1. Elenata auf Raspberry 3, 4

Das Image muss auf eine SD von mindestens 16GB geschrieben werden.

1. Simplexnode-Betrieb, nur ein TRX an der Elenata mit SQL-PIN
2. Anschlüsse **eines** TRX (FT7800, FT8800, FTM100, FTM400) in Richtung Elenata:  
   RX an L-RX  
   TX an L-TX  
   SQL an L-SQL  
   PTT an L-PTT  
   GND an GND
3. Sollte kein SQL vom TRX zur Verfügung stehen dann in der svxlink.conf auf VOX umstellen
4. Netzwerkzugang via LAN
5. Raspberry  
   User: pi  
   PW: raspberry  
   Passwort mit passwd bitte ändern
6. Anpassung der svxlink.conf  
   sudo nano /etc/svxlink/svxlink.conf  
     
   [SimplexLogic]  
   CALLSIGN=DeinCall  
     
   [ReflectorLogic]

TYPE=Reflector

HOST=master1.thueringen.link

PORT=5308

CALLSIGN="xxxxxxxxx"

AUTH\_KEY="xxxxxxxxx"

AUDIO\_CODEC=OPUS

OPUS\_ENC\_BITRATE=64000

JITTER\_BUFFER\_DELAY=100

DEFAULT\_TG=xx

MONITOR\_TGS=xx,xx  
  
[LocationInfo]

APRS\_SERVER\_LIST=euro.aprs2.net:14580

STATUS\_SERVER\_LIST=aprs.echolink.org:5199

LON\_POSITION=12.10.30E

LAT\_POSITION=52.20.10N

CALLSIGN=EL-XXXXXX

FREQUENCY=xxx.xxx

TX\_POWER=1

ANTENNA\_GAIN=x

ANTENNA\_HEIGHT=xxm

ANTENNA\_DIR=-1

1. Anpassung der ModuleEchoLink.conf

sudo nano /etc/svxlink/svxlink.d/ModuleEchoLink.conf  
  
[ModuleEchoLink]

NAME=EchoLink

ID=2

TIMEOUT=60

#MUTE\_LOGIC\_LINKING=1

ALLOW\_IP=xxx.xxx.xxx.xx/24

#DROP\_ALL\_INCOMING=0

#DROP\_INCOMING=^()$

#REJECT\_INCOMING=^()$

#ACCEPT\_INCOMING=^(.\*)$

#REJECT\_OUTGOING=^()$

#ACCEPT\_OUTGOING=^(.\*)$

#REJECT\_CONF=0

#CHECK\_NR\_CONNECTS=2,300,120

SERVERS=europe.echolink.org

CALLSIGN=xxxxxx-L

PASSWORD=xxxxxxxxxxxx

SYSOPNAME=xxxxx

LOCATION=[Svx] xxxxxxxxxxx

#PROXY\_SERVER=the.proxy.server

#PROXY\_PORT=8100

#PROXY\_PASSWORD=PUBLIC

#BIND\_ADDR=10.20.30.40

MAX\_QSOS=10

MAX\_CONNECTIONS=11

LINK\_IDLE\_TIMEOUT=300

#AUTOCON\_ECHOLINK\_ID=9999

#AUTOCON\_TIME=1200

#USE\_GSM\_ONLY=1

DEFAULT\_LANG=de\_DE

#COMMAND\_PTY=/dev/shm/echolink\_ctrl

REMOTE\_RGR\_SOUND=0

1. Anpassung der node\_info.json  
   sudo nano /etc/svxlink/node\_info.json

"Location": "xxxxxxx",

"Locator": "xxxxxxx",

"SysOp": "xxxxxxx",

"LAT": "52.528",

"LONG": "13.141",

"TXFREQ": "144.825",

"RXFREQ": "144.825",

"Website": "https://thueringen.link",

"Mode": "FM",

"Type": "2",

"Echolink": "xxxxxx",

"nodeLocation": "xxxxxxxx",

"Sysop": "xxxxxxxxxx",

"Verbund": "Thueringen.link",

"CTCSS": ""

}

1. Ansage Temp. aus aprs.fi

Alle dreissig Minuten erfolgt die Durchsage des Rufzeichens inkl. der Ausgabe der Temperatur einer Wetterstation, die auf aprs.fi Wetterdaten anzeigt. Welche Wetterstation eingetragen wird ist eigentlich egal. Wichtig ist nur, dass man zur Abfrage der Daten seinen API-Key verwendet den man sich nach dem Einloggen auf der Seite aprs.fi im Bereich „Einstellungen“ anzeigen lassen kann. Dazu muss man sich vorher bei aprs.fi registrieren.

Man kann aber auch auf den Raspi auf dem svxlink läuft, weewx installieren und dann über den USB-Anschluss eine eigene Wetterstation anschließen, welche Wetterdaten an aprs.fi sendet. Diese Daten können dann für die Ansage der Temperatur in svxlink eingebunden werden.

Anpassung der Wetterscript-Datei „wetter.pl“  
sudo nano /etc/svxlink/scripts/wetter.pl  
  
#!/usr/bin/perl

#

# kleines Programm zum Auslesen von Wetterdaten von aprs.fi

#

# Adi Bier / DL1HRC

#

# V0.03 liest Temperatur von aprs.fi

#

use LWP::UserAgent;

# Hier steht das Call der Wetterstation welche bei aprs.fi abgefragt werden soll

$call = "call";

# Hier wird dein API-Key eingetragen den du in den Einstellungen deinen Accounts bei aprs.fi findest

$apikey = "xxxxxxxxxxxxxxxxxxxxxx";

$owninfo = "SvxLink - no website";

1. Intervall Abfrage Wetterdaten erfolgt in der Datei crontab  
     
   sudo nano /etc/crontab  
     
   Hier brauch im Regelfall nichts verändert werden.
2. Abfrage Systemsattus svxlink wie folgt:  
   tail -f --lines=500 /var/log/svxlink