
Praktikum Rechnernetze

Protokoll zu Versuch 3 (Router-Betriebssystem Cisco IOS)
von Gruppe 1

Jakob Waibel, Daniel Hiller, Elia Wüstner, Felix Pojtinger

2021-10-19

Inhaltsverzeichnis

1	Einführung	3
1.1	Mitwirken	3
1.2	Lizenz	3
2	Konfiguration	4
2.1	Konfiguration des Routers, so dass er mittels ping oder telnet von ihrem Rechner erreichbar ist	4
3	Internet-Verbindung unter einsatz von NAT	4
3.1	Konfigurieren Sie ihren Router unter Einsatz von NAT so, dass von einem angeschlossenen PC aus eine Internet verbindung moeglich ist.	4
3.2	Erläutern Sie in der Ausarbeitung die Bedeutung der einzelnen Zeilen der Konfiguration	6
3.3	Dokumentieren Sie die Router-Konfiguration und die Routing-Tabelle des Routers und des PCs	8
3.4	Experimentieren Sie mit nachfolgenden Befehlen nach Aufruf einer beliebigen Website und dokumentieren Sie Ihre Ergebnisse	10
4	Internet-Verbindung ohne NAT	13
4.1	Konfigurieren Sie Ihren Router ohne NAT so, dass vom Subnetz ihrer Wahl eine Internet-Verbindung moeglich ist. Richten Sie dabei jeweils zwei Subnetze ein und stellen Sie zusaetzlich sicher, dass beide Subnetze sich gegenseitig erreichen koennen.	13
4.2	Dokumenteiren Sie die Konfiguration und auch die Routing-Tabelle des Routers und des PCs	13
5	Konfiguration Access-Liste (nur wenn die Zeit reicht)	14
5.1	Richten Sie eine Access-Liste ein, sodass TCP und UDP Verbinungen vom Router nur erlaubt werden, sofern Sie von ihrem PC kommen. Versuchen Sie mit einer anderen IP-Adresse ins Internet zu gelangen, so werden TCP/UDP-Verbindungen unterbunden.	14
5.2	Richten Sie eine Access-Liste ein, sodass ICMP Pakete (ping etc.) nur beantwortet werden, wenn sie von einem definierten Laborrechner kommen (141.62.66.x/24, suchen Sie sich einen aus).	14

1 Einführung

1.1 Mitwirken

Diese Materialien basieren auf [Professor Kiefers “Praktikum Rechnernetze”-Vorlesung der HdM Stuttgart](#).

Sie haben einen Fehler gefunden oder haben einen Verbesserungsvorschlag? Bitte eröffnen Sie ein Issue auf GitHub (github.com/pojntfx/uni-netpractice-notes):



Abbildung 1: QR-Code zum Quelltext auf GitHub

Wenn ihnen die Materialien gefallen, würden wir uns über einen GitHub-Stern sehr freuen.

1.2 Lizenz

Dieses Dokument und der enthaltene Quelltext ist freie Kultur bzw. freie Software.



Abbildung 2: Badge der AGPL-3.0-Lizenz

Uni Network Practice Notes (c) 2021 Jakob Waibel, Daniel Hiller, Elia Wüstner, Felix Pojtinger

SPDX-License-Identifier: AGPL-3.0

2 Konfiguration

Anders als in der Anleitung beschrieben, haben wir den Versuch mit Ubuntu durchgeführt. Daher im Folgenden eine kleine Anleitung, wie man sich unter Ubuntu mit dem Router verbinden kann.

Zuerst muss man die Anwendung `screen` installieren.

```
1 sudo apt install screen
```

Bevor man den Router nun einsteckt, kann man mit Hilfe von `dmesg` feststellen, welche Gerätebezeichnung der Router hat.

```
1 sudo dmesg | grep -i tty
```

Steckt man das Gerät nun ein, sollte man eine Meldung sehen, in welchem eine Device-Bezeichnung zu finden ist. In unserem Fall `tttyUSB0`.

Abschließend muss man sich nur noch mit der Cisco-Konsole verbinden. Dies lässt sich mit folgendem Kommando erreichen.

```
1 sudo screen /dev/ttyUSB0
```

Nun sollte eine Verbindung zur Cisco-Konsole bestehen.

2.1 Konfiguration des Routers, so dass er mittels ping oder telnet von ihrem Rechner erreichbar ist

TODO

3 Internet-Verbindung unter einsatz von NAT

3.1 Konfigurieren Sie ihren Router unter Einsatz von NAT so, dass von einem angeschlossenen PC aus eine Internet verbindung moeglich ist.

Konfiguration `interface GigabitEthernet 0/1`

Interface `GigabitEthernet 0/1` ist in unserer Konfiguration das LAN-Interface

```
cisco-gruppe1(config)#interface GigabitEthernet 0/1
cisco-gruppe1(config-if)#ip address 192.168.1.1 255.255.255.0
cisco-gruppe1(config-if)#ip nat inside
```

```
Nov  2 13:39:50.107: %LINEPROTO-5-UPDOWN: Line protocol on Interface NVI0, changed state to up
```

Abbildung 3: Konfiguration interface GigabitEthernet 0/1

Konfiguration **interface** GigabitEthernet 0/0

Interface GigabitEthernet 0/0 ist in unserer Konfiguration das WAN-Interface

Anfangs haben wir die falsche IP 141.62.67.2 gesetzt. Diese haben wir im Nachhinein korrigiert.

```
cisco-gruppel(config)#interface GigabitEthernet 0/0
cisco-gruppel(config-if)#ip address 141.62.67.2 255.255.255.248
cisco-gruppel(config-if)#ip nat outside
cisco-gruppel(config-if)#$ HDM 141.62.67.2 141.62.67.2 prefix-length 29
cisco-gruppel(config)#ip nat inside source list 8 pool HDM overload
cisco-gruppel(config)#access-list 8 permit 192.168.1.0 0.0.0.255
```

Abbildung 4: Erste, Fehlerbehaftete Konfiguration

Mit `clear ip nat translation *` können die falschen Konfigurationen rückgängig gemacht werden.

```
cisco-gruppel#clear ip nat translation *
cisco-gruppel#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
cisco-gruppel(config)#no ip nat pool HDM 141.62.66.161 141.62.66.161 prefix-le$
cisco-gruppel(config)#ip nat pool HDM 141.62.66.161 141.62.66.161 prefix-lengt$
```

Abbildung 5: Konfiguration interface GigabitEthernet 0/0

Nun muss noch sichergestellt werden, dass wirklich alle interfaces den Status **up** besitzen. Andernfalls können diese mit `no shutdown` in der jeweiligen Interface-Konfiguration aktiviert werden.

```
cisco-gruppel#show ip interface
Interface                               IP-Ad
Embedded-Service-Engine0/0            unass
GigabitEthernet0/0                     141.6
GigabitEthernet0/1                     192.1
NVI0                                    141.6
```

Interfaces mit `show ip interface brief` anzeigen und deren Status abfragen.

Danach kann am Router im `config` mode mit `ip route 0.0.0.0 0.0.0.0 141.62.66.250` die Route zum Router festgelegt werden und die Verbindung zum Internet sollte hergestellt sein.

Bevor der Lokale Computer über unseren Router eine Internetverbindung aufbauen kann, muss auch dieser konfiguriert werden.

Zuerst entfernen wir die alte IP von unserem Netzwerkinterface `enp0s31f6`.

```
praktikum@rn05:~$ sudo ip addr flush enp0s31f6
praktikum@rn05:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: enp0s31f6: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast state DOWN group default qlen 1000
    link/ether 4c:52:62:0e:54:8b brd ff:ff:ff:ff:ff:ff
```

Abbildung 6: IP entfernen

Danach fügen wir unsere neu bestimmte IP-Adresse zum Netzwerk-Interface hinzu.

```
praktikum@rn05:~$ sudo ip route add default via 192.168.1.1
praktikum@rn05:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: enp0s31f6: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast state DOWN group default qlen 1000
    link/ether 4c:52:62:0e:54:8b brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.5/24 scope global enp0s31f6
        valid_lft forever preferred_lft forever
praktikum@rn05:~$ ip route show
default via 192.168.1.1 dev enp0s31f6 linkdown
192.168.1.0/24 dev enp0s31f6 proto kernel scope link src 192.168.1.5 linkdown
```

Abbildung 7: Hinzufügen der neuen IP

Testen der Internetverbindung unseres Lokalen Computers mit einem ping zu 8.8.8.8 (Googles Public DNS-Server). Dafür kann der Command `ping 8.8.8.8` verwendet werden.

```
praktikum@rn05:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=113 time=46.1 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=113 time=26.9 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=113 time=14.3 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=113 time=4.75 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=113 time=4.86 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=113 time=4.68 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=113 time=4.90 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=113 time=4.75 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=113 time=4.82 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=113 time=4.87 ms
64 bytes from 8.8.8.8: icmp_seq=11 ttl=113 time=4.68 ms
64 bytes from 8.8.8.8: icmp_seq=12 ttl=113 time=4.82 ms
64 bytes from 8.8.8.8: icmp_seq=13 ttl=113 time=5.32 ms
^C
--- 8.8.8.8 ping statistics ---
13 packets transmitted, 13 received, 0% packet loss, time 12019ms
rtt min/avg/max/mdev = 4.675/10.436/46.105/11.995 ms
praktikum@rn05:~$ █
```

Abbildung 8: Ping an den Google-DNS-Server

3.2 Erläutern Sie in der Ausarbeitung die Bedeutung der einzelnen Zeilen der Konfiguration

```
1 interface GigabitEthernet 0/1
2 ip address 192.168.1.1 255.255.255.0
3 ip nat inside
4
5 interface GigabitEthernet 0/0
6 ip address 141.62.66.161 255.255.255.0
7 ip nat outside
8 ip nat pool HDM 141.62.66.161 141.62.66.161 prefix-length 24
9 ip nat inside source list 8 HDM overload
10 access-list 8 permit 192.168.1.0 0.0.0.255
```

interface GigabitEthernet 0/1

In den Interface-Konfigurations-Modus des Interfaces `GigabitEthernet 0/1` wechseln, um dieses zu konfigurieren. Dieses Interface ist in unserem Versuch das LAN-Interface.

ip address 192.168.1.1 255.255.255.0

Dem Router, in dem momentan konfigurierbaren Interface `GigabitEthernet 0/1` die IP `192.168.1.1` mit der Subnetzmaske `255.255.255.0` zuweisen.

ip nat inside

Verbindet das interface `GigabitEthernet 0/1` mit dem inneren Netzwerk, welches von NAT betroffen ist.

interface GigabitEthernet 0/0

Wechselt in den Interface-Konfigurations-Modus des Interfaces `GigabitEthernet 0/0`, um dieses zu konfigurieren. Dieses Interface ist in unserem Versuch das WAN-Interface.

ip address 141.62.66.161 255.255.255.0

Mit diesem Command wird dem Router, in dem momentan konfigurierbaren Interface `GigabitEthernet 0/0`, die IP-Adresse `141.62.66.161` mit der Subnetzmaske `255.255.255.0` zugewiesen.

ip nat outside

Verbindet das Interface `GigabitEthernet 0/0` mit dem außenstehenden Netzwerk.

ip nat pool HDM 141.62.66.161 141.62.66.161 prefix-length 24

Definiert einen NAT-Pool mit der Adress-Range von `141.62.66.161` bis `141.62.66.161`, also genau diese Adresse. Zusätzlich ist noch der Netzwerk-Präfix angegeben, der in unserem Beispiel 24 Bit lang ist.

ip nat inside source list 8 HDM overload

Verändert die Source-IP der Pakete, die von innen nach aussen geleitet werden. Übersetzt die Destination-IP der Pakete, die von außen nach innen geleitet werden.

access-list 8 permit 192.168.1.0 0.0.0.255

Konfiguriert die Access-Control-List, insoweit, dass Pakete der IP 192.168.1.0 weitergeleitet werden dürfen.

3.3 Dokumentieren Sie die Router-Konfiguration und die Routing-Tabelle des Routers und des PCs

Die Konfiguration lässt sich mit `show running-config` anzeigen.

```
1  cisco-gruppe1#show running-config
2  Building configuration...
3
4  Current configuration : 1483 bytes
5  !
6  ! Last configuration change atstname cisco-gruppe1
7  !
8  boot-start-marker
9  boot-end-marker
10 !
11 !
12 !
13 no aaa new-model
14 !
15 no ipv6 cef
16 ip source-route
17 ip cef
18 !
19 !
20 !
    !   !   r
21 multilink bundle-name authenticated
22 !
23 --More--  default removal timeout 0
24 !
25 !
26 license udi pid CISC01941/K9 sn FTX1636824P
27 --More--
28      !
29 !
30 !
31 !
32 !
    shutdown
33 !
34 interface GigabitEthernet61 255.255.255.0
35 ip nat outside
```



```
36 ip virtual-reassembly in
37 duplex auto
38 speed auto
39 !
    speed auto
40 !
41 ip forward-protocol nd
42 !
43 no ip http server
44 no ip http secure-server
45 !
46 ip nat pool HDM 141.62.66.161 141.62.66.161 prefix-length 24
47 --Moreermit 192.168.1.0 0.0.0.255
48 !
49 !
50 !
51 control-plane
    !
52 line con 0
    line 2 --
53 no activation-character
54 no exec
55 transport preferred none
56 transport input all
57 transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
58 stopbits 1
59 line vty 0 4
60 password hdm
61 login
62 transport input all
63 !
64 scheduler allocate 20000 1000
65 end
```

Die Routing-Tabelle des Routers kann mit `show ip route` angezeigt werden.

```
cisco-gruppel#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is 141.62.66.250 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 141.62.66.250
      141.62.0.0/16 is variably subnetted, 2 subnets, 2 masks
C      141.62.66.0/24 is directly connected, GigabitEthernet0/0
L      141.62.66.161/32 is directly connected, GigabitEthernet0/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C      192.168.1.0/24 is directly connected, GigabitEthernet0/1
L      192.168.1.1/32 is directly connected, GigabitEthernet0/1
cisco-gruppel#
```

Abbildung 9: Routing-Tabelle des Routers

Die Routing Tabelle des Lokalen Computers kann mit `ip route show` angezeigt werden. Zusätzlich nutzen wir `ip a`, um die Netzwerk-Interfaces und deren jeweilige IP-Adressen zu betrachten.

```
praktikum@rn05:~$ ip route show
default via 192.168.1.1 dev enp0s31f6
192.168.1.0/24 dev enp0s31f6 proto kernel scope link src 192.168.1.5
praktikum@rn05:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
2: enp0s31f6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
   link/ether 4c:52:62:0e:54:8b brd ff:ff:ff:ff:ff:ff
   inet 192.168.1.5/24 scope global enp0s31f6
       valid_lft forever preferred_lft forever
praktikum@rn05:~$
```

Abbildung 10: Routing-Tabelle des Lokalen Computers

3.4 Experimentieren Sie mit nachfolgenden Befehlen nach Aufruf einer beliebigen Website und dokumentieren Sie Ihre Ergebnisse

Als Erstes wurde unser Router von unserem Lokalen Computer angepingt.

```
praktikum@rn05:~$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=255 time=0.412 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=255 time=0.579 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=255 time=0.509 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=255 time=0.365 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=255 time=0.436 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=255 time=0.415 ms
■
```

Abbildung 11: Ping an unseren Router

Danach wurde der Router im Rechnernetze-Labor von unserem Router angepingt.

```
cisco-gruppel#ping 141.62.66.250
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 141.62.66.250, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

Abbildung 12: Ping an Router im Rechnernetze-Labor

Danach haben wir den Google DNS-Server angepingt.

```
praktikum@rn05:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=113 time=46.1 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=113 time=26.9 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=113 time=14.3 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=113 time=4.75 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=113 time=4.86 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=113 time=4.68 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=113 time=4.90 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=113 time=4.75 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=113 time=4.82 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=113 time=4.87 ms
64 bytes from 8.8.8.8: icmp_seq=11 ttl=113 time=4.68 ms
64 bytes from 8.8.8.8: icmp_seq=12 ttl=113 time=4.82 ms
64 bytes from 8.8.8.8: icmp_seq=13 ttl=113 time=5.32 ms
^C
--- 8.8.8.8 ping statistics ---
13 packets transmitted, 13 received, 0% packet loss, time 12019ms
rtt min/avg/max/mdev = 4.675/10.436/46.105/11.995 ms
praktikum@rn05:~$ ■
```

Abbildung 13: Ping an den Google DNS-Server

Von den folgenden Kommandos haben wir vergessen Screenshots zu machen, daher finden sich im Folgenden Bilder aus dem Internet, die die Funktionalität der Kommandos illustrieren sollen. Die Konfiguration unterscheidet sich offensichtlich.

```
show ip nat statistics
```

```
testrouter#show ip nat statistics
Total active translations: 1 (0 static, 1 dynamic; 1 extended)
Peak translations: 298, occurred 00:15:07 ago
Outside interfaces:
  GigabitEthernet0/1
Inside interfaces:
  GigabitEthernet0/0
Hits: 20807 Misses: 0
CEF Translated packets: 19285, CEF Punted packets: 1380
Expired translations: 1096
Dynamic mappings:
-- Inside Source
[Id: 1] access-list 100 interface GigabitEthernet0/1 refcount 1

Total doors: 0
Appl doors: 0
Normal doors: 0
Queued Packets: 0
testrouter#
```

Abbildung 14: <http://blog.soundtraining.net/2013/02/nat-configuration-on-cisco-router-port.html>

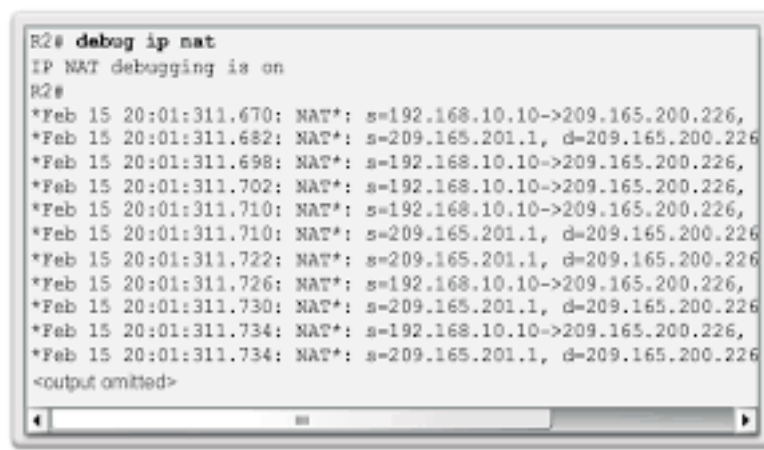
show ip nat translation

```
R1#show ip nat translation
Pro  Inside global      Inside local      Outside local      Outside global
icmp 50.0.0.1:1        10.0.0.20:1       200.0.0.10:1       200.0.0.10:1
icmp 50.0.0.1:2        10.0.0.20:2       200.0.0.10:2       200.0.0.10:2
icmp 50.0.0.1:3        10.0.0.20:3       200.0.0.10:3       200.0.0.10:3
icmp 50.0.0.1:4        10.0.0.20:4       200.0.0.10:4       200.0.0.10:4
tcp  50.0.0.1:1024     10.0.0.10:1025    200.0.0.10:80      200.0.0.10:80
tcp  50.0.0.1:1025     10.0.0.20:1025    200.0.0.10:80      200.0.0.10:80

R1#
```

Abbildung 15: <https://www.computernetworkingnotes.com/ccna-study-guide/configure-pat-in-cisco-router-with-examples.html>

debug ip nat



```
R2# debug ip nat
IP NAT debugging is on
R2#
*Feb 15 20:01:31.670: NAT*: s=192.168.10.10->209.165.200.226,
*Feb 15 20:01:31.682: NAT*: s=209.165.201.1, d=209.165.200.226
*Feb 15 20:01:31.698: NAT*: s=192.168.10.10->209.165.200.226,
*Feb 15 20:01:31.702: NAT*: s=192.168.10.10->209.165.200.226,
*Feb 15 20:01:31.710: NAT*: s=192.168.10.10->209.165.200.226,
*Feb 15 20:01:31.710: NAT*: s=209.165.201.1, d=209.165.200.226
*Feb 15 20:01:31.722: NAT*: s=209.165.201.1, d=209.165.200.226
*Feb 15 20:01:31.726: NAT*: s=192.168.10.10->209.165.200.226,
*Feb 15 20:01:31.730: NAT*: s=209.165.201.1, d=209.165.200.226
*Feb 15 20:01:31.734: NAT*: s=192.168.10.10->209.165.200.226,
*Feb 15 20:01:31.734: NAT*: s=209.165.201.1, d=209.165.200.226
<output omitted>
```

Abbildung 16:

https://www.google.com/search?q=debug+ip+nat&rlz=1C5CHFA_enDE964DE964&source=lnms&tbm=isch&sa=X&ved=84D0AhWE2KQKHWMIbfIQ_AUoAXoECAEQAw&biw=1622&bih=857&dpr=2#imgsrc=40zndQOWKiYsiM&imgdii=t1e4jXoag

4 Internet-Verbindung ohne NAT

4.1 Konfigurieren Sie Ihren Router ohne NAT so, dass vom Subnetz ihrer Wahl eine Internet-Verbindung moeglich ist. Richten Sie dabei jeweils zwei Subnetze ein und stellen Sie zusaetzlich sicher, dass beide Subnetze sich gegenseitig erreichen koennen.

TODO

4.2 Dokumentieren Sie die Konfiguration und auch die Routing-Tabelle des Routers und des PCs

TODO

5 Konfiguration Access-Liste (nur wenn die Zeit reicht)

5.1 Richten Sie eine Access-Liste ein, sodass TCP und UDP Verbindungen vom Router nur erlaubt werden, sofern Sie von ihrem PC kommen. Versuchen Sie mit einer anderen IP-Adresse ins Internet zu gelangen, so werden TCP/UDP-Verbindungen unterbunden.

TODO

5.2 Richten Sie eine Access-Liste ein, sodass ICMP Pakete (ping etc.) nur beantwortet werden, wenn sie von einem definierten Laborrechner kommen (141.62.66.x/24, suchen Sie sich einen aus).

TODO