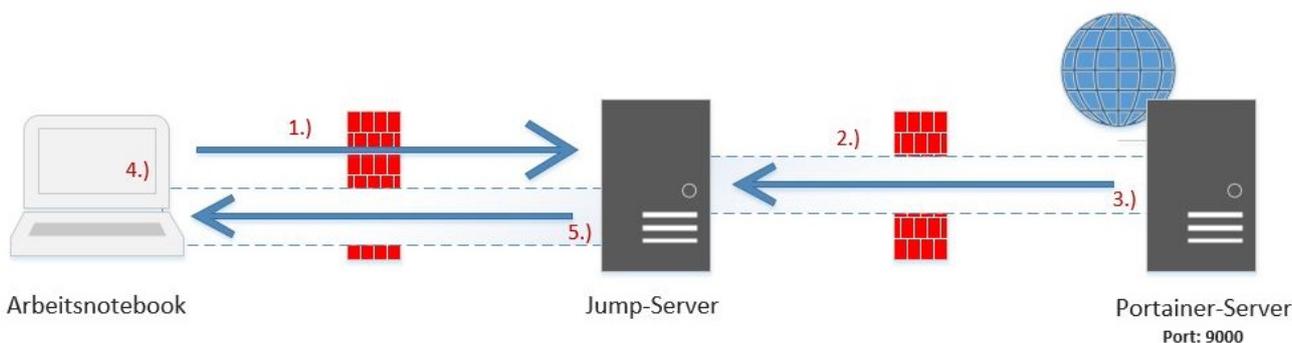


Tunnel Application over SSH

Wenn es Firewall-technisch nicht möglich ist, direkt auf eine bestimmte eingerichtete Applikation auf einem Applikationsservers zuzugreifen; kann versucht werden die Firewalls via SSH-Tunnel zu umgehen.

Als Beispiel habe ich hier den Portainer-Server, welcher in einem separaten Servernetzwerk steht und nur via SSH über den Jump-Server erreicht werden kann ausgewählt. Der Portainer-Server, respektive das Web-GUI läuft unter dem Port 9000. Gebraucht werden somit zwei Tunnel, in denen wir jeweils den Traffic des Portainer-Servers verschlüsselt durch den Jump-Host hindurch auf den localhost des Arbeitsnotebooks forwarden werden.

Vorgehen zum erstellen der benötigten Tunnel



1.) Aufbau einer neuen SSH Session auf den Jump-Hosts

2.) Erstellen des ersten Tunnels vom Jump-Host auf den Portainer Server

3.) Zwischen-test des ersten Tunnels

Um zu überprüfen, ob nun der erste Tunnel und die zeitgleich eingerichtete Port-Weiterleitung korrekt funktioniert, kann mithilfe eines CURLs bestimmen werden.

Dazu machen wir ein CURL auf den lokal weitergeleiteten Port und schauen, ob die Backend-Applikation auf dem Jump-Host ansprechbar ist.

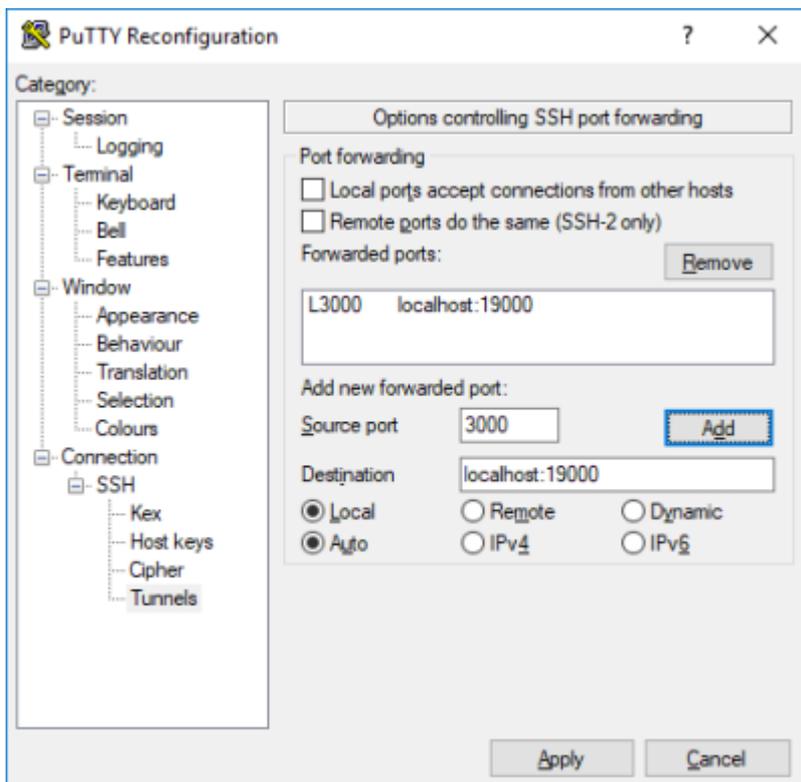
```
# curl 127.0.0.1:19000 | grep portainer
```

```
[rebermi@vnx1a ~]$ curl 127.0.0.1:19000 | grep portainer
% Total    % Received % Xferd  Average Speed   Time    Time     Time
Current
                                Dload  Upload  Total  Spent  Left
Speed
```

```
100 2748 100 2748 0 0 486k 0 --:--:-- --:--:-- --:--:--
536k
<html lang="en" ng-app="portainer">
  open: toggle && ['portainer.auth', 'portainer.init.admin',
'portainer.init.endpoint'].indexOf($state.current.name) === -1,
  nopadding: ['portainer.auth', 'portainer.init.admin',
'portainer.init.endpoint'].indexOf($state.current.name) > -1 ||
applicationState.loading
```

So wie es aussieht, funktioniert der erste Tunnel! ;)

4.) Aufbau einer neuen SSH Session auf den Jump-Hosts



5.) Aufbau einer neuen SSH Session auf den Jump-Hosts

Last update: **2018/08/23 13:30**