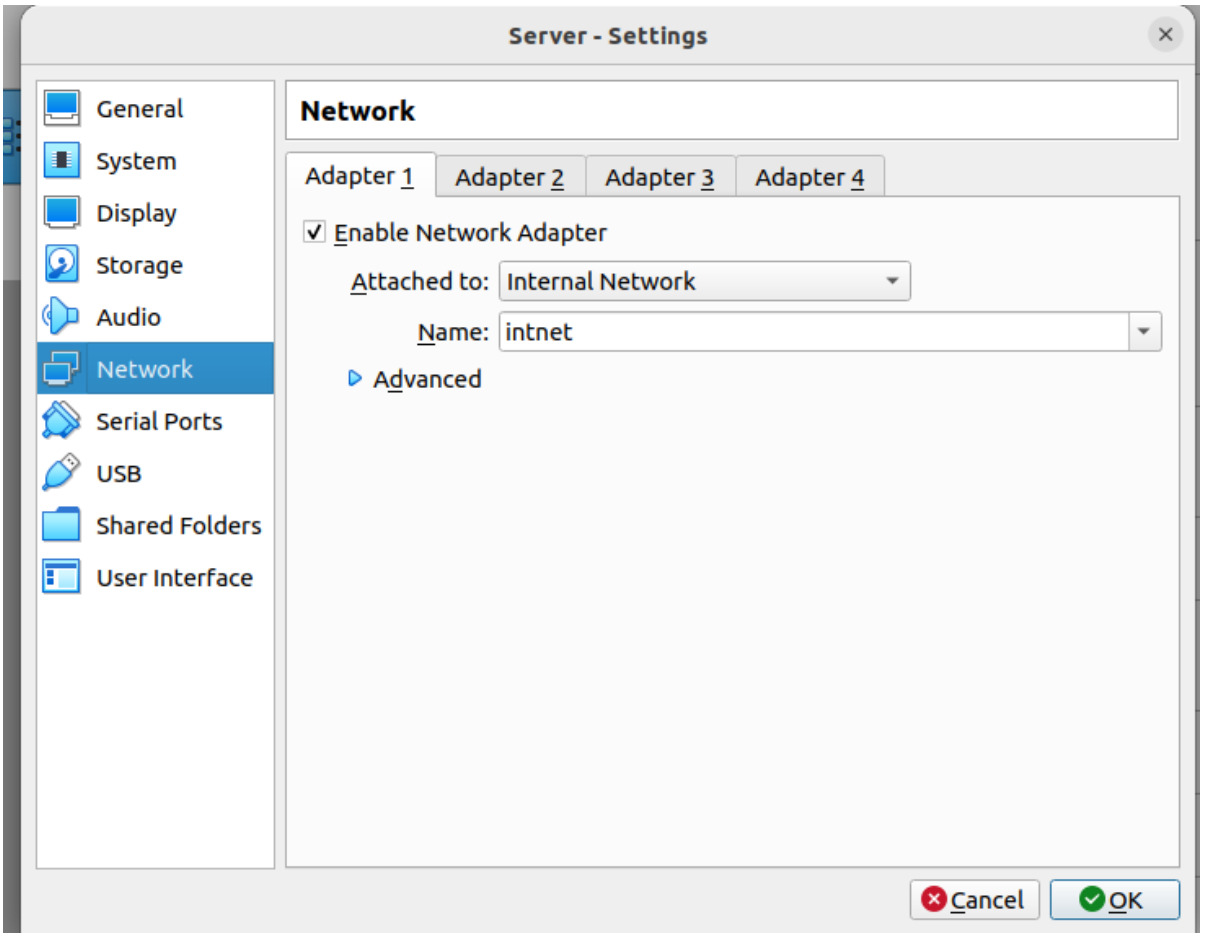
Firewall

author Kacper Bohaczyk

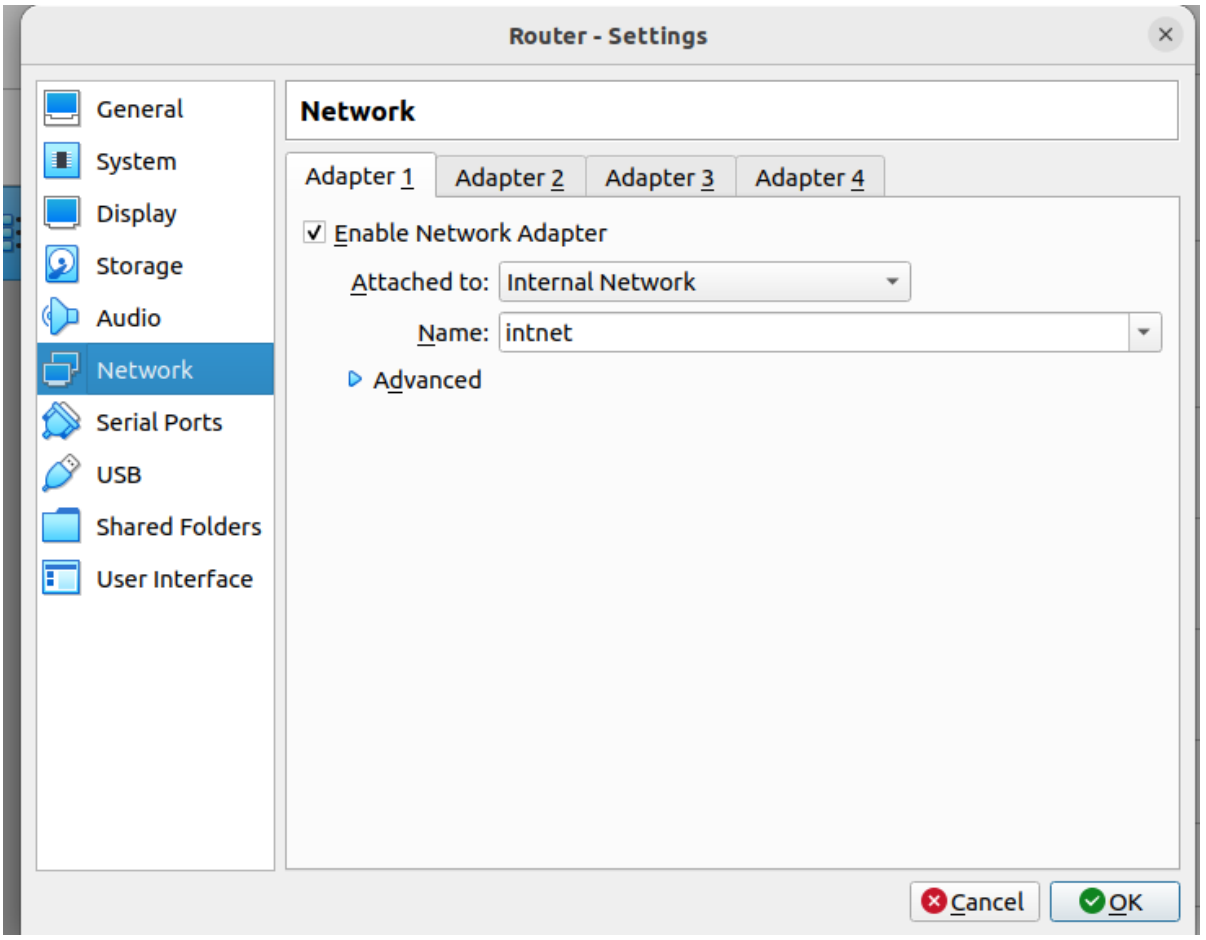
version 26-05-2024

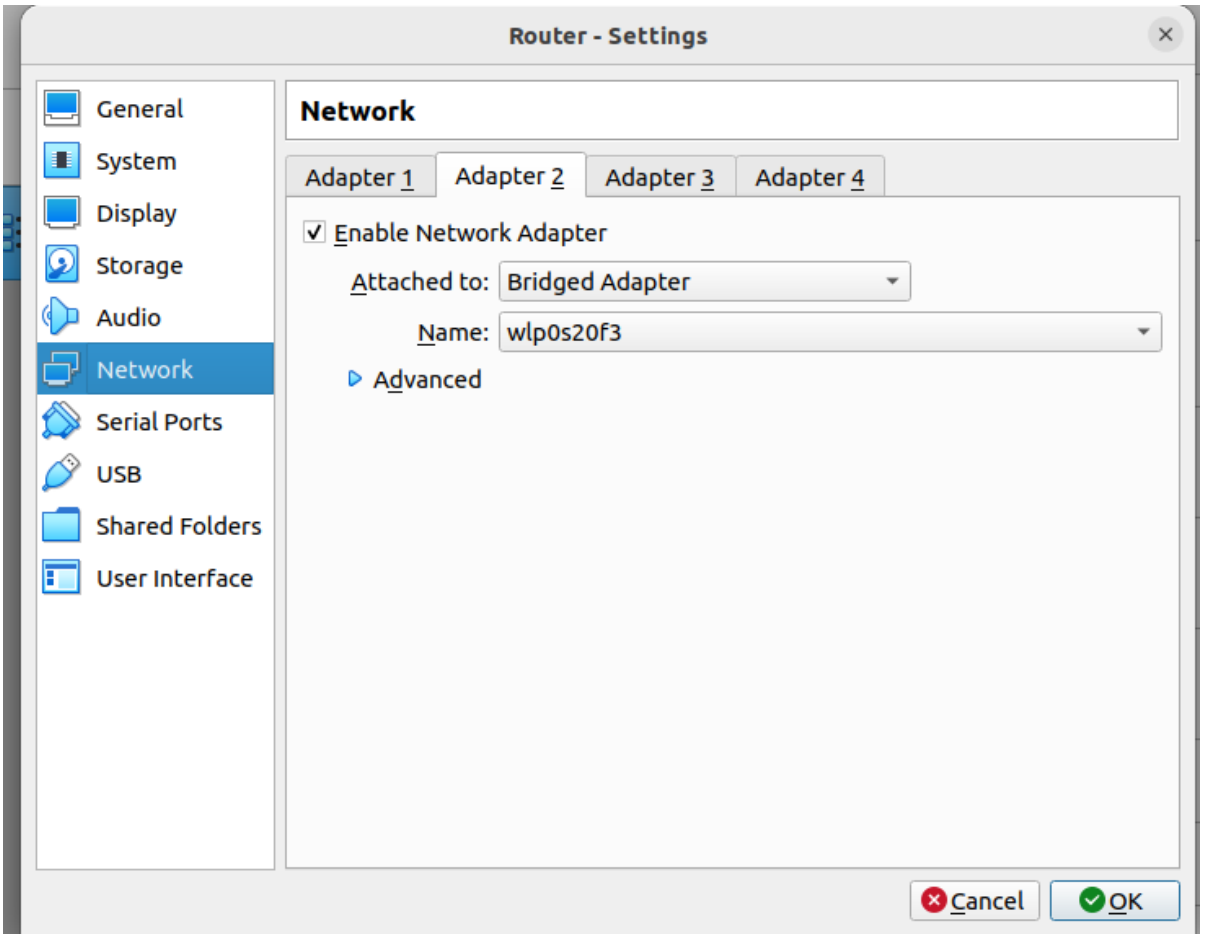
---

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



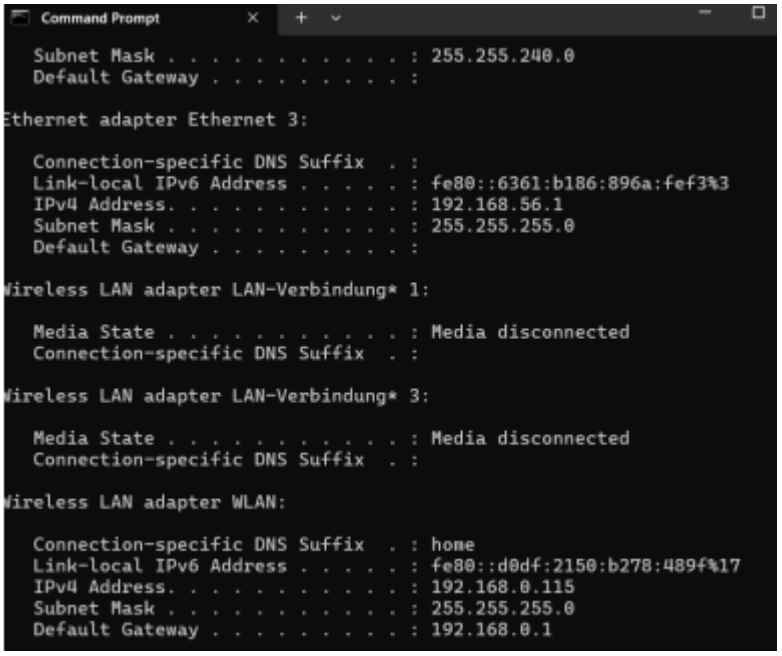


Herausfinden der Briged Address von meinem PC mittels **ipconfig**

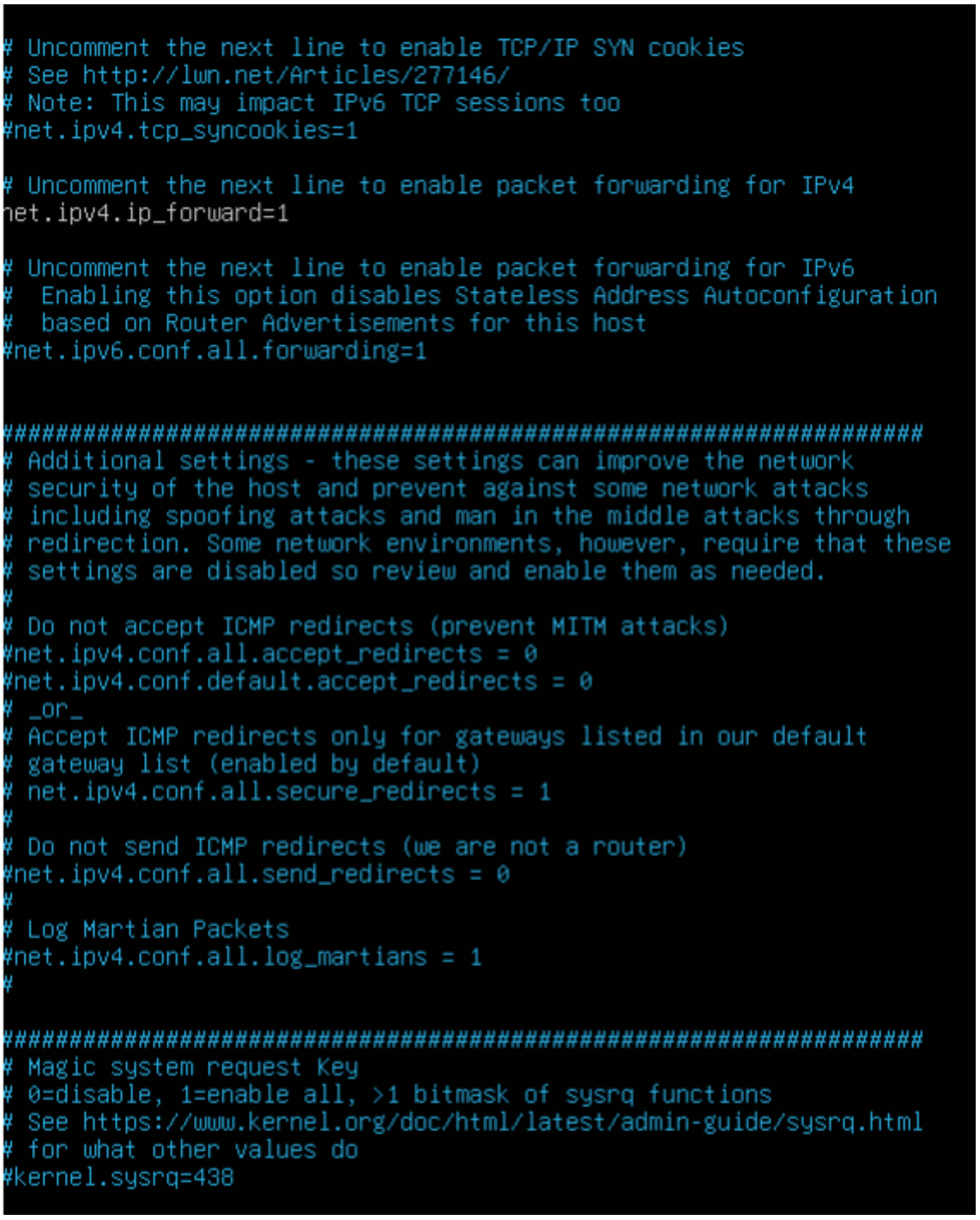
```
Wireless LAN adapter WLAN:

   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



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



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 tco -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 tco -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 --
todestination 192.168.16.10:22
iptables -A FORWARD -p tco -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 Addresse

Subnet 192.168.16.0/24

Address 192.168.16.10

Gateway 192.168.16.1

Installation der SSH-Servers mittels OpenSSH

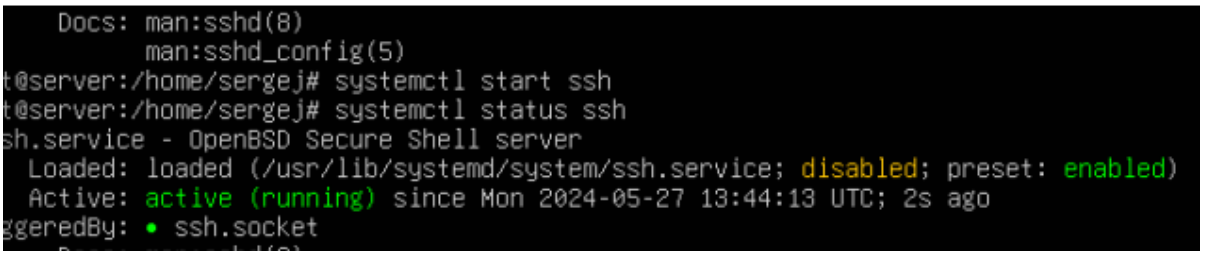
```
sudo apt install openssh-server
```

Starten

```
sudo systemctl start ssh
```

SSH enablen

```
sudo systemctl enable ssh
```



Funktioniert :)

Installation des Webservers 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