AVM Fritz!Box 7362 SL mit OpenWRT

Quellen:

- OpenWRT Homepage
- OpenWRT Wiki Seite zur Fritz!Box 7362 SL
- VLAN Konfiguration unter OpenWRT
- Extending the router ports with a managed switch with VLANs
- Ethernet Switch Dokumentation
- VLAN einrichten auf WRT1200ACv2 und OpenWRT
- FreiFunk Berlin: Switch VLAN Konfiguration
- OpenWRT Network Configuration, mit gutem Blockdiagramm

Installation

- FRITZ!Box 7362 SL (1&1 Branding) auf Werkseinstellungen zurück setzen oder Firmware Recovery durchführen

FRITZ!Box 7362 SL LAN 1 und PC via Switch verbinden und PC NIC auf 192.168.1.2 einstellen, Recovery nach AVM Anleitung durchführen

Firmware Recovery

```
wine FRITZ.Box_7362_SL-07.18-recover.exe
```

- FRITZ!Box starten und sofort eva_ramboot.py ausführen

eva_ramboot.py

```
#!/usr/bin/env python3

import argparse

from ftplib import FTP
from os import stat

parser = argparse.ArgumentParser(description='Tool to boot AVM EVA
ramdisk images.')
parser.add_argument('ip', type=str, help='IP-address to transfer the
image to')
parser.add_argument('image', type=str, help='Location of the ramdisk
image')
```

```
parser.add_argument('--offset', type=lambda x: int(x,0), help='Offset
to load the image to in hex format with leading 0x. Only needed for
non-lantiq devices.')
args = parser.parse_args()

size = stat(args.image).st_size
# arbitrary size limit, to prevent the address calculations from
overflows etc.
assert size < 0x2000000

if args.offset:
    addr = size
    haddr = args.offset
else:
    # We need to align the address.
    # A page boundary seems to be sufficient on 7362sl and 7412
    addr = ((0x8000000 - size) & ~0xfff)
    haddr = 0x80000000 + addr

img = open(args.image, "rb")
ftp = FTP(args.ip, 'adam2', 'adam2')

def adam(cmd):
    print("> %s"%(cmd))
    resp = ftp.sendcmd(cmd)
    print("< %s"%(resp))
    assert resp[0:3] == "200"

ftp.set_pasv(True)
# The following parameters allow booting the avm recovery system with
this
# script.
adam('SETENV memsize 0x%08x'%(addr))
adam('SETENV kernel_args_tmp mtdram1=0x%08x,0x88000000'%(haddr))
adam('MEDIA SDRAM')
ftp.storbinary('STOR 0x%08x 0x88000000'%(haddr), img)
img.close()
ftp.close()
```

```
python3 eva_ramboot.py 192.168.178.1 openwrt-24.10.4-lantiq-xrx200-
avm_fritz7362sl-initramfs-kernel.bin
# wait for FRITZ!Box to reboot new kernel
# change IP to 192.168.1.1
# when successfull ping 192.168.1.1, transfer sysupgrade to FRITZ!Box (as
root!)

scp -O openwrt-24.10.4-lantiq-xrx200-avm_fritz7362sl-squashfs-sysupgrade.bin
root@192.168.1.1:/tmp
# log in via ssh as root
```

```
ssh root@192.168.1.1
# start sysupgrade
sysupgrade /tmp/openwrt-24.10.4-lantiq-xrx200-avm_fritz7362sl-squashfs-
sysupgrade.bin
# wait for reboot
```

Open LuCi web interface via http://192.168.1.1/

Grundlegendes

Nach der erfolgreichen Installation kann man sich per SSH auf der Fritz!Box einloggen.

```
ssh -l root <IP-DER-FRITTE>
```

```
# Zeige alle physischen und virtuellen Netzwerkschnittstellen
ls -l /sys/class/net

# Netzwerkkonfiguration anzeigen
cat /etc/config/network

# Pakete updaten
opkg update && opkg list-upgradable | cut -f 1 -d ' ' | xargs opkg upgrade

# Midnight Commander installieren
  opkg install mc

# Netzwerk neu starten
  /etc/init.d/network restart
# oder
  service network reload

# Interface-Liste anzeigen:
ubus list network.interface.*

network.interface.lan
network.interface.loopback
network.interface.wan
network.interface.wan6

# physische Switch Konfiguration anzeigen
swconfig list
Found: switch0 - eth0

swconfig dev switch0 show
```