SSH Tunnel und Mehr

Kielux
2022Sep16 @ 11.00
Big Blue Button

der.hans (https://floss.social/@FLOX_advocate)
Database Support Engineering Manager
Object Rocket, a Rackspace Technology Company
https://www.ObjectRocket.com/

Ja, wir stellen ein :)

Rackspace Technologies
https://rackspace.jobs/careers/?location=null&search=germany

Wo und Wie Hans zu finden

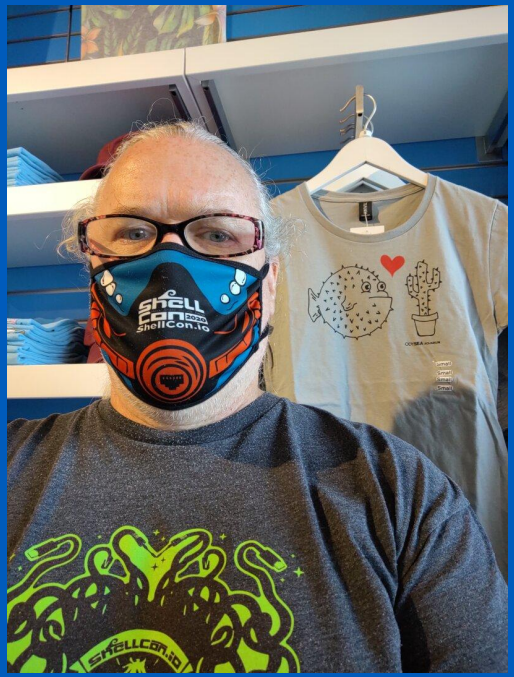
Meine Konferenzen und Vorträge

- https://www.SpiralArray.com/talks

sociale Medien und Fediverse

- FLOX_advocate auf Mastodon, https://floss.social/@FLOX_advocate
- LuftHans auf Libera.chat IRC, normalerweise in #SeaGL and #PLUGaz

SSH Forschungsreise



Vorausgesetzte Kennnisse

Grundlegende SSH Anwendung

Von SSH erforderte Zugangsrechte auf Dateisystem

Grundlegende Nutzung SSH Schlüssel und Fingerabdrücke

SSH

SSH == Secure SHell

OpenSSH ist eine Projekt von OpenBSD

Essentielles Werkzeug für Systemadministratoren sowie DevOps

Benötigt ein Konto auf dem Fernrechner

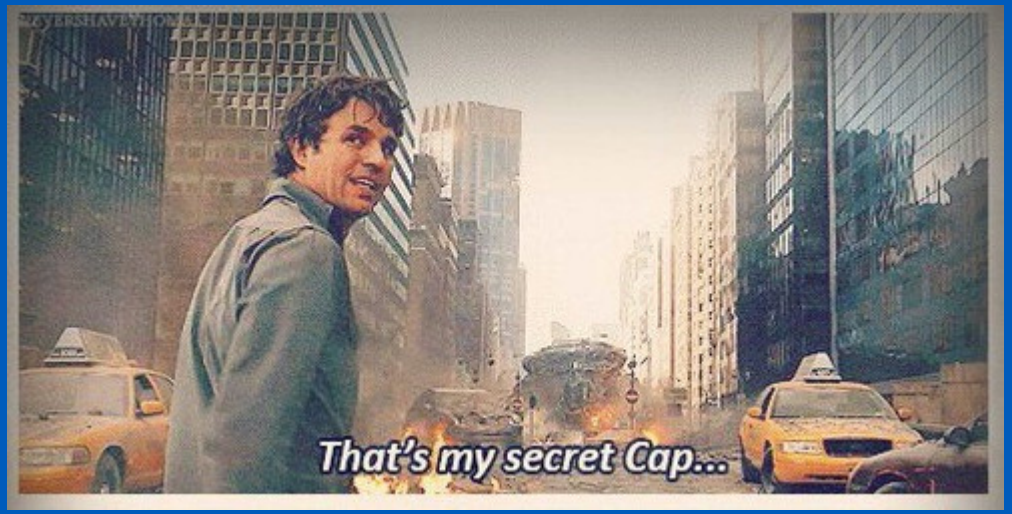
Ermöglicht sicheren Austausch von Dateien über verschlüsselte TCP Verbindungen

Stellt sichere, authentifizierte, verschlüsselte Verbindungen zwischen Computern über feindliche Netzwerke her

Ja, genau

Ja, ich habe "feindliche Netzwerke" gesagt

Es gibt ein Geheimnis



Das Internetz bleibt immer ein feindliches Netzwerk

einfache Verbindung

```
ssh -p 22 fernhost.beispiel.de /bin/bash
```

Profitieren

```
// include::sicherheitsvortrag.adoc[]
```

Ausführen und verlassen

```
$ date; ssh fernhost "hostname; sleep 1"; date
Fri Sep  2 09:48:34 PM UTC 2022

fernhost.beispiel.de
Fri Sep  2 09:48:35 PM UTC 2022
$
```

einfache Tunnel

```
ssh firewall.beispiel.de -L 2222:firewall.beispiel.de:22
```

In den Tunnel von einer anderen Shell

```
ssh -p 2222 localhost
```

```
scp -P 2222 -pr lokale_verz_um_zu_syncen localhost:
```

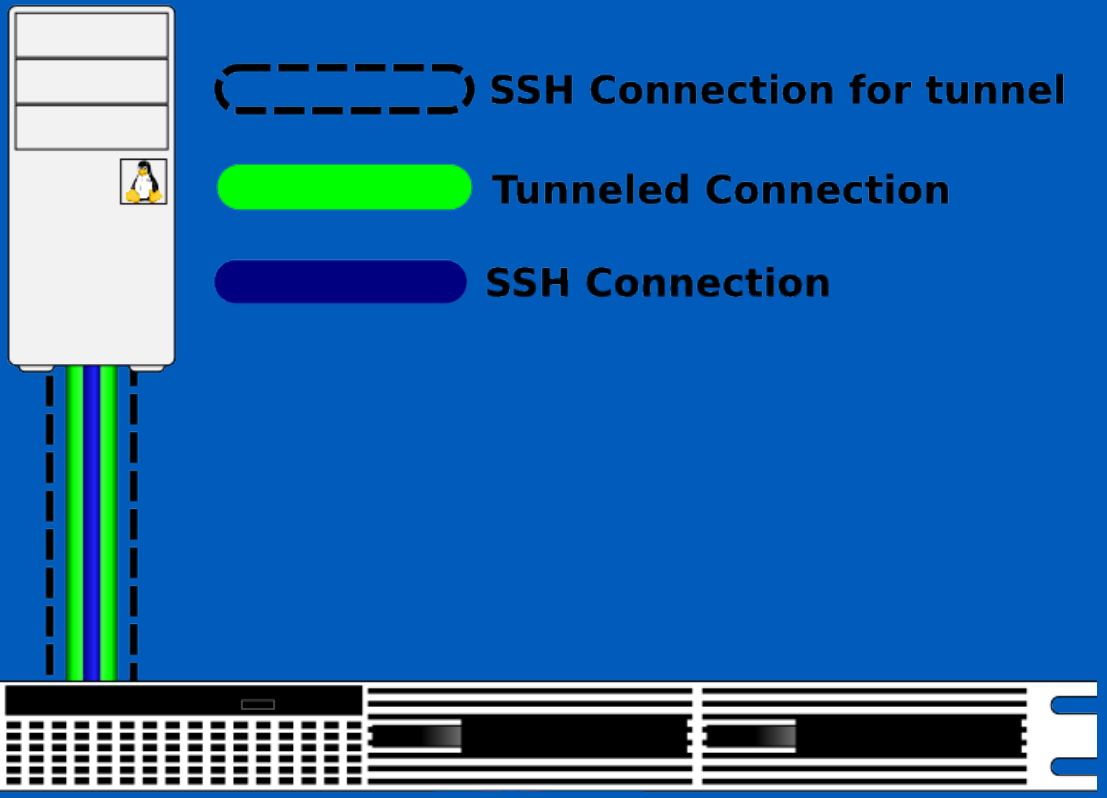
Tunneldiagramm

```
ssh -L 2222:firewall.beispiel.de:22 firewall.beispiel.de
```

```
ssh -p 2222 localhost
```

```
scp -P 2222 -pr localhost:fernverz_um_zu_syncen .
```

Your Machine



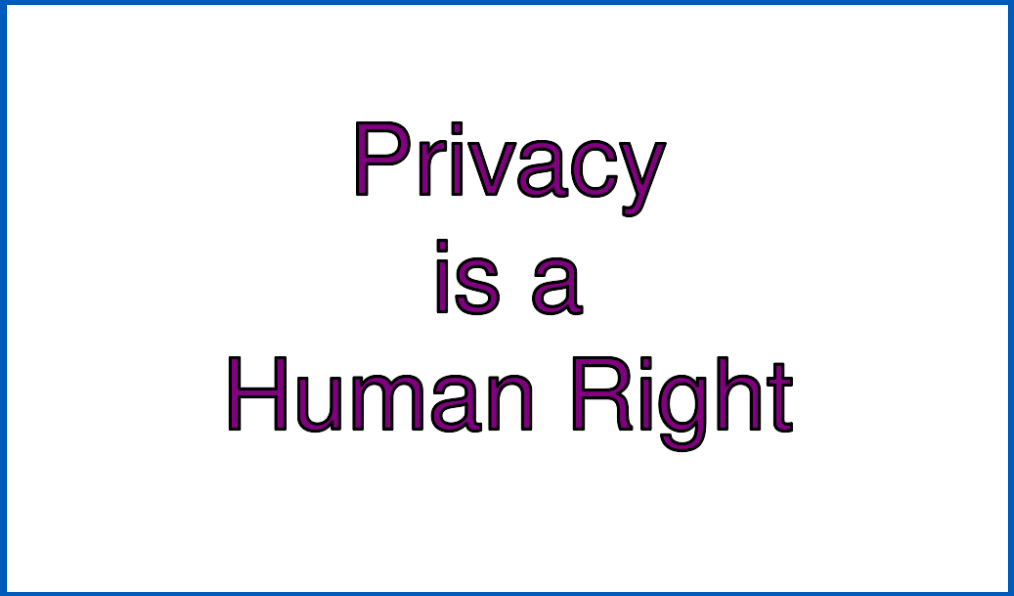
Gateway

Jetzt geht’s los

```
ssh -L 2222:127.0.0.1:22 demo
```

```
ssh -p 2222 localhost
```

Wieso ist SSH wichtig?



Tunneln aber rückwärts

```
ssh firewall.beispiel.de -R 2222:firewall.beispiel.de:22
```

in den Tunnel vom Fernhost

```
ssh -p 2222 localhost
```

Wo ist denn localhost?

hostname

```
ssh firewall.beispiel.de -R 2222:firewall.beispiel.de:22
```

localhost

```
ssh firewall.beispiel.de -R 2222:localhost:22
```

Tunneln über Fremdpartei

Umkehrtunnel von host2 erstellen

```
host2$ ssh -R 2222:firewall.beispiel.de:22 firewall.beispiel.de
```

Tunnel von host1 erstellen

```
host1$ ssh -L 2222:firewall.beispiel.de:2222 firewall.beispiel.de
```

```
host1$ ssh -p 2222 localhost
host2$
```

vereinfachter Bastiontunnel

```
host2$ ssh -R 2222:localhost:22 firewall.beispiel.de
```

```
host1$ ssh -L 2222:localhost:2222 firewall.beispiel.de
```

Doppelumkehr

```
host2$ ssh -L 3333:localhost:3333 -R 2222:localhost:22 firewall.beispiel.de
```

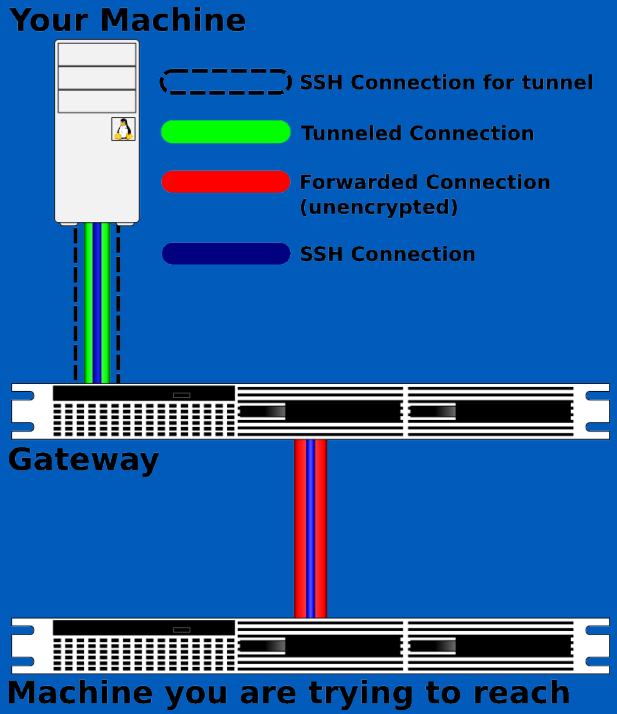
```
host1$ ssh -L 2222:localhost:2222 -R 3333:localhost:22 firewall.beispiel.de
```

Die Verbindung weiterleiten

```
ssh -N -f -L 3306:mysql.beispiel.de:3306 firewall.beispiel.de
```

```
$ grep 3306 /etc/services
mysql           3306/tcp
mysql           3306/udp
```

```
host1 <====verschlüsselt====> firewall/gateway <----UNverschlüsselt----> mysql
```



SOCKS Proxy / dynamischer Proxy

```
ssh -D 1080 firewall.beispiel.de
```

TIPP: FoxyProxy

graphisches Tunneln

```
schlepptop$ ssh -Y desktop.beispiel.de
```

```
desktop$ firefox -new-instance -ProfileManager presentations
```

Trennung über VM oder Container

```
desktop$ ssh -Y vm.beispiel.de
```

```
vm$ firefox -new-instance -ProfileManager javascriptIstGefaehrlich
```

Schlüssel

```
ssh-keygen -f .ssh/id_neu
```

authorized_keys

ssh-copy-id

Beispiel für Dienst: MySQL

```
ssh -N -f -L 3306:mysql.beispiel.de:3306 firewall.beispiel.de
```

```
mysql -h localhost -p 3306 --protocol=TCP
```

Bei MySQL lieber 127.0.0.1 anstatt localhost verwenden

```
mysql -h 127.0.0.1
```

Beispiel für Dienst: Email

```
ssh -L 2143:imap.beispiel.de:143 -L 2993:imap.beispiel.de:993 -L 2025:smtp.beispiel.de:25
firewall.beispiel.de
```

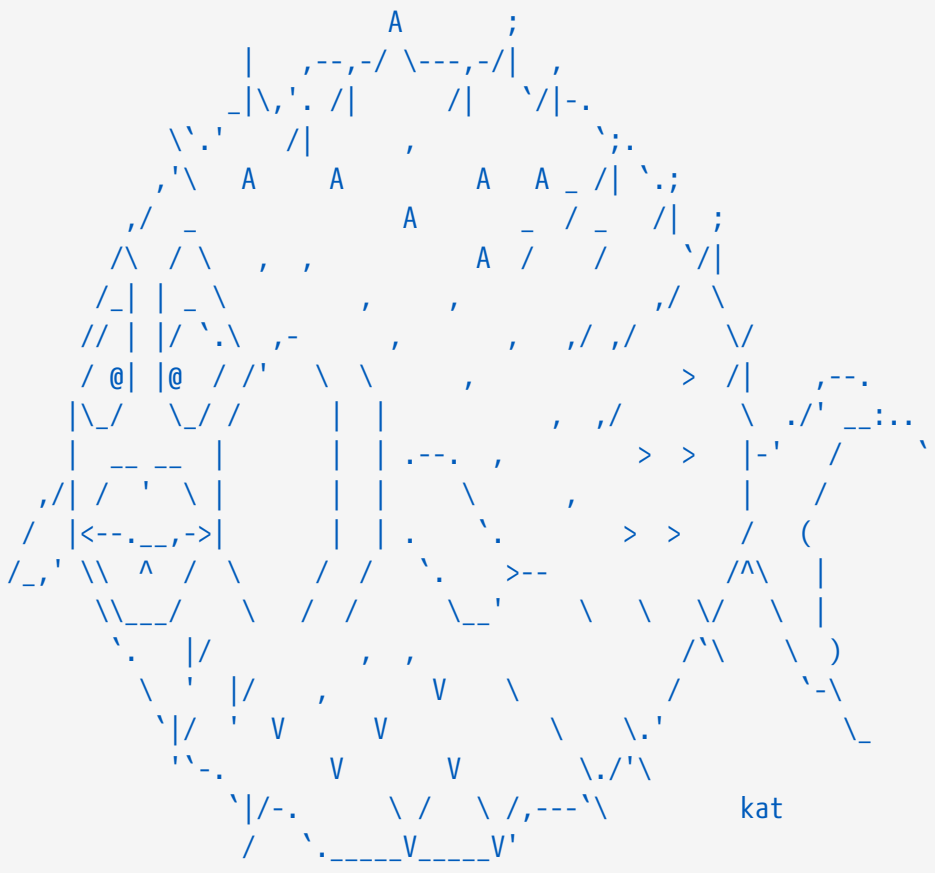
Beispiel für Dienst: Web

```
ssh -L 8080:www.beispiel.de:80 firewall.beispiel.de
```

```
links -http.extra-header "Host: www.beispiel.de" http://localhost:8080/
```

```
ssh -D 1080 firewall.beispiel.de
```

TIPP: FoxyProxy



Tipps für ~/.ssh/config

Mehrere known_hosts Dateien
z.B. eine für normale Verwendung und eine für Orchestrierung

```
UserKnownHostsFile ~/.ssh/known_hosts ~/.ssh/known_hosts_automagic
```

Tipps für ~/.ssh/authorized_keys

Beschränkungen pro Schlüssel im Eintrag authorized_keys
Wichtig für automatisierte Funktionen

```
no-port-forwarding,no-X11-forwarding,no-agent-forwarding,no-pty
```

Erfordere Befehl pro Schlüssel (auch bekannt als forced command)
der gegebene Befehl von SSH wird ignoriert

```
command="hostname"
command="echo ${SSH_ORIGINAL_COMMAND}" >>/tmp/was_wollten_sie_eigentlich.log
```

Zugangsbeschränkungen auf das Netzwerk pro Schlüssel festlegen

```
from="localhost,127.0.0.1"
```

Tipps für die Befehlszeile

-N == kein Fernbefehl (no remote command)
-f == Hintergrund nach Authentifizierung (background after authentication)

```
ssh -N -f -L 2222:localhost:22 firewall.beispiel.de
```

-v == Ausführlichkeit, maximal 3

-G == zeige bevorzugte Konfiguration

-t == erfordere pseudo-terminal

```
ssh -p 2222 localhost screen -x myScreen
```

-o == gib gewünschte Option für Konfigurationsdatei auf der Befehlszeile ein

```
ssh -o FingerprintHash=md5 firewall.beispiel.de
```

Shell Variablen

PS1

SSH_AUTH_SOCK

Werkzeuge

ssh-copy-id

sshfs ist abgekündigt

rsync

scp: jetzt als unsicher angesehen

sftp

autossh

Verwende ssh-keyscan nicht. ssh-keyscan prüft die Schlüssel nicht!

aus der ssh-keyscan man Seite

```
Falls mittels ssh-keyscan eine Datei »ssh_known_hosts« erstellt wird,
ohne die Schlüssel zu überprüfen, sind die Benutzer durch
Man-In-The-Middle -Angriffe verwundbar.
```

Noch was

```
ssh firewall.beispiel.de "sudo tar -C /etc cfz -" | tar -C /tmp xfz -
```

```
ssh firewall.beispiel.de "ps auxw" | tee /tmp/firewall_ps.txt | less
```

```
rsync -e ssh -avHS photos/ mybackupserver:photos/
```

SSH das Heimspiel

Gehet nun hin in Sicherheit!

Herzlichen Dank!

Credits

- SSH Home Game von Brian C, Nutzung genehmigt
- SSH Thrown Tunnel von Brian C, Nutzung genehmigt
- SSH Single Tunnel abgeänderte Version von Brian's Thrown Tunnel, Nutzung genehmigt
- Bruce Banner images: https://weheartit.com/entry/185262694
- Dierk S für seine Fehlerkorrektur

Ressourcen

- OpenSSH site
  - https://www.OpenSSH.com/
- FoxyProxy
  - https://getfoxyproxy.org/
- My Linux Journal article on SSH tunneling
  - https://www.LinuxJournal.com/magazine/use-ssh-cross-suspect-host-securely
- OpenSSH book
  - https://en.wikibooks.org/wiki/OpenSSH
- OpenBSD site (creators of OpenSSH)
  - https://www.openbsd.org/
- scp has issues, consider rsync
- Nixie Pixel's "Protect your Packets with SSH" video (good review of FoxyProxy)
  - https://www.youtube.com/watch?v=5mCNO_aL4BA