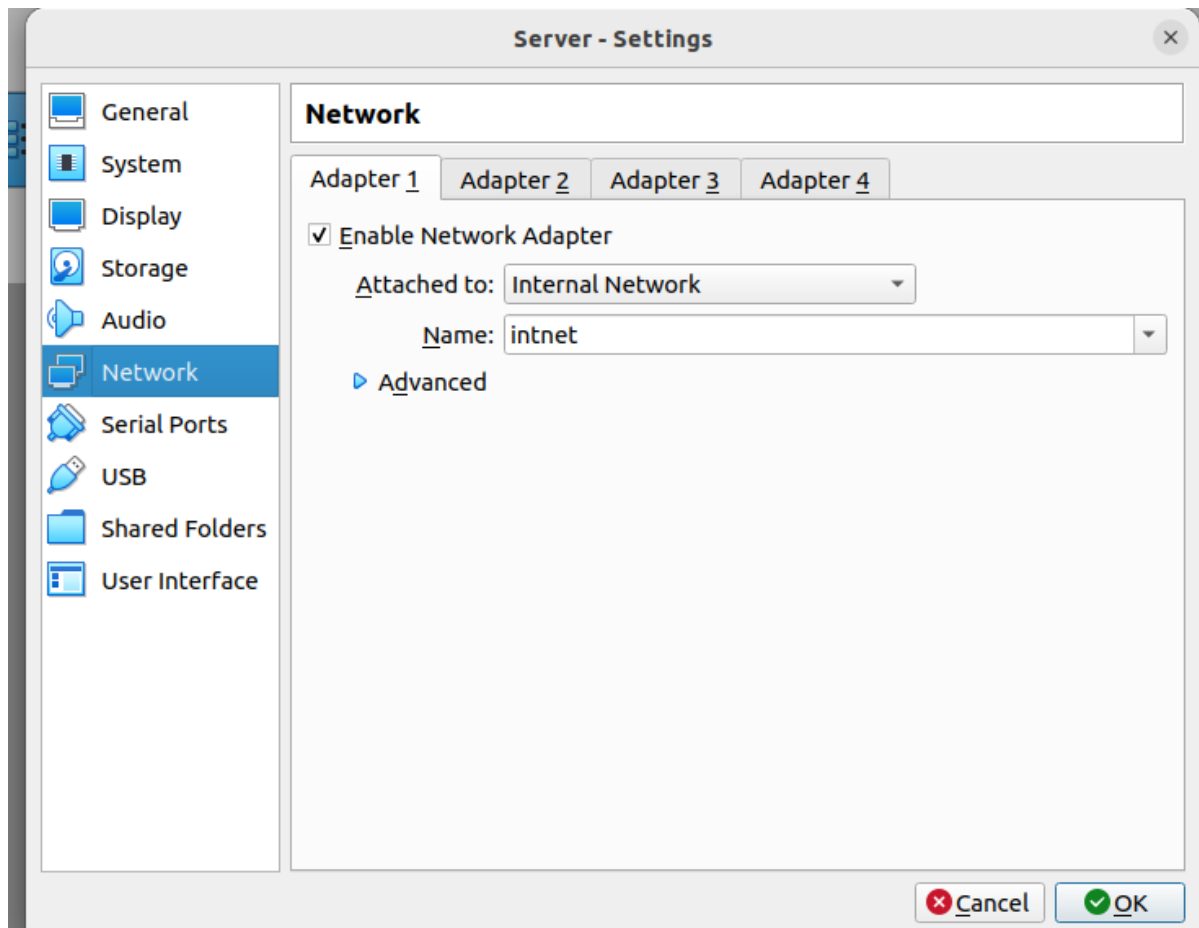


# Firewall

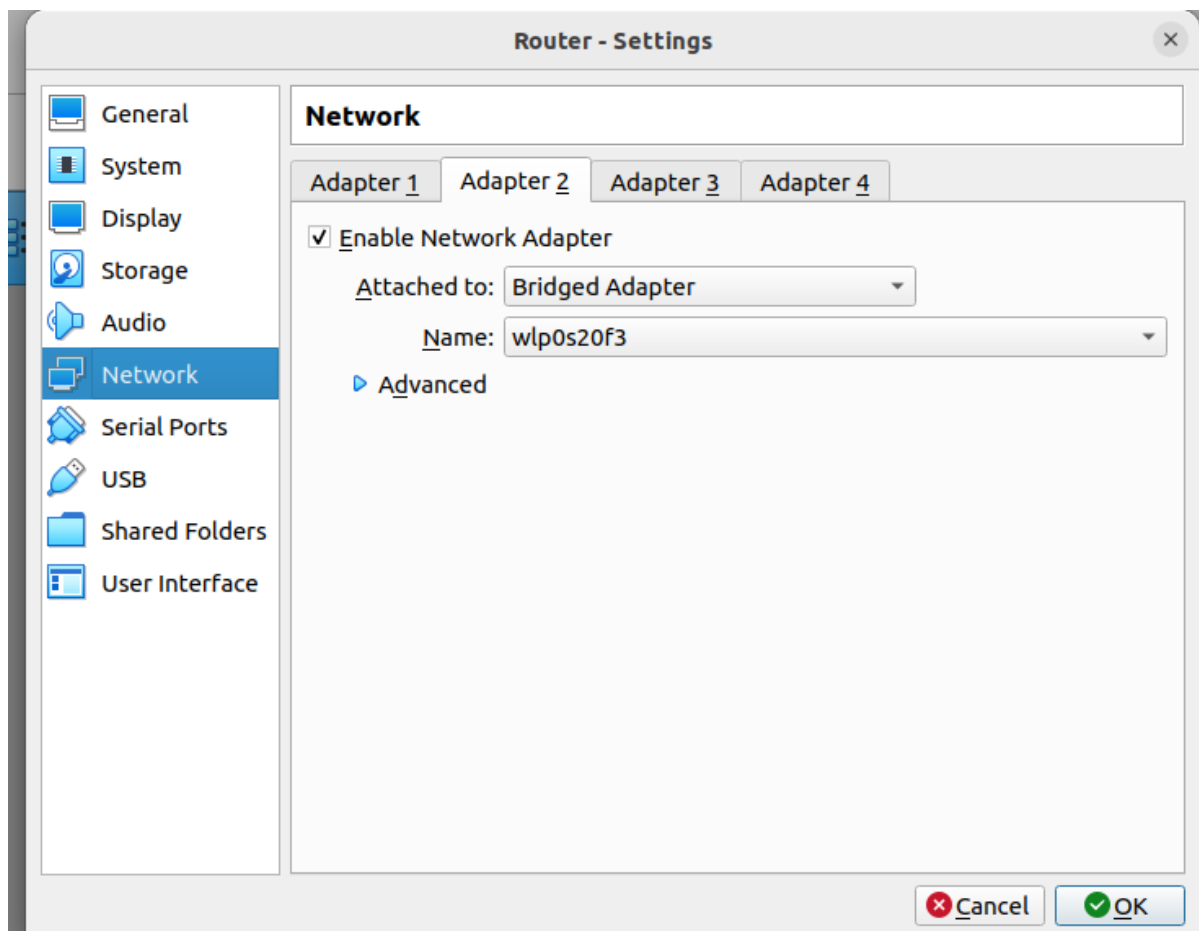
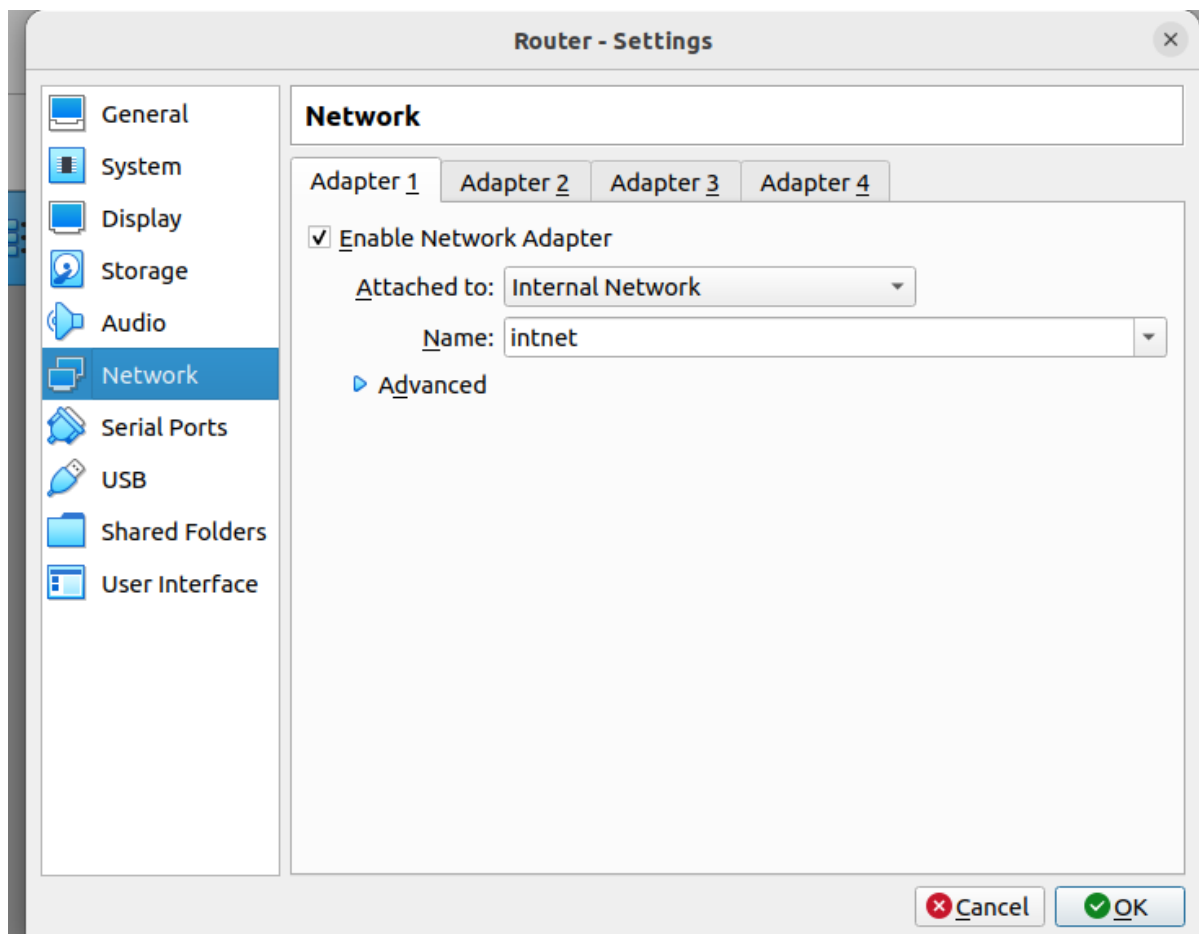
**author** Kacper Bohaczyk

**version** 26-05-2024

1. Erstellen von 2 Debian VM's (Server,Router)
2. **Beim Server:** unter Setttings --> Network --> Attached to Internal Network



1. **Beim Router:** unter Setttings --> Network --> Attached to Internal Network | |Adapter 2 --> Attached to Bridged Adapter



Herausfinden der Bridged Address von meinem PC mittels **ipconfig**

#### Wireless LAN adapter WLAN:

```
Connection-specific DNS Suffix . : home
Description . . . . . : Intel(R) Wi-Fi 6 AX201 160MHz
Physical Address. . . . . : BC-17-B8-C5-18-EA
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::d0df:2150:b278:489f%17(Preferred)
IPv4 Address. . . . . : 192.168.0.115(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : Samstag, 25. Mai 2024 16:38:37
Lease Expires . . . . . : Montag, 27. Mai 2024 20:37:38
Default Gateway . . . . . : 192.168.0.1
DHCP Server . . . . . : 192.168.0.1
DHCPv6 IAID . . . . . : 146544568
DHCPv6 Client DUID. . . . . : 00-01-00-01-27-32-0B-65-08-97-98-C7-D9-67
DNS Servers . . . . . : 212.186.211.21
                        195.58.161.123
NetBIOS over Tcpip. . . . . : Enabled
```

Herausgefundene Gateway ins Bridged einfügen

```
Command Prompt
Subnet Mask . . . . . : 255.255.240.0
Default Gateway . . . . . :

Ethernet adapter Ethernet 3:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::6361:b186:896a:fef3%3
IPv4 Address. . . . . : 192.168.56.1
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . :

Wireless LAN adapter LAN-Verbindung* 1:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :

Wireless LAN adapter LAN-Verbindung* 3:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :

Wireless LAN adapter WLAN:

Connection-specific DNS Suffix . : home
Link-local IPv6 Address . . . . . : fe80::d0df:2150:b278:489f%17
IPv4 Address. . . . . : 192.168.0.115
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.0.1
```

Navigiere in zur sysctl.conf file um ip-forwarding zu aktivieren PATH= "/etc/sysctl.conf" (Ändere auf 1)

```

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

#####
# Additional settings - these settings can improve the network
# security of the host and prevent against some network attacks
# including spoofing attacks and man in the middle attacks through
# redirection. Some network environments, however, require that these
# settings are disabled so review and enable them as needed.
#
# Do not accept ICMP redirects (prevent MITM attacks)
#net.ipv4.conf.all.accept_redirects = 0
#net.ipv4.conf.default.accept_redirects = 0
# _or_
# Accept ICMP redirects only for gateways listed in our default
# gateway list (enabled by default)
# net.ipv4.conf.all.secure_redirects = 1
#
# Do not send ICMP redirects (we are not a router)
#net.ipv4.conf.all.send_redirects = 0
#
# Log Martian Packets
#net.ipv4.conf.all.log_martians = 1
#

#####
# Magic system request Key
# 0=disable, 1=enable all, >1 bitmask of sysrq functions
# See https://www.kernel.org/doc/html/latest/admin-guide/sysrq.html
# for what other values do
#kernel.sysrq=438

```

Man kann die Funktionsweise mittels "sysctl net.ipv4.ip\_forward" überprüfen

Erstellen eines Skripts mit Hilfe von iptables zum Wechsel und Übergabe von Ports (Bridged zu internal und zurück)

```

# Lösche alle bestehenden Regeln
iptables -F
iptables -X
iptables -t nat -F
iptables -t nat -X
iptables -t mangle -F
iptables -t mangle -X
sudo iptables -t nat -A PREROUTING -i enp0s8 -p tcp --dport 80 -j DNAT --
todestination 192.168.16.10:80
iptables -A FORWARD -p tcp -d 192.168.16.10 --dport 80 -m state --state
NEW,ESTABLISHED,RELATED -j ACCEPT

```

```
sudo iptables -t nat -A PREROUTING -i enp0s8 -p tcp --dport 443 -j DNAT --to-destination 192.168.16.10:443
iptables -A FORWARD -p tcp -d 192.168.16.10 --dport 443 -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT
sudo iptables -t nat -A PREROUTING -i enp0s8 -p tcp --dport 22 -j DNAT --to-destination 192.168.16.10:22
iptables -A FORWARD -p tcp -d 192.168.16.10 --dport 22 -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -p icmp -j ACCEPT
```

Script ausführbar machen

```
sudo chmod +x /etc/port_forwarding.sh
```

Speicherung der neuen Elemente, sodass sie beim neustart gleichbleiben

```
sudo apt install iptables-persistent
sudo netfilter-persistent save
sudo netfilter-persistent reload
```

## Server

Aufsetzen einer statischen Adresse

**Subnet** 192.168.16.0/24

**Address** 192.168.16.10

**Gateway** 192.168.16.1

Installation der SSH-Servers mittels OpenSSH

```
sudo apt install openssh-server
```

Starten

```
sudo systemctl start ssh
```

SSH enablen

```
sudo systemctl enable ssh
```

```
Docs: man:sshd(8)
      man:sshd_config(5)
t@server:/home/sergej# systemctl start ssh
t@server:/home/sergej# systemctl status ssh
ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: enabled)
   Active: active (running) since Mon 2024-05-27 13:44:13 UTC; 2s ago
     TriggeredBy: • ssh.socket
```

Funktioniert :)

## Installation des Webserver mittels Nginx

```
sudo apt install nginx
```

### Nginx erlauben

```
sudo systemctl enable nginx
```

### öffnet der Firewall shell

```
nano /etc/firewall.sh
```

### Eine statefull Firewall aufsetzen

```
# Löschen aller bestehenden Regeln
sudo iptables -F
sudo iptables -X
sudo iptables -t nat -F
sudo iptables -t nat -X
sudo iptables -t mangle -F
sudo iptables -t mangle -X
sudo iptables -t raw -F
sudo iptables -t raw -X

# Standard Richtlinien setzen
sudo iptables -P INPUT DROP
sudo iptables -P FORWARD DROP
sudo iptables -P OUTPUT ACCEPT

# Erlauben von bestehenden Verbindungen
sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

# Erlauben von SSH, HTTP und HTTPS
sudo iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 80 -m conntrack --ctstate NEW -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 443 -m conntrack --ctstate NEW -j ACCEPT

# Erlauben des Loopback-Interfaces
sudo iptables -A INPUT -i lo -j ACCEPT

# Optional: Erlauben von ICMP (Ping)
sudo iptables -A INPUT -p icmp -m conntrack --ctstate NEW,ESTABLISHED,RELATED -j ACCEPT
```

### Ausführbar machen

```
chmod +x /etc/firewall.sh
```

### Ausführen

```
sudo /etc/firewall.sh
```

Speicherung der neuen Elemente, sodass sie beim neustart gleichbleiben

```
sudo apt install iptables-persistent  
sudo netfilter-persistent save  
sudo netfilter-persistent reload
```

## Testen

---

http://192.168.0.141

### Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*