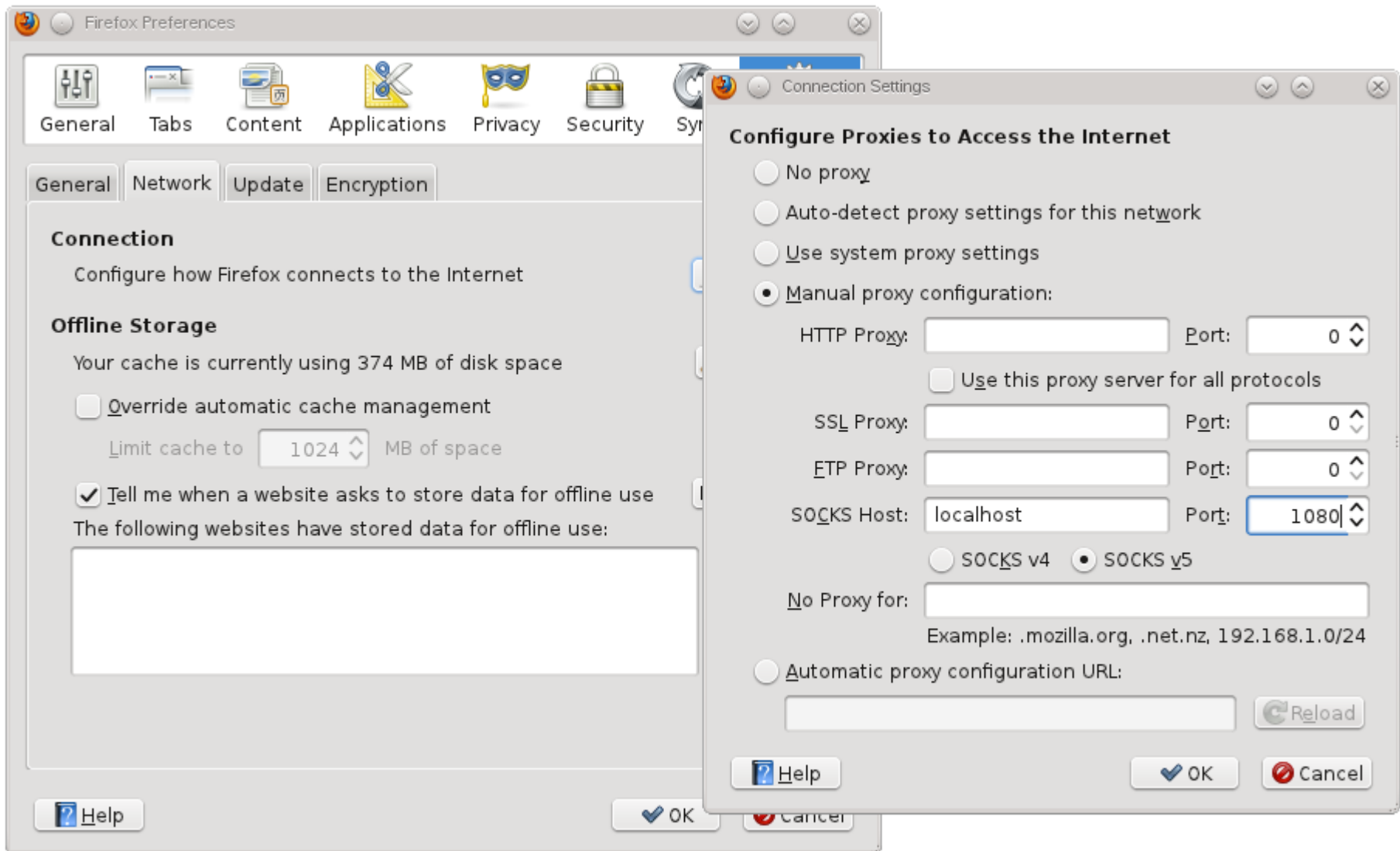
```
$ ssh -D1080 benutzer@mein.server.de
```
- TUN-Interface

```
# ssh -w any root@mein.server.de
```
- PPP over SSH

```
# pppd persist noauth nodetach silent 172.16.0.1:172.16.0.2  
pty "ssh -t root@mein.server.de pppd noauth nodetach"
```

 - vorweg mit Host-Route (z.B. übers Defaultgateway) Erreichbarkeit von mein.server.de sicherstellen

SOCKS-Konfiguration in Firefox



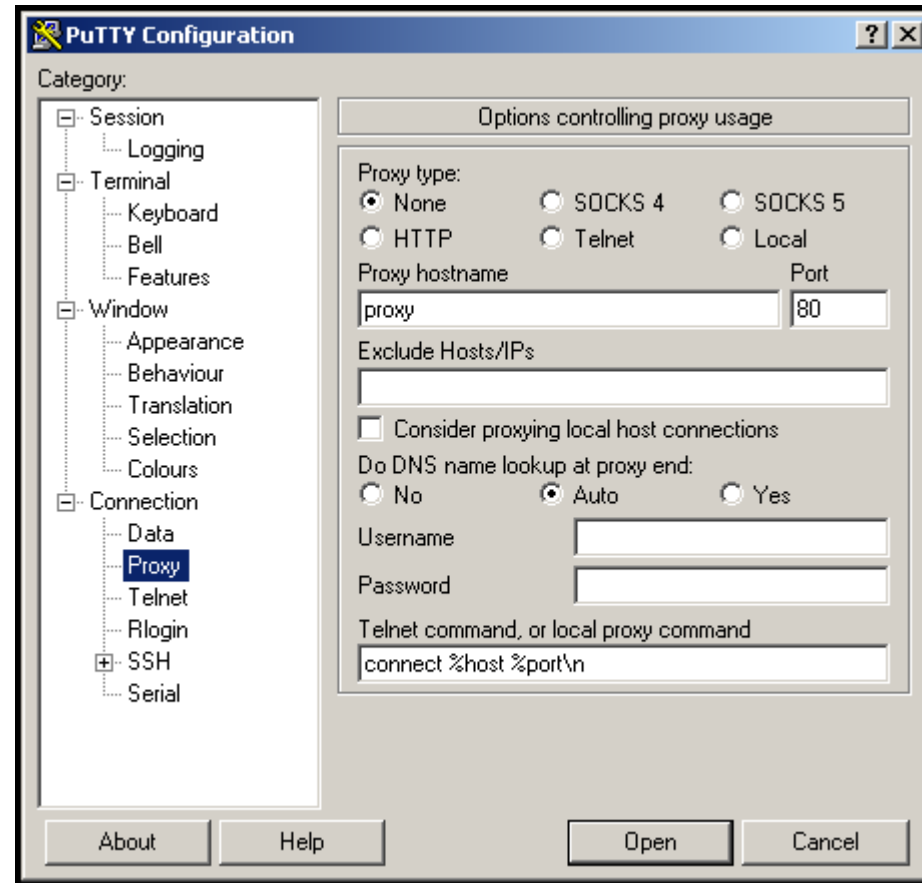
Dateiübertragung mit sftp automatisieren

```
$ cat <<'EOF' > transfer.txt
! mv datei1.txt datei1.txt.transferring
put datei1.txt.transferring
datei1.txt.transferring
rename datei1.txt.transferring datei1.txt
! mv datei1.txt.transferring
datei1.txt.transferred
EOF
$ sftp -b transfer.txt benutzer@mein.server.de
```

Firewalls

- Connect via HTTP-Proxys hinweg
 - <http://proxytunnel.sourceforge.net/> => Froscon Slides!
~/.ssh/config
Host *
ProtocolKeepAlives 15
ProxyCommand proxytunnel -p proxyserver:8080
-u proxyuser -s proxypasswort -d %h:%p
 - HTTP CONNECT auf Port 22 nicht zugelassen?
 - iptables -t nat -A PREROUTING -d \$serverip -p tcp -m tcp --dport 443 -j REDIRECT --to-port 22
 - <http://www.nocrew.org/software/httptunnel.html>

Proxykonfiguration in Putty



Firewalls aus Adminsicht

- SSH eingehend erlaubt? Auch die Server im Griff?
 - /etc/ssh/sshd_config
 - PermitOpen
 - PermitTunnel
 - AllowTcpForwarding
 - GatewayPorts
- SSH ausgehend erlaubt? Auch die Clients im Griff?
- SSH nicht erlaubt?
 - HTTP-Weg abgesichert? DPI?

Zeichensatzprobleme

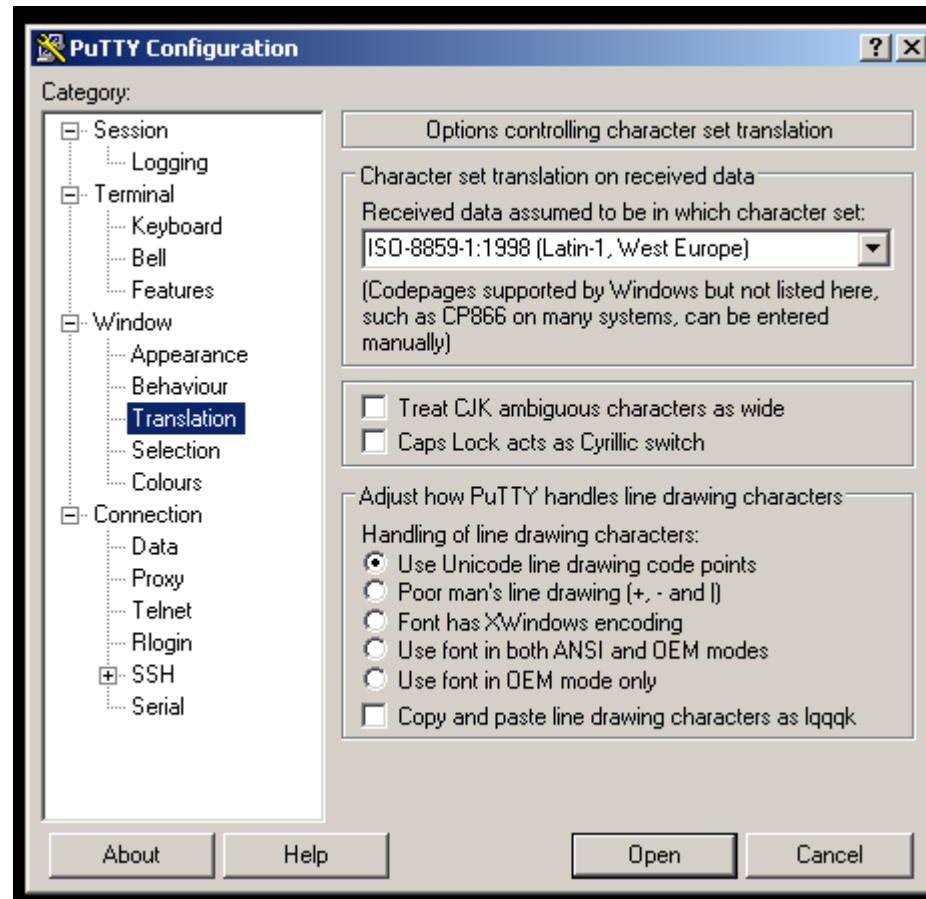
- Die Lösung* :-)

```
luit -encoding ISO8859-15 ssh  
benutzer@mein.server.de
```

*) Voraussetzungen

- lokales Environment (LC_CTYPE, LC_ALL, LANG) passt
 - z.B. LANG=de_DE.UTF-8
- entferntes Environment passt - z.B. LANG=C

Zeichensatzprobleme - putty



Danke!

- Fragen?
- Kritik?
- "Ich hab da noch was!"

E-Mail: andreas@schiermeier.name

Jabber: [aschiermeier@imfarkt.de](jabber:aschiermeier@imfarkt.de)