Raspberry Pi setup til repeater styring

Kogebog for installation af Raspberry Pi processor kort, med relevant software, så den kan fungere som hjertet i en repeater station.

Der installeres og indstilles følgende:

1. Operativsystem Debian ”wheezy” download og installation på 4Gb SD kort
2. Grundlæggende installation af Raspberry Pi kort, tastatur – tidszone – password m.v
3. Oprettelse af brugerkonto Root
4. Sikker nedlukning af RaspPi
5. Godt SSH program
6. Opsætning af fast IP adr. på Ethernet port
7. Ændring af SSH port til privat nummer
8. Opsætning af SSH, med kodenøgle og ændret SSH port (kan udelades)
9. Opsætning af IPtables firewall
10. Installation USB memory stick til lagring af log filer
11. Installation af G4KLX D-Star programmer efter DL5DI metoden
12. Opsætning så D-Star aktivetet kan følges på tilkoblet monitor eller via SSH
13. Et par gode kommandoer
14. Konfigurationer

På det færdige image der kan hentes på www.dstar4all.dk er default password **HamRadio**

Bemærk at dette setup indtil videre er dækkende for D-Star repeater med det tyske DV-RPTR version 1 modul!!

Det er planen at afprøve andre setups

OZ5WU Michael
2013.01.16

Installation af operativsystem

Hent operativsystem på : www.raspberrypi.org , vælg: Soft-float Debian "wheezy" det er en .rar file, som skal ud pakkes fx med WinZip

Opret en mappe til udpakningen fx C:\Raspi og udpak det downloadede operativsystem til C:\Raspi

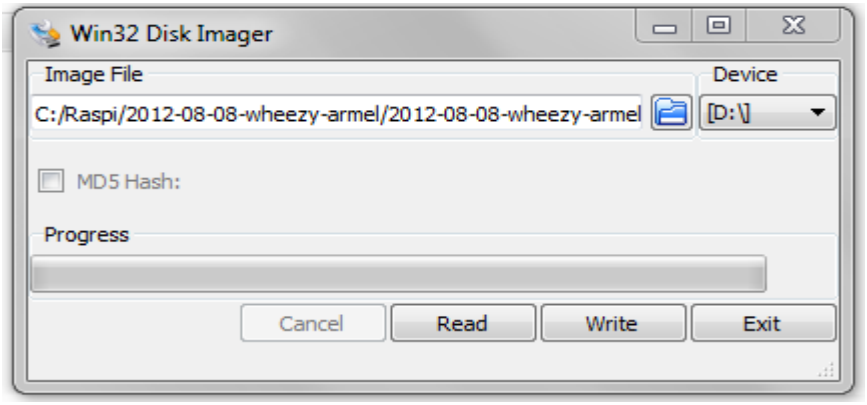
Da Debian ikke benytter samme filsystem som Windows skal der hentes endnu et lille værktøj i form af en SD manager, der kan kopiere OS filen til SD kortet.

Hent SD manager på: https://launchpad.net/win32-image-writer eller på : http://sourceforge.net/projects/win32diskimager/

Opret en mappe til SD tool, fx C:\Diskimag og udpak SD tool til denne mappe

Indsæt et tomt SD kort i PC'ens kortlæser og noter hvilket drev bogstav Windows tildeler kortet.

Start Win32DiskImager fra mappen og vælg drev bogstav for SD kort. Vælg derefter imagefil der skal kopieres til SD kort, filen findes i undermappe til C:\Raspi



Aktiver "Write" og vent på at image filen kopieres til SD kortet, når kopieringen er sket lukkes Win32 og SD kort frigives via "eject" media.

SD kortet kan nu monteres på din Raspberry Pi.

Opstart af RaspPi med nyt OS

Ved første opstart, tilsluttes skærm – tastatur og en god 5V USB forsyning (min. 1000mA).

Når vidunderet startes vil den skrive en masse fra boot forløbet for at ende i en ”initial” konfigurations menu. Menuen indeholder nogle vigtige punkter, der skal løbes igennem for at få et godt resultat.

a. Først aktiveres **expand_rootfs**, denne funktion re-partiotinerer SD kortet, så vi får adgang til hele SD kortet. Glemmes dette vil der kun være omkring 2Gb anvendeligt på kortet.
b. Dansk keyboard vælges med **configure_keyboard**, -> **Generic 105Key** -> **Other** -> **Danish**
c. PASSWORD bør skiftes med **change_pass**, vælg et PW indeholdende min 8 karakterer, med store og små bogstaver, samt tal.
d. Indstil **RAM split** til mindste video ram, 32Mb.
e. Indstil tidszone **change_timezone** til Europe -> Copenhagen.
f. Locale sættes til en_US.UTF-8, som default og kun den! Skal den rettes senere kan det gøres med: **dpkg-reconfigure locales**
g. Opdater OS
h. Afslut config program og RaspPi genstarter.

Er der senere behov for ændring af boot konfiguration kan programmet startes med:
sudo raspi-config eller blot **raspi-config** ved user root

Bruger konti

RaspPi er født med bruger kontoen ”Pi”. Men nogle installations pakker er bygget til at skulle eksekveres fra ROOT, dette gælder fx D-Star pakken. Derfor skal vi have oprettet ROOT konto

a. Kommandoen **sudo passwd root** eksekveres og root pasword vælges. Husk at notere det et sikkert sted ☺
b. Den nye konto afprøves, ved at skrive exit så vendes tilbage til login.

Det følgende udføres nemmest som bruger ”root”, så anbefalingen er at logge på nu som ”root”

Sikker nedlukning af RaspPi

Når en RaspPi computer skal lukkes ned, bør man ikke bare afbryde forsyningen, da dette kan skade / ødelægge SD kortets indhold, helt som det kendes fra windows.

Derfor lukkes ned med denne kommando: **shutdown –h now** eller blot **halt**

Når det kun er den røde power LED der lyser kan der slukkes for power til RaspPi

SSH klient

For at have nem adgang til Raspberry Pi, er det en god ting med en SSH klient som fx Bitvise

Bitvise kan hentes på: http://www.bitvise.com/ssh-client-download
Hent og installer programmet på den PC du skal benytte til SSH adgang til din RaspPi.

Opsætning af fast IP adr. på Ethernet port

Det kan være ganske upraktisk hvis IP adressen på netværks porten ændrer sig, fordi standard opsætning for porten er DHCP. Ændringen kan ske hvis forbindelsen til netværk har været afbrudt.
Dette kan imødegås ved at ændre port til fast IP, med følgende:

a. Først vi kende netværket RaspPi skal tilsluttes, dette gøre nemmest ved at logge på Internet router leveret af udbyder.

   Typisk setup vil vil være:
   IP adresse: 192.168.1.1 (GW) Gateway for tilsluttede enheder
   Netmaske: 255.255.255.0
   Broadcast: 192.168.1.255
   DHCP start IP adr.: 192.168.1.10 (eksempel!)
   DHCP slut IP adr.: 192.168.1.90 (eksempel!)
   Dette betyder at vi kan vælge ”faste” IP i området 192.168.1.100 til 192.168.1.254

   Er vi så heldige, at have en fast IP fra internet leverandør, findes info i klarmeldingen fra leverandøren.

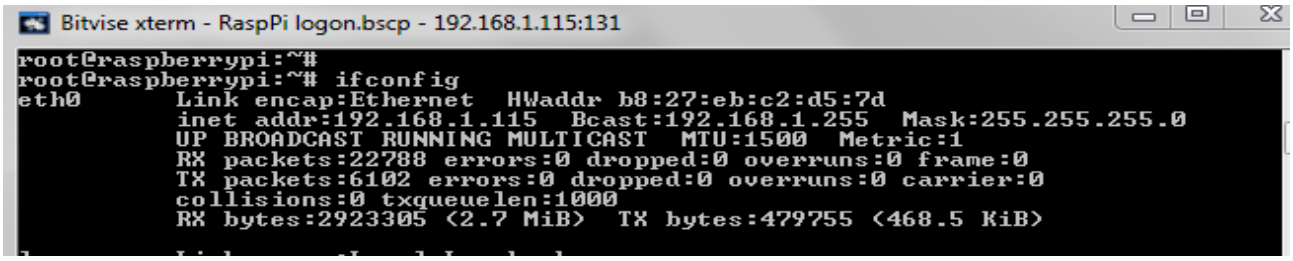
b. RaspPi etherport setup er gemt i en fil ”interfaces” som kan ændres med den indbyggede editor. Kommando: **nano /etc/network/interfaces** åbner config i teksteditoren indholdet erstattes med følgende tekst med tilpassede data:

```
#iface eth0 inet dhcp
# The loopback interface
 auto lo
 iface lo inet loopback
auto eth0
 iface eth0 inet static
 #your static IP
address 192.168.1.118
#your gateway IP
gateway 192.168.1.1
netmask 255.255.255.0
#your network address "family"
network 192.168.1.0
broadcast 192.168.1.255
```

   Ændringerne gemmes med **CTRL+O** -> **enter** -> **CTRL+X** og editoren lukkes med sidste kommando.

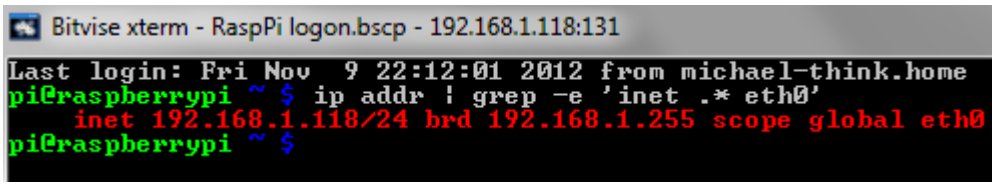
c. For at tage den ny IP adr. i anvendelse, skal netværks routinen genstartes, dette gøres med:
Kommando: **reboot** eller **/etc/init.d/networking restart**
Hvis man er connected via SSH, brydes forbindelsen nu.

d. Ændringen kan checkes med kommandoen **ifconfig** og resultatet ser således ud:



Kommandoen kan også benyttes til at se om status på porten, trafik og ganske vigtigt om der er pakketab på porten. Kommandoen **hostname –I** kan også benyttes.
Pakketab kan give R2D2 ☹

e. IP adr. rettes i SSH program, forbindelsen kan genetableres og voila vi har nu fast IP adresse på RaspPi. Check med: **ip addr | grep -e 'inet .* eth0'**



Ændring af SSH port nummer

Det kan være meget praktisk at ændre port nummer på SSH adgangen for at holde ubudne gæste bare lidt udenfor. Derfor skal vi ændre fra standarden port 22, til et privat port nummer. HUSK at vælge nummer som ikke benyttes til andre formål, se evt. på Wikipedia:
http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

Vælg et nummer der ikke er i rækken og noter det! I dette setup benyttes port 131

Dette gøres på følgende måde og ændringen virker efter reboot:

a. Kommando: **nano /etc/ssh/sshd_config** åbner sshd config i teksteditoren nano

b. Derefter findes : **# What ports, IPs and protocols we listen for** I starten af filen
I den næste linie står SSH port nummer : **Port 22** som rettes til den valgte **port 131**

Ændringerne gemmes med **CTRL+O** -> **enter** -> **CTRL+X** og editoren lukkes med sidste kommando.

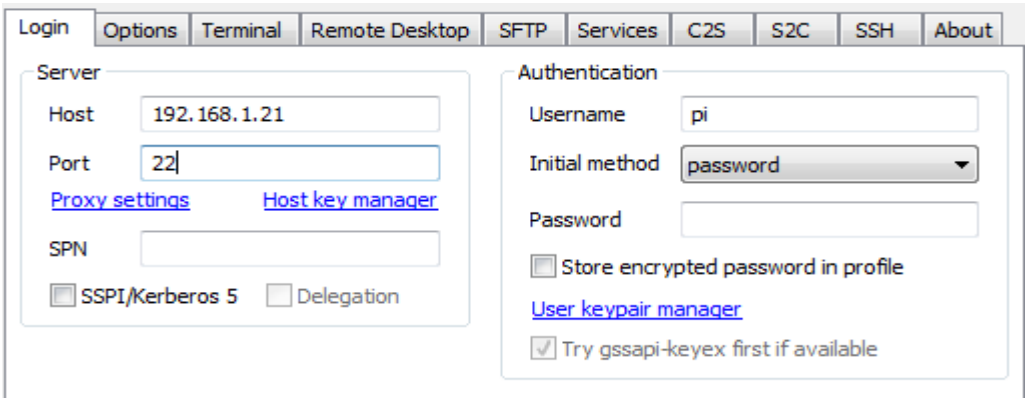
Opsætning af SSH, til adgang med kodenøgle

Dette afsnit er ikke standard og kan springes over. Men man slipper for at skulle taste password hver gang hvis nøglefunktionen tilføjes ☺

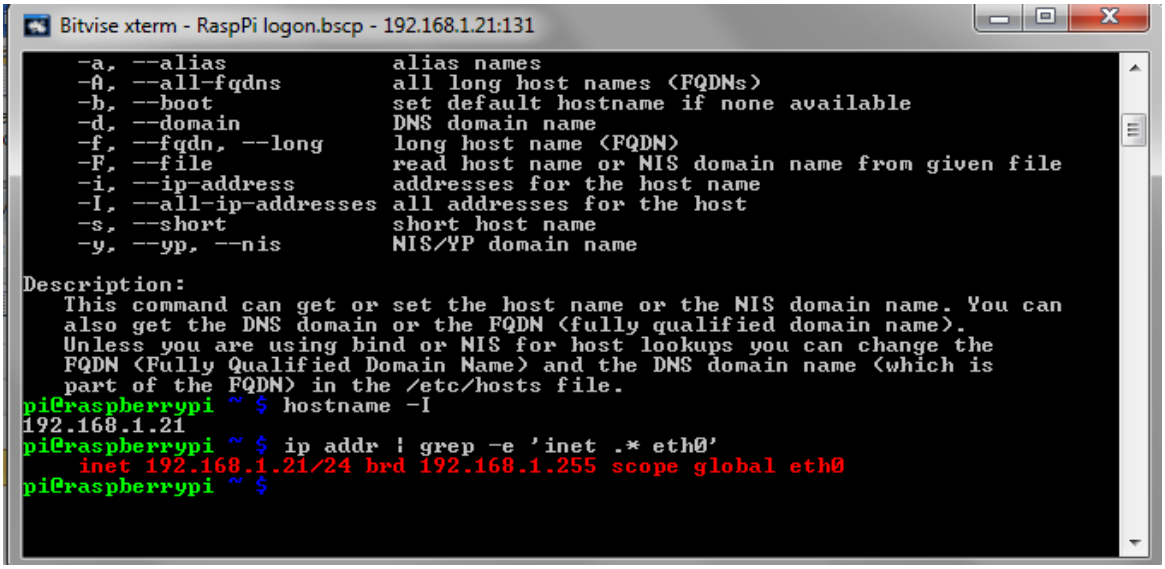
Operativ systemet er født med SSH funktion enablet, via standard port 22 (nu port 131) og med brug af maskinens password. Dette synes jeg er for simpelt hvis maskineriet skal være tilsluttet "Den stygge sky" 24/7/365. Derfor er her en opskrift på at gøre verden noget mere sikker.

For at benytte kodenøgle i en SSH session skal vi benytte en SSH klient, som kan dette fx Bitvise som i tilgift har funktion til at generere vores kodenøgle.

a. Bitvise startes, IP adr. fra RaspPi samt bruger og password indtastes og der etableres SSH session til RaspPi enheden med "Log-in"

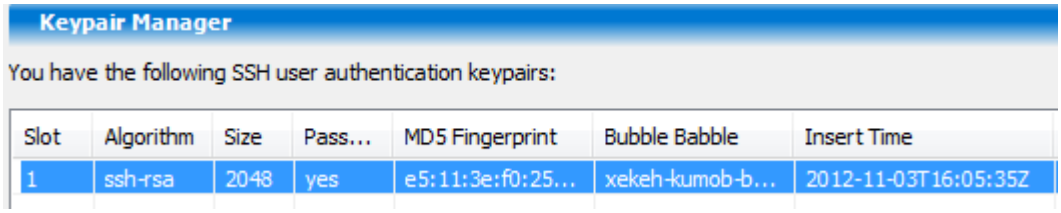


Ved forbindelse åbnes et ny Xterm vindue på PC'en:



Og som tilgift åbnes en SFTP applikation, der kan benyttes til at kopiere fx log filer fra RaspPi, den er meget rar til fx at hente log filer, hvis noget driller, eller ryde op i logfiler m.v

b. Nu skal Nøglefilen til den forbedrede SSH genereres, dette gøres ved at aktivere "User keypair manager" i login billedet, under en opkoblet SSH forbindelse.
Nøglefilen er en 2048bit rsa nøgle. Som genereres ved hjælp af "**Generate new…**"



Nøglefilen eksporteres og gemmes på et passende sted på PC'en, vi skal bruge den om lidt, når RaspPi skal lære den. Filnavn er default : **publickeyopenssh**.txt

c. Så skal vi have ændret SSH opsætningen på RaspPi til at benytte nøgle.
Der benyttes SSH forbindelse til RaspPi, så vi kan copy paste nøglefilen ombord uden problemer.
Der er en række kommandoer der skal eksekveres på RaspPi for at gøre ændringen:

cd~ sikrer at "home" dir for brugeren benyttes
pwd "print working directory" viser hvor vi er i filsystemet ok svar er : /home/pi
mkdir .ssh Etablerer en folder med navn .ssh
cd .ssh skifter til folderen .ssh
nano authorized_keys åbner tekst editor på RaspPi med filnavn **authorized_keys**

Sekvensen ser således ud:

```
pi@raspberrypi ~ $ cd ~
pi@raspberrypi ~ $ pwd
/home/pi
pi@raspberrypi ~ $ mkdir .ssh
pi@raspberrypi ~ $
pi@raspberrypi ~ $ cd .ssh
pi@raspberrypi ~/.ssh $ nano authorized_keys
pi@raspberrypi ~/.ssh $
```

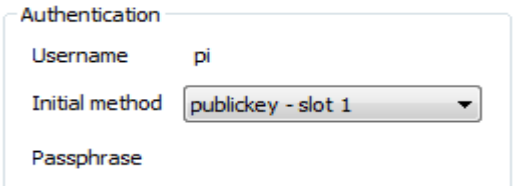
Nøglefilen fra før kopieres ind i teksteditor vinduet.
Filen gemmes med **CTRL+O** -> **enter** -> **CTRL+X** og editoren lukkes med sidste kommando.
Derefter skal følgende kommandoer eksekveres:

ls -l viser indhold I bibliotek .ssh, tilladelser og ejerskab
chmod 700 ~/.ssh/ tilladelser for .ssh bibliotek ændres for at SSH program kan benytte
chmod 600 ~/.ssh/authorized_keys tilladelser for key file
ls –l viser indhold I bibliotek .ssh, tilladelser og ejerskab efter ændring

Dette skal se sådan ud:

```
pi@raspberrypi ~/.ssh $ ls -l
total 4
-rw-r--r-- 1 pi pi 409 Sep 17 20:05 authorized_keys
pi@raspberrypi ~/.ssh $ chmod 700 ~/.ssh/
pi@raspberrypi ~/.ssh $ chmod 600 ~/.ssh/authorized_keys
pi@raspberrypi ~/.ssh $ ls -l
total 4
-rw------- 1 pi pi 409 Sep 17 20:05 authorized_keys
pi@raspberrypi ~/.ssh $
```

d. Næste trin er at afprøve den nye nøgle. Obs. lad den nuværende SSH forblive åben, så kan eventuelle fejl rettes online.
Dette gøres ved at åbne endnu en SSH session, denne gang benyttes igen port 131 + nøgle, Authentication feltet skal se ud som:



Kan du kalde kortet med nøglen er alt til nu i orden.

e. Næste step er at fjerne password adgangen, så RaspPi kun kan nås på SSH med nøglen
Dette gøres ved at rette i sshd_config.

Kommando: **nano /etc/ssh/sshd_config** åbner sshd config i teksteditoren

Vi skal nu finde en linie med følgende indhold : **#PasswordAuthentication yes**
Den findes ca. midt på side 2 (CTRL-V).
Linien rettes til : **PasswordAuthentication no**

Rettelserne gemmes med **CTRL+O** -> **enter** -> **CTRL+X** og editoren lukkes med sidste kommando.

Rettelserne træder først i kraft efter kommandoen: **/etc/init.d/ssh restart** eller **reboot**

ADGANGEN ER BRUGER (Pi / root / peter) relateret, og skal således gentages pr. bruger. Dette kan være med forskellige nøgler, eller med en fælles

Opsætning af IPtables firewall

Da de fleste D-star repeatere sidder direkte på "Den Stygge Sky" via en dum router fra en Internet leverandør, er det en rigtig god ide at sikre sig lidt mere mod ubudne gæster. Dette kan gøres relativt simpelt med følgende program og lidt kommandoer.

Det forudsættes at man er user "**root**" i det følgende:

a. OS kernen skal opdateres, med nyeste CA(Certificate Authority) certificater og der skal installeres forskellige ekstra komponenter.
   Følgende kommandoer skal eksekveres:
   cd ~ sikrer at "home" dir for brugeren benyttes .
   apt-get install ca-certificates opdatering af certificater
   wget http://goo.gl/1BOfJ -O /usr/bin/rpi-update && sudo chmod +x /usr/bin/rpi-update
   henter Hexxeh rpi-update program
   apt-get install git-core installeres git-core, for adgang til seneste UNIX build
   rpi-update opdaterer til seneste UNIX build, dette tager nogle min.!!
   reboot for at få ændringerne til at virke.

b. Næste trin er at finde gateway ip på det net RaspPi er tilsluttet. Denne øvelse er gjort hvis Raspberry Pi har fået fast IP-adr. afsnit er, ellers må man lige logge på Internet routeren. Typisk vil GW være 192.168.1.1

c. Så skal vi have samlet en liste med port numre, der må have adgang til vores RaspPi. Listen skal indeholde definitioner for alle de port numre der er nødvendige for drift og opdatering af en repeater.
   Port numre med funktion er samlet i denne tabel:

Funktion	UDP	TCP
Generelle		
HTTP		80
SSH remoteacc.		131
Dstar		
Remote control	10131	
APRS	14580	
DPLUS Linking	20001	
DEXTRA Linking	30001	
DCS Linking	30051	
G2 Callsign Routing	40000	40000

d.

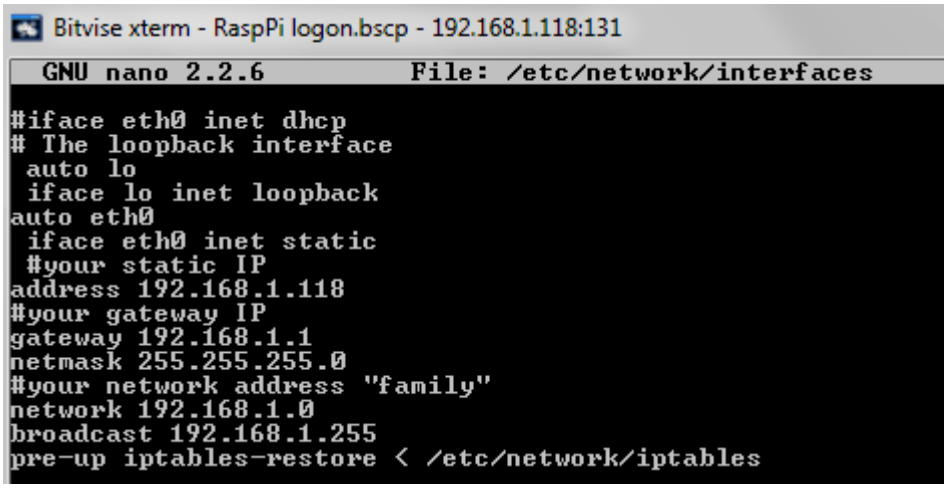
Der skal oprettes en fil til firewall reglerne, dette gøres med følgende kommando:

sudo bash -c 'iptables-save > /etc/network/iptables'

e. Netværksporten Eth0 skal ændres, så netværks trafik sendes igennem vores firewall, dette gøres ved at tilføje ny linie i Eth0 setup.
Kommando: **nano /etc/network/interfaces** åbner config i teksteditoren
Den følgende linie indsættes, som ny linie sidst i tekstfilen:

pre-up iptables-restore < /etc/network/iptables

konfigurationen ser sådan ud, her med fast IP og firewall installeret:



Ændringerne gemmes med **CTRL+O** -> **enter** -> **CTRL+X** og editoren lukkes med sidste kommando.

f. Firewall rule filen skal nu tilføjes indhold, dette gøres med teksteditoren nano, følgende eksekveres:
nano /etc/network/iptables

Reglerne bygges nemmest i fx note-pad og gemmes som rules.txt på PC. Regel indholdet ser ud som følger:

***filter**
:INPUT DROP [23:2584]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [1161:105847]
-A INPUT -i lo -j ACCEPT
-A INPUT -i eth0 -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -i eth0 -p tcp -m tcp --dport 131 -j ACCEPT
-A INPUT -i eth0 -p udp -m udp --dport 10131 -j ACCEPT
-A INPUT -i eth0 -p udp -m udp --dport 14580 -j ACCEPT
-A INPUT -i eth0 -p udp -m udp --dport 20001 -j ACCEPT
-A INPUT -i eth0 -p udp -m udp --dport 30001 -j ACCEPT

-A INPUT -i eth0 -p udp -m udp --dport 30051 -j ACCEPT
-A INPUT -i eth0 -p tcp -m tcp --dport 40000 -j ACCEPT
-A INPUT -i eth0 -p udp -m udp --dport 40000 -j ACCEPT
-A INPUT -s 192.168.1.0/24 -j ACCEPT
-A INPUT -s 192.168.1.1/32 -i tcp -p tcp -m tcp --dport 22 -j DROP
-A INPUT -p icmp -m icmp --icmp-type 8 -j ACCEPT
-A INPUT -i eth0 -m state --state RELATED,ESTABLISHED -j ACCEPT
COMMIT

Teksten kopieres over i nano editor vinduet, husk at have rettet IP-adr. så de passer på de lokale forhold.

Ændringerne gemmes med **CTRL+O** -> **enter** -> **CTRL+X** og editoren lukkes med sidste kommando.

g. Nu skal firewall loades med vores regler, dette gøres med denne kommando:

iptables-restore /etc/network/iptables

Nu er FW installeret og er der stadig SSH forbindelse og kan man pinge RaspPi IP adr. , virker tingen!
En test kan være at fjerne : **-A INPUT -i eth0 -p tcp -m tcp --dport 80 -j ACCEPT**
og reloade iptables, så må man ikke kunne til gå repeaterens hjemmeside!
HUSK at tilføje igen!!

h. Firewall reglerne kan altid ses på et kørende system, med denne kommando:

iptables-save eller bedre **iptables-apply /etc/network/iptables**

i. Med kommandoen : netstat -anp | grep ircddbgateway ses hvad der er koblet op

Hermed har vi sikret RaspPi mod de fleste forsøg på uønskede ”besøg”

Installation USB memory stick til lagring af log filer

Det kan være ganske praktisk at log filerne fra fx D-Star programmerne ikke skrives på SD kortet, derfor kan disse filer lægges på en USB memory stick, som installeres på følgende måde:

Det forudsættes at man er user "**root**" i det følgende:

a. Kommandoen: **tail -f /var/log/messages** eksekveres for at finde drev navn og giver dette :

```
pi@raspberrypi ~ $ tail -f /var/log/messages
Nov 10 11:36:53 raspberrypi kernel: [36143.001493] usb 1-1.2: Product: Cruzer Ed
ge
Nov 10 11:36:53 raspberrypi kernel: [36143.001505] usb 1-1.2: Manufacturer: SanD
isk
Nov 10 11:36:53 raspberrypi kernel: [36143.001517] usb 1-1.2: SerialNumber: 2005
49634004F58171F8
Nov 10 11:36:53 raspberrypi kernel: [36143.007019] scsi0 : usb-storage 1-1.2:1.0
Nov 10 11:36:54 raspberrypi kernel: [36144.001326] scsi 0:0:0:0: Direct-Access
   SanDisk  Cruzer Edge      1.26 PQ: 0 ANSI: 5
Nov 10 11:36:54 raspberrypi kernel: [36144.006523] sd 0:0:0:0: [sda] 15633408 51
2-byte logical blocks: (8.00 GB/7.45 GiB)
Nov 10 11:36:54 raspberrypi kernel: [36144.007984] sd 0:0:0:0: [sda] Write Prote
ct is off
Nov 10 11:36:54 raspberrypi kernel: [36144.009108] sd 0:0:0:0: [sda] Write cache
: disabled, read cache: enabled, doesn't support DPO or FUA
Nov 10 11:36:54 raspberrypi kernel: [36144.021643]  sda: sda1
Nov 10 11:36:54 raspberrypi kernel: [36144.025841] sd 0:0:0:0: [sda] Attached SC
SI removable disk
```

USB stick sætte på RaspPi og tail rapporterer sidste messages, i log der holder rede på bl.a I/O devices.
I vores tilfælde ses USB stick ID i næst sidste linje, som **sda: sda1** hvor **sda1** er drev navn.
I CTRL C stopper tail.

b. Der skal oprettes et directory til vores USB stick, dette gøres med følgende kommando:

mkdir /usbstick

c. Så skal vi have alle RaspPi brugere til at kunne benytte USB drevet, dette gøres ved at tilrette filen **fstab** i folderen **/etc**
Editor fx nano benyttes, kan startes med: **nano /etc/fstab**
Følgende linie indsættes nederst i filen, gerne lige over # **:**

/dev/sda1 /usbstick vfat auto,users,noatime,umask=0 0 0

```
  GNU nano 2.2.6                File: /etc/fstab

proc            /proc           proc    defaults          0       0
/dev/mmcblk0p1  /boot           vfat    defaults          0       0
/dev/mmcblk0p2  /               ext4    defaults,noatime  0       0
/dev/sda1 /usbstick             vfat auto,users,noatime,umask=0 0 0
# a swapfile is not a swap partition, so no using swapon|off from here on, use $
```

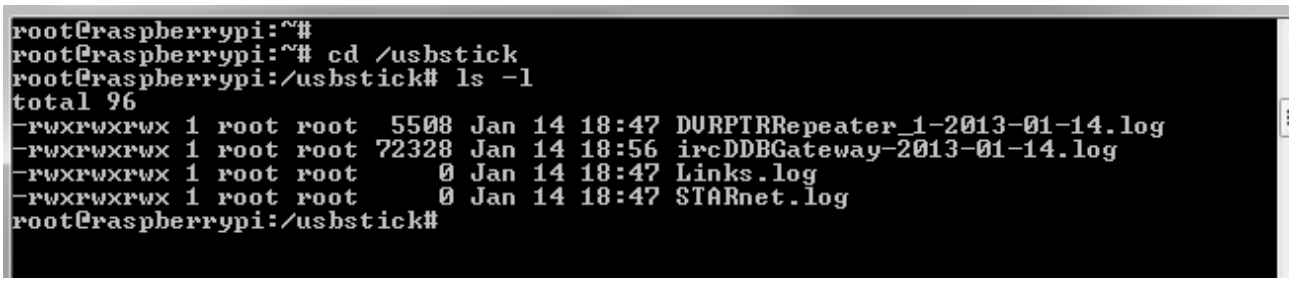
Ændringerne gemmes med **CTRL+O** -> **enter** -> **CTRL+X** og editoren lukkes med sidste kommando.

System rebootes for at ændringen virker, kommando: **reboot**

d. Nu kan vi afprøve om drevet er tilstede, med disse kommandoer:

cd /media
ls -l

Dette giver en oversigt over indholdet på USB drevet:



e. BEMÆRK konfigurations log fra ircddbgateway og repeater stadig skrives i:
/var/log/opendv

f. Som udgangspunkt er USB stick fast monteret på RaspPi.Skal den afmonteres, skal de processer der benytter stick stoppes (Ircddbgw log og repeater_1 log), derefter kan USB stick fjernes. Der vil komme en fejlmelding ved reboot af RaspPi uden USB stick monteret.

Installation af G4KLX D-Star programmer.

Så er vi klar til at installere selve repeater programmet, det er den komplette pakke der installeres. Her benyttes DL5DI's opskrift som direkte er "lånt" fra ircDDBGateway-DEB-instructions-ENG-20120918.pdf

Husk at logge på som bruger **root**

Så skal følgende punkter afvikles:

a. Der dannes en forbindelse til den server, som program og opdateringer findes på ude i skyen. Dette gøres med denne kommando:

   curl ftp://141.75.245.226:8021/debian/opendv.list -o /etc/apt/sources.list.d/opendv.list

b. Så skal der oprettes et midlertidigt sted til programmeerne og det skal hentes.Dette gøres med følgendekommandoer:

   cd /tmp
   wget ftp://141.75.245.226:8021/debian/dl5di.pk
   apt-key add dl5di.pk

c. Derefter skal der hentes opdateringer, dette gøres med:

   apt-get update

d. Så skal repeater delen installeres, dette gøres med:

   apt-get install repeater

   Stop konfigurationen når scriptet spørger, med mindre det er en rent stand alone repeater unden gateway.

e. Så skal gateway programmet installeres, installationen starter samtidig en konfigurations menu, hvor selve opsætningen af gateway og repeater klares. Dette gøres med:

   apt-get install ircddbgateway

   I selve installations menu, kan man i store træk nøjes med at rette callsign, ircddb password og APRS server. Selve repeater modul konfigureres. Værdierne for de enkelte punkter kan ses i i

skemaerne bagerst i dette dokument.
Når dette punkt er gennemført, er installationen færdig…

Opsætning så D-Star aktivitet kan følges på skærm eller SSH.

Det kan være en rar ting at kunne se hvad der foregår live på en repeater i fejlsøgnings øjemed.
Dette kræver at log er enablet, både på gateway og på repeater modul.

f. Ircddb konfiguration startes med **ircddbgw_conf** og logning starts under punkt 15, med "0" som parameter. Default sti bibeholdes. Timeserver benyttes ikke pt. gateway reloades med kommando 32.
g. Repeater konfiguration startes med **repeater_conf 1** og logning startes under punkt 4, med "0" som parameter. Default sti bibeholdes. Audio log benyttes ikke. Repeater genstartes med kommando 32.
h. For at få live log på skærmen på en nem måde, skal der opfindes et par alias kommandoer

   dvstatus til at få vist repeaters live log.

   gwstatus til at få vist gateways live log.

   Dette skal gøres i en fil i root directory, der hedder **.bashrc**

   Vi skal lige sikre at det er den rigtige fil vi tager fat i dette gøres med : **echo $SHELL** som meget gerne skal svare med /bin/bash

   Filen åbnes med kommandoen **nano ~/.bashrc** Dermed åbnes nano tekst editor med den ønskede fil.
   Disse to linier kopieres ind nederst i filen:
   Med USB stick:
   alias dvstatus=' tail -f /usbstick/DVRPTRRepeater_1-`date -u +%Y-%m-%d`.log'
   alias gwstatus='tail - f /usbstick/ircDDBGateway-`date -u +%Y-%m-%d`.log'

   Uden USB stick:
   alias dvstatus=' tail -f /var/log/opendv/DVRPTRRepeater_1-`date -u +%Y-%m-%d`.log'
   alias gwstatus='tail -f /var/log/opendv/ircDDBGateway-`date -u +%Y-%m-%d`.log'

   Ændringerne gemmes med **CTRL+O** -> **enter** -> **CTRL+X** og editoren lukkes med sidste kommando.

i. Så skal RaspPi genstartes, hvilket nemmest gøres med **reboot**

j. Når RaspPi er oppe igen, kan repeater live status startes med **dvstatus** og afsluttes med **CTRL+C**
   Og gateway status kan ses med **gwstatus** og afsluttes med **CTRL+C**

Et par gode kommandoer

exit	benyttes til at skifte til ny bruger
halt	lukker RaspPi ned
ifconfig	Viser interface konfigurationen
uptime	Som fortæller oppetid, men næsten vigtigst gennemsnitlig load.