

Rapport

**Introduktion till Linux och små nätverk, 7,5 hp
VT-2025**

Projektuppgift

Laboration 4: Nätverk, IPv4

av

Sixten Peterson (050402-XXXX)

**Akademin för teknik och miljö
Avdelningen för industriell utveckling, IT och samhällsbyggnad**

Högskolan i Gävle
S-801 76 Gävle, Sweden

Datorpost:

education@snicon.rip

Innehållsförteckning

| | |
|--|---|
| 1. Inledning..... | 1 |
| 1.1. Bakgrund..... | 1 |
| 1.2. Syfte..... | 1 |
| 2. Planering och genomförande..... | 1 |
| 2.1. Planering..... | 1 |
| 2.2. Genomförande..... | 2 |
| 2.2.1. Del ett – Det lokala nätverket..... | 2 |
| 2.2.1.1. IP-nummer..... | 2 |
| 2.2.1.2. Nät- och Nodnummer..... | 4 |
| 2.2.1.3. Routeradresser..... | 4 |
| 2.2.1.4. MAC-adresser..... | 5 |
| 2.2.2. Del två – Fiktiv nätverkskonfiguration..... | 5 |
| 2.2.2.1. Maskin 1 (m1)..... | 6 |
| 2.2.2.2. Maskin 2 (m2)..... | 7 |
| 3. Beskrivning av slutresultat..... | 7 |
| 4. Diskussion..... | 7 |
| 5. Slutsatser..... | 7 |
| 6. Referenser..... | 8 |
| I. Bilaga 1: MAC-adress Macbook Pro..... | 9 |

1. Inledning

Laborationen är det fjärde momentet i kursen med två delar som omfattar en undersökning av det lokala nätverket och framställandet av fiktiva nätverkskonfigurationer. Syftet är att bilda förståelse för hur ett lokalt hemmanätverk är uppbyggt samt hur nätverkskonfigurationsfiler kan framställas.

1.1. Bakgrund

Denna laboration utgör det fjärde momentet i kursen ”Introduktion till Linux och små nätverk” (DVG001) och består av två delar. Samtliga delar utförs på labbmiljön som tagits fram under den första laborationen.

I del ett undersöks det lokala nätverket i syfte att ta reda på följande:

1. Vilken ip-adress laborationsmiljön har, samt hur den erhållits.
2. Vilken nätadress maskinen har med såväl CIDR som nätmask samt var informationen är hämtad.
3. Vilka routrar maskinen känner till, samt vilken som är standardroutern.
4. Vilka MAC- och IP-adresser som de andra maskinerna som finns på nätverket har.

Under del två utformas två skilda nätverkskonfigurationer för två olika fiktiva maskiner i ett fiktivt nätverk utifrån följande premisser:

- Två skilda maskiner (m1 och m2) är anslutna till samma nätverk.
- Nätverket har en router (router) med adressen 192.168.133.162/25.
- Maskin m1 har nodadressen 15, medan m2 har nodadressen 16 i samma nät som routern.
- För maskin m1 gäller statisk inställning av nätverket eth0.
- För maskin m2 gäller dynamisk (dhcp) inställning av nätverket på eth0.

1.2. Syfte

Laborationen syftar till att bilda en förståelse för hur ett lokalt hemmanätverk är uppbyggt. Ytterligare syftar laborationen till att undersöka hur konfigurationsfiler för nätverket kan framställas i Linux (Debian).

2. Planering och genomförande

Genom hela laborationen sker kontinuerlig dokumentation samt en SSH-anslutning till Linuxmiljön för utförandet av praktiska moment. Vid eventuella problem nyttjas föreläsningsmaterial och internet.

2.1. Planering

Arbetet dokumenteras kontinuerligt i realtid i samband med laborationens utförande för att säkerställa så akkurat information i rapporten som möjligt. Vid eventuella problem eller funderingar nyttjas i första hand föreläsningsmaterialet och i andra hand en sökmotor såsom DuckDuckGo eller Google. Där officiell dokumentation för distributionen finns tillgänglig på internet prioriteras denna högst. Kommunikation upprättas mot labbmiljön som kör Debian 12 Bookworm på en gammal Dell Inspiron 570 genom en SSH-anslutning från min Macbook Pro med macOS Sequoia 15.3.2.

2.2. Genomförande

Genomförandet är uppdelat i två olika delar som speglar de olika delarna av uppgiftsbeskrivningen. Nedan följer även en redogörelse av det som utförts innan samtliga delar.

Först upprättades en anslutning till labbmiljön genom `ssh hig-25sipe01@192.168.1.250` i terminalen på min Macbook. Därefter skrevs lösenordet för kontot in, väl ansluten användes `sudo apt update` och `sudo apt upgrade` för att installera de senaste säkerhetsuppdateringarna i enlighet med kursboken [1, pp 116, 120].

2.2.1. Del ett – Det lokala nätverket

Nedan följer såväl uppgiftsbeskrivningar som lösningar för vardera deluppgift som utförs under den första delen av laborationen.

2.2.1.1. IP-nummer

Uppgiften är att ta reda på vilken IP-adress som maskinen har samt hur den får sin adress utifrån att använda kommandot `ip` och titta i loggar, exempelvis `/var/log/messages`.

Jag vet sedan innan att maskinen får sin IP-adress genom *DHCP-servern* på routern, men för att bekräfta detta samt stilla nyfikenhet av hur interfaces-filen ser ut så börjar jag först med att skriva ut innehållet i `/etc/network/interfaces`. Detta utfördes genom att skriva `cat /etc/network/interfaces`, se resultat i Figur 1. Mycket riktigt används *DHCP*. Eftersom jag anslutit till servern över SSH så har jag såklart även koll på IP-adressen som jag hämtat från klientlistan på routern som snabbt gick att hitta tack vare värdnamnet som är baserat på kurskoden. Men för att testa hur `ip`-kommandot fungerar – jag har tidigare bara använt `ifconfig` i äldre versioner av Ubuntu Server LTS – så kör jag kommandot `ip address show`. Resultatet av kommandot återfinns i Figur 2, och redovisar att den lokala IPv4-adressen är 192.168.1.250.

```
hig-25sipe01@dvg001:~$ cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug enp2s0
iface enp2s0 inet dhcp
```

Figur 1: Utdata från `cat`-kommandot.

```
hig-25sipe01@dvg001:~$ ip address show
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
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 84:2b:2b:93:a3:a9 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.250/24 brd 192.168.1.255 scope global dynamic enp2s0
        valid_lft 67174sec preferred_lft 67174sec
    inet6 fe80::862b:2bff:fe93:a3a9/64 scope link
        valid_lft forever preferred_lft forever
hig-25sipe01@dvg001:~$
```

Figur 2: Utdata från `ip`-kommandot, här visas bland annat lokal IP-adress.

Som Figur 3 visar så gick det inte att hitta katalogen `/var/log/messages` i min labbmiljö. Således provade jag att istället besöka `/var/log` för att undersöka vilka andra kataloger som kan finnas där med hjälp av `cd /var/log`. Väl där användes `ls` för att få en bild av vad som fanns där, bland annat kunde jag skåda filen `README` som såklart väckte mitt intresse. Genom att använda `cat README` kunde jag läsa om att `syslog` blivit utbytt mot `Journal` och att `journalctl(1)` kunde användas för att söka bland loggar.

```
hig-25sipe01@dvg001:~$ cd /var/log/messages
-bash: cd: /var/log/messages: No such file or directory
hig-25sipe01@dvg001:~$
```

Figur 3: Katalogen `/var/log/messages` kunde inte hittas.

Därpå användes man `journalctl` för att få en uppfattning av hur kommandot fungerar. Genom manualen hittades flaggan `-g` för `grep`, således användes kommandot `sudo journalctl -g "192.168.1.250"` för att filtrera fram relevanta rader bland loggarna – se Figur 4. Utan att vara expert på `DHCP` så ser det ut som att `dhclient` på maskinen för kommunikation med routern (192.168.1.1) och tar emot ett erbjudande om att använda IP-adressen 192.168.1.250, och att det sker en kommunikation mellan klienten och routern upprepade gånger. Fenomenet uppstår rimligtvis eftersom `DHCP` bygger på att en ip-adress blir hyrd/leasad i en viss tid och efter en viss tid behöver hyran av adressen förnyas. Således är det härifrån labbmiljön får sin IP-adress (192.168.1.250).

```
-- Boot bbda7983c422412b89531dd8ed3d48fa --
Mar 08 22:11:17 dvg001 dhclient[342]: DHCPPOFFER of 192.168.1.250 from 192.168.1.1
Mar 08 22:11:17 dvg001 sh[342]: DHCPPOFFER of 192.168.1.250 from 192.168.1.1
Mar 08 22:11:17 dvg001 sh[342]: DHCPREQUEST for 192.168.1.250 on enp2s0 to 255.255.255.255 port 67
Mar 08 22:11:17 dvg001 dhclient[342]: DHCPREQUEST for 192.168.1.250 on enp2s0 to 255.255.255.255 port 67
Mar 08 22:11:17 dvg001 dhclient[342]: DHCPACK of 192.168.1.250 from 192.168.1.1
Mar 08 22:11:17 dvg001 sh[342]: DHCPACK of 192.168.1.250 from 192.168.1.1
Mar 08 22:11:17 dvg001 dhclient[342]: bound to 192.168.1.250 -- renewal in 33934 seconds.
Mar 08 22:11:17 dvg001 sh[342]: bound to 192.168.1.250 -- renewal in 33934 seconds.
Mar 08 22:58:01 dvg001 ntpd[859]: IO: Listen normally on 3 enp2s0 192.168.1.250:123
Mar 08 23:09:19 dvg001 ntpd[859]: PROTO: 98.128.175.45 unlink local addr 192.168.1.250 -> <null>
Mar 08 23:17:11 dvg001 ntpd[859]: PROTO: 98.128.230.186 unlink local addr 192.168.1.250 -> <null>
Mar 08 23:18:13 dvg001 ntpd[859]: PROTO: 194.58.204.20 unlink local addr 192.168.1.250 -> <null>
Mar 08 23:23:53 dvg001 ntpd[859]: PROTO: 77.239.113.15 unlink local addr 192.168.1.250 -> <null>
Mar 08 23:45:55 dvg001 ntpd[859]: PROTO: 150.241.82.187 unlink local addr 192.168.1.250 -> <null>
Mar 08 23:50:41 dvg001 ntpd[859]: PROTO: 194.58.203.148 unlink local addr 192.168.1.250 -> <null>
Mar 08 23:54:29 dvg001 ntpd[859]: PROTO: 194.58.204.148 unlink local addr 192.168.1.250 -> <null>
Mar 09 07:36:51 dvg001 dhclient[342]: DHCPREQUEST for 192.168.1.250 on enp2s0 to 192.168.1.1 port 67
Mar 09 07:36:51 dvg001 dhclient[342]: DHCPACK of 192.168.1.250 from 192.168.1.1
Mar 09 07:36:51 dvg001 dhclient[342]: bound to 192.168.1.250 -- renewal in 32198 seconds.
Mar 09 16:33:29 dvg001 dhclient[342]: DHCPREQUEST for 192.168.1.250 on enp2s0 to 192.168.1.1 port 67
Mar 09 16:33:29 dvg001 dhclient[342]: DHCPACK of 192.168.1.250 from 192.168.1.1
Mar 09 16:33:29 dvg001 dhclient[342]: bound to 192.168.1.250 -- renewal in 38209 seconds.
Mar 10 03:10:18 dvg001 dhclient[342]: DHCPREQUEST for 192.168.1.250 on enp2s0 to 192.168.1.1 port 67
Mar 10 03:10:18 dvg001 dhclient[342]: DHCPACK of 192.168.1.250 from 192.168.1.1
Mar 10 03:10:18 dvg001 dhclient[342]: bound to 192.168.1.250 -- renewal in 32186 seconds.
Mar 10 12:06:44 dvg001 dhclient[342]: DHCPREQUEST for 192.168.1.250 on enp2s0 to 192.168.1.1 port 67
Mar 10 12:06:44 dvg001 dhclient[342]: DHCPACK of 192.168.1.250 from 192.168.1.1
Mar 10 12:06:44 dvg001 dhclient[342]: bound to 192.168.1.250 -- renewal in 37621 seconds.
Mar 10 14:33:49 dvg001 ntpd[859]: PROTO: 192.36.143.130 unlink local addr 192.168.1.250 -> <null>
Mar 10 14:33:49 dvg001 ntpd[859]: PROTO: 162.159.200.1 unlink local addr 192.168.1.250 -> <null>
lines 1-76
```

Figur 4: Utdata från `journalctl` filtrerat på den lokala ip-adressen.

2.2.1.2. Nät- och Nodnummer

Uppgiften är att ange vilken nätadress jag har med hjälp av CIDR, ytterligare ska nätmasken anges separat.

Först användes `ip route list` i enlighet med informationen som finns med i uppgiftsbeskrivningen, se Figur 5. I figuren står det klart och tydligt 192.168.1.0/24, vi kan även konstatera att maskinens nodnummer är 250 utifrån informationen. Utöver detta ville jag även se vad programmet `ipcalc` skriver ut för IP-adressen, varför jag körde `ipcalc 192.168.1.250`. Som Figur 6 så smidigt visar så är nätmasken 255.255.255.0, ytterligare har vi bekräftat det vi redan vet – att nätadressen är 192.168.1.0 och att de första 24 bitarna av IP-adressen används för ”nätverksdelen” medan de sista åtta bitarna används som nodnummer för maskiner.

```
hig-25sipe01@dvg001:~$ ip route list
default via 192.168.1.1 dev enp2s0
192.168.1.0/24 dev enp2s0 proto kernel scope link src 192.168.1.250
```

Figur 5: Utdata från kommandot `ip route list` som bland annat visar nätnummer med CIDR.

```
hig-25sipe01@dvg001:/var/log$ man ipcalc
hig-25sipe01@dvg001:/var/log$ ipcalc 192.168.1.250
Address:      192.168.1.250      11000000.10101000.00000001. 11111010
Netmask:      255.255.255.0 = 24  11111111.11111111.11111111. 00000000
Wildcard:     0.0.0.255         00000000.00000000.00000000. 11111111
=>
Network:      192.168.1.0/24     11000000.10101000.00000001. 00000000
HostMin:      192.168.1.1       11000000.10101000.00000001. 00000001
HostMax:      192.168.1.254     11000000.10101000.00000001. 11111110
Broadcast:    192.168.1.255     11000000.10101000.00000001. 11111111
Hosts/Net:    254               Class C, Private Internet
```

Figur 6: Resultat av körning av `ipcalc` med den lokala IP-adressen.

2.2.1.3. Routeradresser

Uppgiften är att ange vilka routrar som labbmaskinen känner till och vilken router som är standardroutern.

Kort och gott används utdatan från `ip route list` i förra steget – se Figur 5 - och slutsatsen är att standardroutern är 192.168.1.1 samt att maskinen inte känner till någon annan router.

2.2.1.4. MAC-adresser

Uppgiften är att ange vilka MAC- och IP-adresser de andra maskinerna som finns på datorns nätverk har.

Förest kördes kommandot `ip neighbour show`, resultatet av detta blev att jag kunde se min Macbook Pro (192.168.1.183) i tabellen samt routern (192.168.1.1). För att testa och se hur tabellen uppdateras så bad jag min sambo öppna terminalen på sin Chromebook och pinga labbmiljön genom `ping 192.168.1.250`. Därefter körde jag `ip neighbour show` igen för att se om tabellen uppdaterats, vilket den hade, se Figur 7. Slutligen kan vi alltså konstatera att följande är de uppgifter som finns vad gäller andra maskiner på datorns nätverk:

1. 192.168.1.1, d2:21:f9:d1:13:52; Router
2. 192.168.1.244, 34:6f:24:18:7b:8d; Chromebook
3. 192.168.1.183, c6:79:3f:ab:59:bc; Macbook Pro

Jag har för övrigt verifierat att MAC-adressen stämmer överens med min Macbook Pro, se bilaga I.

```
[hig-25sipe01@dvg001:~$ ip neighbour show
192.168.1.1 dev enp2s0 lladdr d2:21:f9:d1:13:52 STALE
192.168.1.244 dev enp2s0 lladdr 34:6f:24:18:7b:8d STALE
192.168.1.183 dev enp2s0 lladdr c6:79:3f:ab:59:bc REACHABLE
```

Figur 7: Resultat från ny körning av `ip neighbour show` efter att sambon pingat labbmaskinen.

2.2.2. Del två – Fiktiv nätverkskonfiguration

Precis som i beskrivningen i 1.1 Bakgrund finns det två maskiner: maskin 1 (*m1*) och maskin 2 (*m2*). Dessa är anslutna till samma nätverk, detta nätverk har en router (*router*). För *m1* gäller att nodadressen är 15 och att den har en statisk inställning av nätverket *eth0*. Den andra maskinen (*m2*) har nodadressen 16 och har dynamisk inställning genom *dhcp* på *eth0*. Routern har adressen 192.168.133.162/25. Uppgiften är att utforma konfigurationen i `/etc/network/interface` för vardera maskin med *ipcalc* som grund.

Till att börja med används *ipcalc* för att se vad vi har att jobba med enligt följande, `ipcalc 192.168.133.162/25`, se Figur 8. Bara från detta vet vi nätmasken, som är 255.255.255.128. Eftersom den minsta möjliga adressen är 192.168.133.129 kan vi göra $129 + 15 - 1 = 143$, detta ger oss den IP vi söker för *m1* som är: 192.168.133.143. I uträkningen står +15 för att vi vill ha det femtonde numret för vår nod, och -1 för att vi räknar från den första giltiga IP-adressen som såklart också räknas med, genom att skriva ut allt hoppas jag det blir lättare att följa med i resonemanget. Vi kan med samma logik räkna ut vilken IP vi förväntar oss att *m2* ska få från DHCP-servern, $129 + 16 - 1 = 144$. Vi vet även från instruktionerna att routern har IP-adressen 192.168.133.162. Nedan finns Tabell 1 som redovisar vad vi kommit fram till, denna ligger till grund för nätverkskonfigurationerna.

```

hig-25sipe01@dvg001:~$ ipcalc 192.168.133.162/25
Address: 192.168.133.162      11000000.10101000.10000101.1 0100010
Netmask: 255.255.255.128 = 25 11111111.11111111.11111111.1 0000000
Wildcard: 0.0.0.127          00000000.00000000.00000000.0 1111111
=>
Network: 192.168.133.128/25  11000000.10101000.10000101.1 0000000
HostMin: 192.168.133.129    11000000.10101000.10000101.1 0000001
HostMax: 192.168.133.254    11000000.10101000.10000101.1 1111110
Broadcast: 192.168.133.255  11000000.10101000.10000101.1 1111111
Hosts/Net: 126              Class C, Private Internet

```

Figur 8: Resultat av ipcalc för routers adress.

Tabell 1: Lista över vardera maskin inklusive adress och nätmask.

| Maskin / Enhet | IP-adress | Nätmask |
|----------------|-----------------|-----------------|
| m1 | 192.168.133.143 | 255.255.255.128 |
| m2 | 192.168.133.144 | 255.255.255.128 |
| router | 192.168.133.162 | 255.255.255.128 |

2.2.2.1. Maskin 1 (m1)

Konfigurationen som syns i Figur 9 är en modifierad version av min labbmiljös *interfaces*-fil och grundar sig på föreläsningsmaterialet [2, pp 4] samt Tabell 1. Nämnvärt är att DNS-servrarna är från Cloudflare och är hämtade från deras dokumentation[3].

```

1 # This file describes the network interfaces available on your system
2 # and how to activate them. For more information, see interfaces(5).
3
4 source /etc/network/interfaces.d/*
5
6 # The loopback network interface
7 auto lo
8 iface lo inet loopback
9
10 # Ställer manuellt in nätverksinställningarna
11 auto eth0
12 iface eth0 inet static
13     address 192.168.133.143
14     netmask 255.255.255.128
15     gateway 192.168.133.162
16     dns-servers 1.1.1.1 1.0.0.1

```

Figur 9: Nätverkskonfiguration för m1. Bild genererad via ray.so för syntaxmarkering.

2.2.2.2. Maskin 2 (m2)

M2 bygger på samma tillvägagångssätt som m1 men innehåller istället konfigurationen som krävs för att DHCP ska göra grovjobbet, se Figur 10.

```
1 # This file describes the network interfaces available on your system
2 # and how to activate them. For more information, see interfaces(5).
3
4 source /etc/network/interfaces.d/*
5
6 # The loopback network interface
7 auto lo
8 iface lo inet loopback
9
10 # Hämtar automatiskt nätverksinställningar genom DHCP
11 auto eth0
12 iface eth0 inet dhcp
```

Figur 10: Nätverkskonfiguration för m2. Bild genererad via ray.so för syntaxmarkering.

3. Beskrivning av slutresultat

Resultatet är en ökad förståelse för hur hemmanätverket ser ut i samband med de olika deluppgifterna från del ett av laborationen. Yttermera har två nätverkskonfigurationer presenterats för det fiktiva nätverket.

4. Diskussion

Denna laboration upplevdes svårare än alla de tidigare laborationerna, förmodligen för att jag inte har full koll på nätverk. Även om det ofta blir jag som får stå vid routern och greja när något krånglar med internet här hemma så har jag länge saknat teorin bakom nätverk och dess uppbyggnad. Att sätta upp nätverksuttag med CAT-6 går bra men sedan tar kunskaperna ofta slut. Det jag kan sedan tidigare har jag plockat upp över flera år med hjälp av sökmotor och YouTube så detta var minst sagt en välkommen strukturerad introduktion. Föreläsningsmaterialet var mycket givande, dock upptäckte jag flera gånger att vissa meningar plötsligt avbryts vilket var lite tråkig och försvårade läsningen. Den kompletterande informationen i uppgiftsbeskrivningen för laborationen var till stor hjälp för det praktiska och gjorde det lätt att testa sig fram med föreläsningsmaterialet som komplement. Slutligen ser jag fram mot att utöka mina nätverkskunskaper i IPv6-delen.

5. Slutsatser

Efter laborationen har kunskapen för hemnätverket ökat. Dessutom visar laborationen på hur nätverksinställningar kan tas fram för såväl statisk som dynamisk (DHCP) IP-adress på en maskin som kör Debian.

6. Referenser

Litteraturförteckning

[1]: R. Hertzog, R. Mas, The Debian Administrator's Handbook, Debian Buster from Discovery to Mastery, 2020.

[2]: A. Jackson, "Introduktion till Linux och små nätverk" Föreläsning 4: Nätverk, och IPv4, 2024 [Online]. Tillgänglig: [https://hig.instructure.com/courses/8261/files/1441479?](https://hig.instructure.com/courses/8261/files/1441479?module_item_id=357684)

module_item_id=357684. Besökt: 2025-03-24.

[3]: Cloudflare, "IP addresses". [Online]. Tillgänglig: <https://developers.cloudflare.com/1.1.1.1/ip-addresses/>. Besökt: 2025-03-24.

I. Bilaga 1: MAC-adress Macbook Pro.

