



FUJITSU

Fujitsu FUTRO S700 Thin Client

für Freifunk nutzen

How to...



Version: 2

Stand: 25. August 2018

Futro S700

für Freifunk nutzen.....	1
How to.....	1
Futro S700 - Was ist das	3
Ausstattung	3
Einsatz im Freifunk-Bereich.....	4
Ziel: Freifunk-Image aufspielen	4
Möglichkeit : OS von USB Stick booten	4
ToDo: Einen Rechner unter Knoppix starten.....	5
Futro mit Knoppix vom Stick starten.....	5
Das richtige Freifunk-Futro Image.....	6
Wie bekomme ich die Datei auf den Futro?.....	7
Freifunk-Image rennt.....	7
Freifunk-Konfiguration	8
Futro geht Online	8
Einrichtung der Futro-Konfig.....	9
Möglichkeit 1: Futro: alles über eine Netzwerkkarte (VLANs).....	9
<i>Möglichkeit 2:</i> Wird hier nicht näher behandelt, weil:	9
Futro: Wir editieren die NETWORK Datei.....	13
Router: Wir editieren die NETWORK Datei	14
Anlagen.....	16
Appendix – 1 - Konfig vom Futro	16
Appendix – 2 - Konfig vom Router (Portnummern für TP-Link 841N).....	17
Appendix – 4 - Konfig vom Router (Portnummern für TL-WDR3600).....	19
Appendix – 5 – Diagnosebefehle	20

Futro S700 - Was ist das

Was der Hersteller schreibt:

Der Fujitsu FUTRO S700 ist ein individuell konfigurierbarer Thin Client für server-basiertes Computing und Desktop Virtualisierung von standardisierten Büroanwendungen. Dieses einfache und zuverlässige Front-end-Zugangsgerät lässt sich bequem integrieren, nutzen und verwalten. Ihre alltägliche Produktivität wird durch zusätzliche, bequeme Funktionen wie Dual Monitoring, schnelles Hochfahren und geringe Geräuschemissionen erhöht. Die für Fujitsu-Produkte typische, hohe Qualität, die aktuelle Flash Drive-Technologie und ausgezeichnetes Wärmemanagement sorgen für eine lange Nutzungsdauer dieser Thin Clients. In Verbindung mit hervorragender Energieeffizienz führt dies zu geringen Gesamtbetriebskosten für Ihr Unternehmen.

Was ein Freifunker schreibt:

Geiles Teil, übernimmt die CPU-Intensive Rechenarbeit einer VPN Verbindung. Da muss viel und möglichst schnell verschlüsselt werden. Genau die richtige Aufgabe für diesen Mini-Rechner. Benötigt wenig Watt (10-max. 20 Watt je nach Last), also ca. soviel wie zwei oder drei Router.

Ausstattung

Prozessor	AMD G-Series G-T44R (1.2 GHz)
Betriebssystem	eLux® RL - Embedded Linux
Arbeitsspeicher	1 x 1 GB, DDR3, 1066 MHz
Speicherkapazität	2 GB (1 Speichersteckplätze gesamt)
Festplatte	1 x 2 GB, mSATA, Flash-Laufwerk
Chipsatz	AMD A55E Controller-Hub
Grafik	AMD Radeon™ HD 6250, bis zu 256 MB
	DirectX 11, Shader Model 5, Open CL™ 1.1 (abhängig von CPU), OpenGL 4.1
Audio	On Board, Realtek ALC262, AC'97 Codec
	1 x Audio: Line-Eingang, 1 x Audio frontseitig: Line-Eingang/Mikrofon
	1 x Audio: Line-Ausgang
Konnektivität	LAN: Integriertes 10/100/1000 Mbit/s
Schnittstellen	8 x USB 2.0 gesamt
	1 DVI-I (integrierter VGA kann mit DVI-VGA-Adapter benutzt werden) x DVI
	1 x DisplayPort
Stromversorgung	100 V - 240 V (AC-Eingang) , 50 - 60 Hz
Lieferumfang	Kein Adapter für VGA/DVI/DP
Datenblatt	http://docs.ts.fujitsu.com/dl.aspx?id=2d0c5f97-04c4-400b-9222-cb0704b3cb9b
Herstellernummer	VFY:S0700PP1R2DE

Zusätzliche Netzwerkkarten gibt es zu kaufen. Für intern oder als USB Interface, jedoch liegen hier keine Erfahrungen vor. **Die Anleitung arbeitet mit der vorhandenen Karte und VLANs.**

Einsatz im Freifunk-Bereich

Als **OFFLOADER** für VPN-Verbindungen (fastd) übers Internet zu den Gateways der Freifunk-MWU Community.

MaxSpeed: 50 bis 90 GByte

Konstellation 1: Futro + TP WD 841N = 55 Mbit – VLAN

Konstellation 2: Futro + TP WDR 3600 = 90 Mbit – VLAN

Konstellation 3: Futro mit zweiter Netzwerkkarte ohne VLAN : -nicht bekannt-

Ziel: Freifunk-Image aufspielen

Möglichkeit : OS von USB Stick booten. Image mit „dd“ auf SSD aufspielen.

Benötigt wird:

1. USB Stick mind. 1 GB
2. OS zum Aufspielen auf den Stick
3. Stick bootbar machen

Zu **(1)** – Hast du oder kaufen

Zu **(2)** – Es bietet sich: **KNOPPIX** an. Link: <http://www.knoppix.org/>. **CD-Version** reicht aus, besonders, wenn Stick weniger als 8 GB Speicher. Nachteil: Man muss Knoppix erst mal booten, bevor man sich einen USB Stick erzeugen kann.

Alternative: **Systemrescuecd** → Link: <http://www.system-rescue-cd.org> Hier kann man auch einen USB-Stick bootbar machen. → Link: <http://www.system-rescue-cd.org/Installing-SystemRescueCd-on-a-USB-stick> Methoden A + B für Linux Umgebungen, C für Windows: Ohne das System booten zu müssen kann direkt ein Stick erzeugt werden (von mir nicht getestet).

Zu **(3)** – Stick wird unter gebootetem KNOPPIX eingerichtet (s. Anleitung unten)

KNOPPIX Download von Mirrors

Folgende Institutionen, die meisten davon (für Studenten und Schüler besonders praktisch) sind gut über das deutsche F einen Download-Bereich für bootbare KNOPPIX GNU/Linux ISO-Images zur Verfügung zu stellen:

Download von	Protokoll	Si
ftp.uni-kl.de	[rsync CD] [rsync DVD] [ftp CD] ftp DVD] [http CD] [http DVD]	
ftp.knoppix.nl	[rsync CD] [rsync DVD] [ftp Mirror UNI-KL	
ftp.gwdg.de	[ftp Download Site Sponsor: Universität	G
sunsite.informatik.rwth-aachen.de	[ftp Kaiserslautern] [http]	

1 - Hier gibts Knoppix in allen Varianten

Nach dem Download liegt dann folgende Datei auf der Platte:

ADRIANE-KNOPPIX_V7.2.0gCD-2013-07-28-DE.iso

ToDo: Einen Rechner unter Knoppix starten

...um USB-Stick mit Knoppix zu erstellen

Zum Erzeugen eines USB-Sticks unter Knoppix benötigt man zunächst ein von der ISO gebootetes System. Da kann man so oder so machen.

- ISO auf CD brennen und davon booten, oder ...
- VIRTUALBOX installieren, einen leeren Rechner anlegen und ISO als CD einbinden. Von CD virtuell booten und dann mit dieser Anleitung weitermachen
 - <https://www.linux.de/anleitungen/36-knoppix-7-auf-usb-stick-installieren>

Futro mit Knoppix vom Stick starten

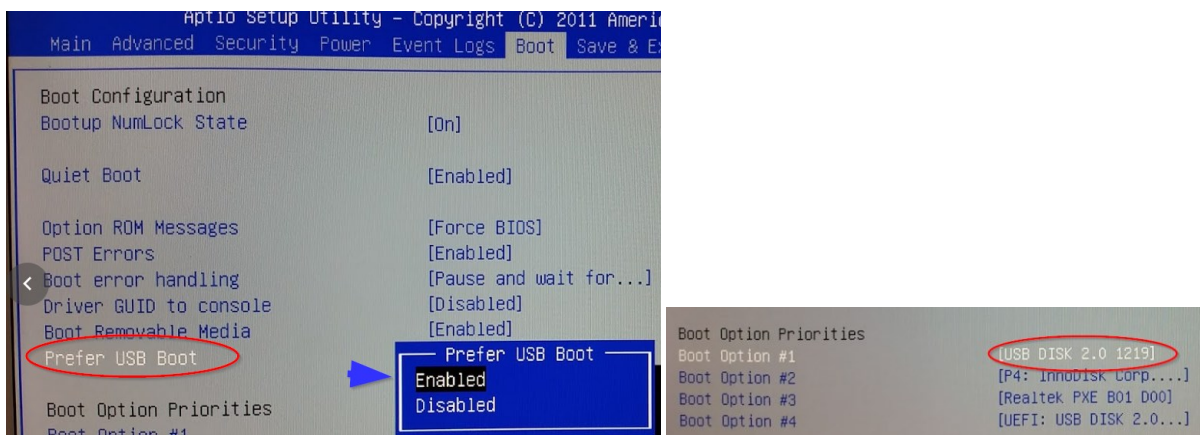
- Futro mit Keyboard, Maus und Screen versorgen (wäre echt gut)
- Ins BIOS: Beim Einschalten wie wild F12 drücken (siehe Abbildung Nr.2 & 3)
- Im BIOS nach dem BOOT-Optionen suchen und USB Anschlüsse bootbar machen (genauer weiß ichs nicht mehr)
- Futro AUSSCHALTEN (Power-Button vorne laaaaaange drücken)
- Stick einstecken (gerne auch vorne) und Futro neu starten.
- KNOPPIX sollte geladen werden.** WARTEN bis Grafikgedöns sich ausgezappelt hat.



Nr.2 // Menü nach viel F12 Gedrücke

Nr.3 // BIOS aufrufen und dauerhaft umstellen

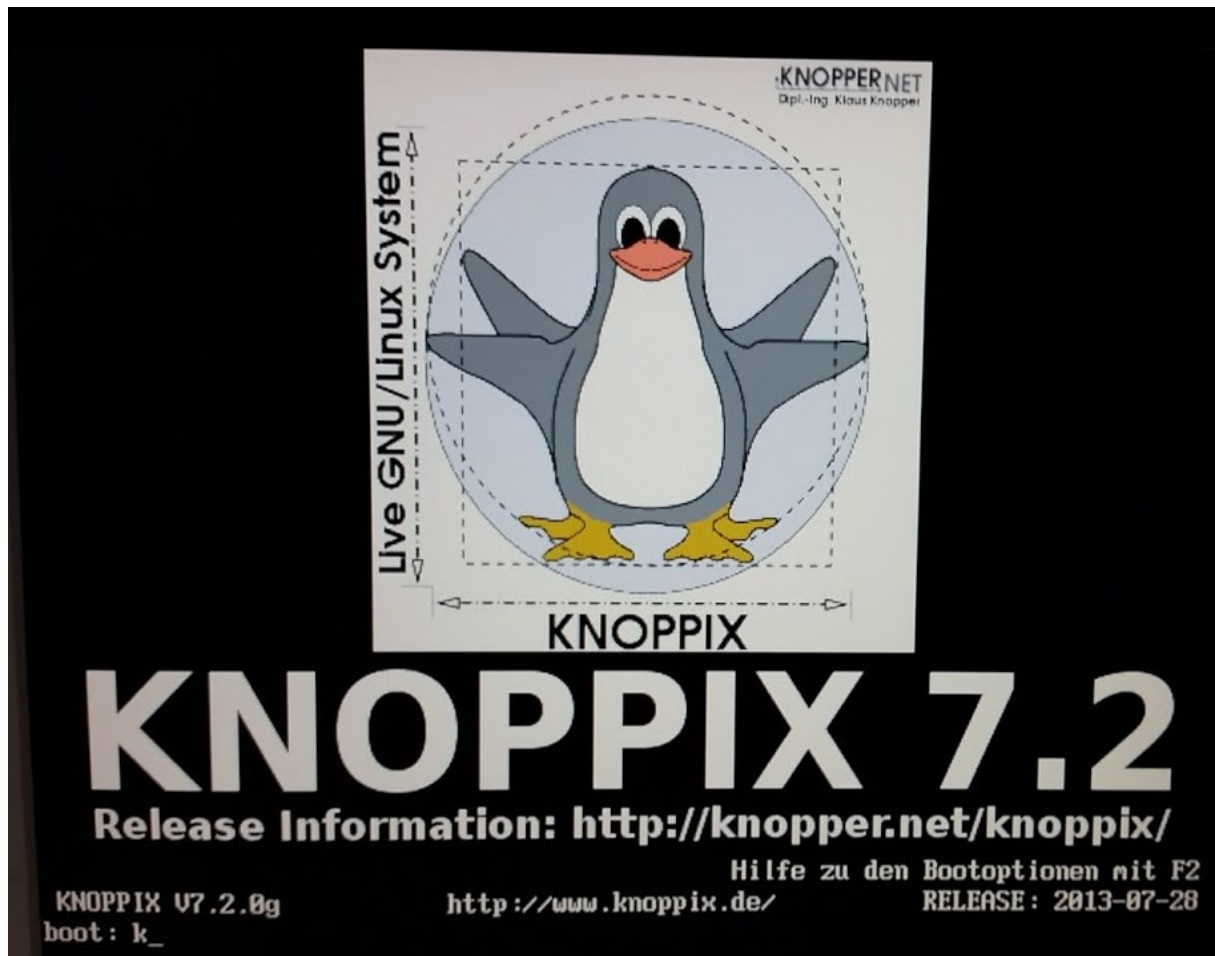
Damit dauerhaft von USB Stick zuerst gebootet wird, gibt es zwei Einstellungen im BIOS zu machen. Folge „Enter Setup“ und stelle unter „Boot“ um auf „Prefer USB Boot“ = Enabled.



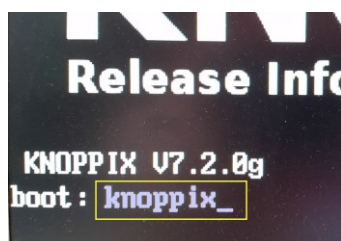
In den Optionen darunter „Boot Option Priorities“ stellst du ein, dass der USB Stick immer zuerst abgefragt wird. Der Rest darunter spielt von der Reihenfolge her keine Rolle. → Save & Exit

Er bootet jetzt wieder und sollte direkt unser Knoppix vom Stick hochfahren.

Wir werden begrüßt mit:



Kaum zu erkennen: unten will er eine Eingabe haben. Hier geben wir schnell „knoppix“ ein bevor ein anderer Modus startet (Drücke F2 wenn du mehr wissen willst).



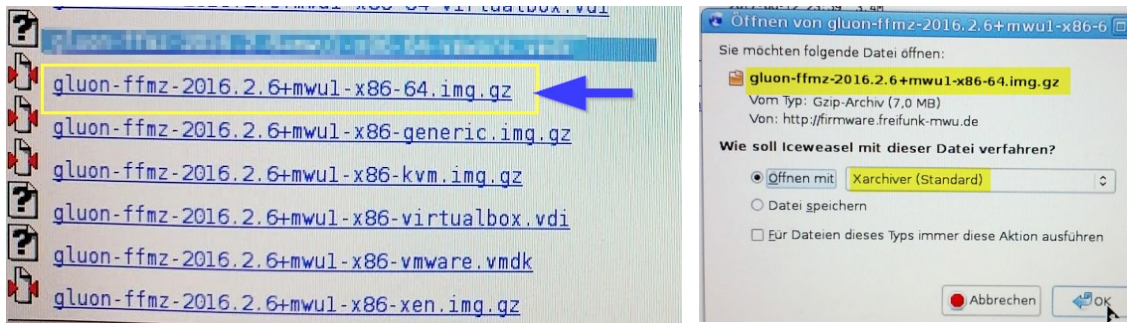
Das richtige Freifunk-Futro Image

Davon gibt's mehrere in unserer Sammlung auf: firmware.freifunk-mwu.de

Bitte wähle die richtige **Region** {Mainz|Wiesbaden|...} aus. Hier wird weiter exemplarisch „Mainz“ benutzt: <http://firmware.freifunk-mwu.de/mainz/stable/factory/>

Bis ich hier etwas anderes behaupte, empfehle ich das Image hier zu benutzen:

gluon-ffmz-yyyy.xx.yy+mwu1-x86-64.img.gz



Das ist ein ZIP-Archiv und die darin enthaltene *.IMG Datei **muss ausgepackt werden!**

Ich empfehle dir nach /tmp zu speichern.

```
knoppix@Microknoppix:/tmp$ ls -alh
insgesamt 60M
drwxrwxrwt 4 root root 240 Jun 30 10:08 .
drwxr-xr-x 18 root root 0 Jun 30 10:05 ..
-rw-r--r-- 1 knoppix knoppix 53M Jun 30 10:07 gluon-ffmz-2016.2.6+mwu1-x86-64.img
-rw-r--r-- 1 knoppix knoppix 7,1M Jun 30 10:07 gluon-ffmz-2016.2.6+mwu1-x86-64.img.gz
srwxr-xr-x 1 knoppix knoppix 0 Jun 30 10:06 .lterminal-socket:0-knoppix
srwxr-xr-x 1 knoppix knoppix 0 Jun 30 10:06 .menu-cached:0-knoppix
srwxr-xr-x 1 knoppix knoppix 0 Jun 30 10:06 .pcmanfm-socket--0-knoppix
-rw-r--r-- 1 knoppix knoppix 57 Jun 30 10:06 serverauth.Vzv37VmdzJ
drwxr-xr-x 2 knoppix knoppix 60 Jun 30 10:06 ssh_3ghZNoX6t81x
```

Wie bekomme ich die Datei auf den Futro?

Wieder gibt es zwei Möglichkeiten (ich empfehle (b) wenn ansatzweise möglich):

- Kopiere die *.IMG auf einen weiteren Stick oder ...
 - .. stecke in den Futro ein LAN Kabel zu deinem Hausnetz und lade mit dem Browser direkt aus dem [Netz](#) ins /tmp Verzeichnis
- Entpacke das IMG mit dem Datei-Explorer von KNOPPIX. Merke dir wo das File liegt
- Jetzt unter Knoppix eine Kommandozeilen-Box öffnen und den Befehl ausführen:

```
dd if=/pfad/zur/datei.img of=/dev/sda
```

Das Device "/dev/sda" ist die SSD-Festplatte in unserer Ausstattung.

```
knoppix@Microknoppix:/tmp$ dd if=/tmp/gluon-ffmz-2016.2.6+mwu1-x86-64.img of=/dev/sda
107520+0 Datensätze ein
107520+0 Datensätze aus
55050240 Bytes (55 MB) kopiert, 5,0202 s, 11,0 MB/s
knoppix@Microknoppix:/tmp$
```

- Das Image wird nun auf die Platte übertragen. Dauert ca. 20-40 Sekunden
- Um sicher zu gehen, dass alle Schreiboperationen auf der CF-Karte abgeschlossen sind, führen wir noch einmal den Befehl `sync` aus und warten bis wir wieder auf der Kommandozeile laden.

```
knoppix@Microknoppix:/tmp$ sync
```

- Dann Rechner runterfahren – Stick entfernen – Neu starten

Freifunk-Image rennt

Es sollte jetzt das Freifunk-Image booten. Die Zeilen rauschen am Bildschirm durch.

Wenn durchgerauscht, einfach RETURN drücken – Der von einer Anmeldung her bekannte Begrüßungsbildschirm erscheint und der Command-Prompt wartet auf Eingaben.

ACHTUNG:

Auch wenn wir die Konsole sehen, **ist der Futro mit dem FF-Image jetzt im Konfig-(Wizard)-Modus**. Das ist ein Web-Interface, dass man in einem Browser per IP-Adresse aufrufen muss. Der Futro bietet auf seiner Netzwerkschnittstelle jetzt DHCP an mit 192.168.1.xxx. Wenn das LAN Kabel vom Hausnetz jetzt noch im Futro steckt gibt's mehr als einen DHCP Server. Es wäre jetzt Zufall von wem dein Rechner jetzt eine IP-Adresse bekommt. Verbinde deswegen besser den Futro jetzt **nur mit einem Rechner per LAN Kabel**. Der Rechner sollte per DHCP eine Adresse vom Futro bekommen und im Browser kannst du die bekannte 192.168.1.1 benutzen.

Freifunk-Konfiguration

Bedingung um hier weiterzumachen: Das WebInterface bietet die Startseite an:

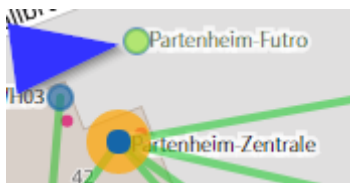
- Erst in den **Erweiterte Einstellungen** (rechts oben wechseln) **ein Passwort** setzen für den SSH Zugang später – Speichern!
- Wähle auf der ersten Seite einen **Namen** und **aktiviere VPN**. Das übliche weiter unten: Bandbreite reduzieren wir besser nicht, da wir mit dem Futro ja HighSpeed machen wollen. Die **Koordinaten** geben wir ein, wenn er auf der Karte zu sehen sein soll. Er kann aber unsichtbar bleiben, da die Anzahl der Nutzer zum Router angezeigt wird auf der map.
- Wir **Aktivieren** den Futro und geben den öffentlichen VPN-Schlüssel per E-Mail weiter ans KEY-Admin-Team zum Eintragen.

Warten ist angesagt. Ohne eingetragenen Key geht's jetzt nicht weiter.

Futro geht Online

Zeit sinnvoll nutzen: Hausnetz-Kabel in den Futro stecken. Ggf. neu durchbooten und warten bis er auf der Karte online geht oder in der Knotenliste als neuer Knoten mit deinem vergebenen Namen begrüßt wird. Dann klick auf den Namen um die Details zum Knoten zu sehen.

Er ist Online





- Wir sehen die **interne IPV6 Adresse** (Blauer Kasten im Bild oben).
- Mit dieser Adresse und dem Kommando **SSH, oder** unter Windows mit dem Tool **Putty**, verbinden wir uns mit dem Futro. Das ist nicht nur bequemer als direkt an der Futro Konsole (kann nur engl. Tastatur), sondern auch ein Test ob das SSH-Passwort funktioniert.

Einrichtung der Futro-Konfig

Möglichkeit 1: Futro: alles über eine Netzwerkkarte (VLANs)

Möglichkeit 2: Wird hier nicht näher behandelt, weil:

Wir mit zwei Netzwerkkarten in einem Futro oder anderem x86-PC **keine VLANs benötigen**.

Hier haben wir dann nämlich *eth0* und *eth1* für jede Karte und können in der Gluon-Oberfläche dann jeweils sagen *mesh-vpn* mit einem DSL Anschluss (*eth0*) und aktivieren „mesh on lan“ im Experten-Modus unter Netzwerk. Dann brauchen wir im Router später nur „mesh on lan“ ebenfalls aktivieren und die beiden Schnittstellen mit einem LAN Kabel verbinden. Das wars dann schon.

Eine weitere Netzwerkkarte nachträglich kostet halt Geld. Ein größerer x86-Rechner verbraucht für den Dauerbetrieb halt auch mehr Strom. Ein Futro benötigt zum Vergleich soviel Leistung wie 3 Router – also gerade noch akzeptabel, oder?

Der Futro hat nur eine Netzwerkkarte wie bekannt ist. Es tut aber not, dass wir mindestens 2 logische Verbindungen benötigen - eine WAN und ein bis zwei LAN Verbindungen zu unterschiedlichen Zielen. Geht nicht? Doch – mit VLAN.

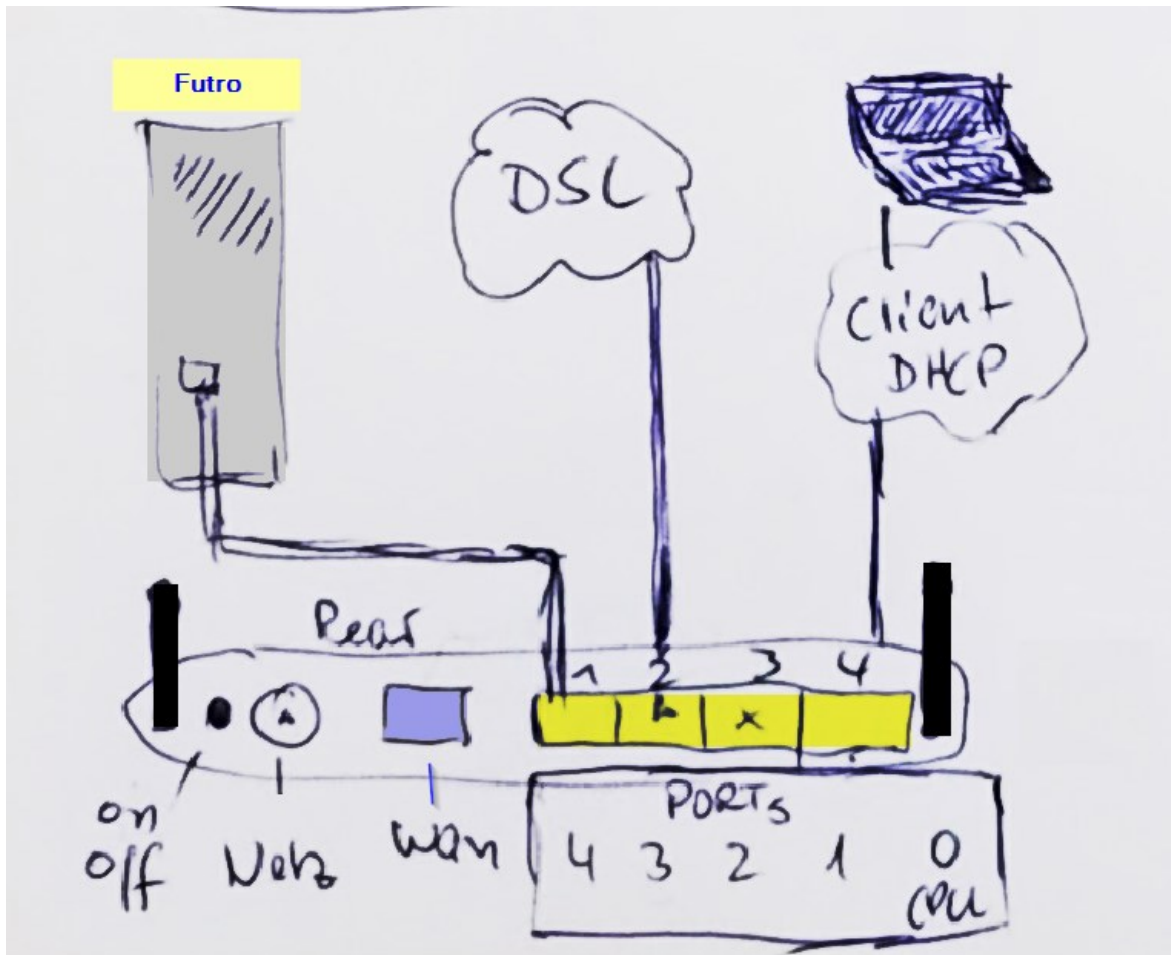
VLAN = Virtuelles LAN, ermöglicht durch zusätzliche Markierungen der Netzwerkdaten, dass man alles über ein Kabel schicken kann und die am anderen Ende wieder auseinanderfieseln kann. Wir markieren „taggen“ die Daten als VLAN mit den Nummern 3, 4, 5 und schicken sie allesamt über den einen Anschluss – aber an wen?

Wir wollen die Daten in verschiedene Richtungen weiter delegieren. Dazu brauchen wir ein Gerät, dass die „getaggten“ markierten Pakete wieder trennen kann. Ein normaler Home-Switch (Verteiler) kann das nicht, aber jeder Switch in einem Router (die 4 gelben Buchsen) kann das, wenn man ihm in

der Netzwerk-Config sagt was er tun soll. Preisgünstig ist das auch noch. Ein richtiger „managed“-Switch kostet da schon schnell mehr als 60 Euro

Ich hatte hier exemplarisch einen TP-Link WDR 3500 benutzt. Die Anleitung gilt aber auch für andere Modelle, wenn man die Bezeichnung der internen Portnummern austauscht. Dazu später mehr.

Erst schauen wir uns mal die Verkabelung an:



(Bild mit allen Geräten, Router von Rückseite)

Das fiese bei vielen Routern: Die (gelben) 4 LAN Ausgänge sind am Gehäuse mit 1 bis 4 nummeriert. Intern werden sie angesprochen durch andere Ziffern, umgekehrte Werte 4 bis 1 (Buchse Nr. 1 = Port Nr. 4 in der Config). Das ist bei jedem Router anders. Siehe immer vorher im **wiki von OpenWRT** nach deinem Modell, wenn du nicht sicher bist.

https://openwrt.org/toh/start?dataflt%5BBrand*~%5D=TP

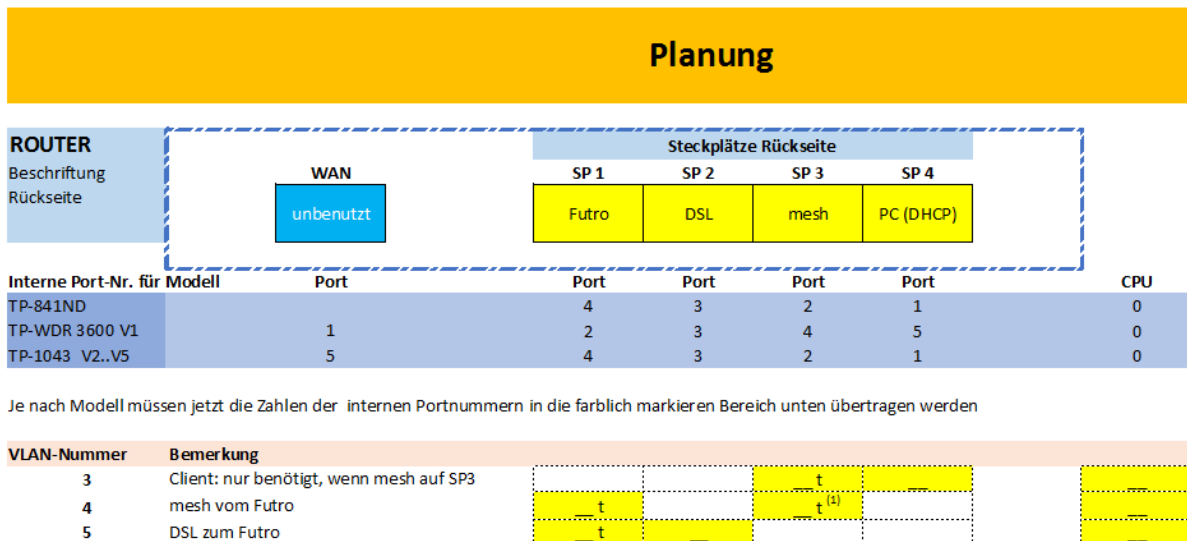
Ein paar Worte noch zum Bild oben

Der Futro wird an Steckplatz Nr. 1 angeschlossen. Hier kommen alle Daten „getaggt“ mit den VLAN Nummern 3,4,5 an und müssen nun zu den anderen Anschlüssen verteilt werden. Pakete mit VLAN5 werden auf den im Steckplatz Nr. 2 verkabelten DSL Anschluss (Dein Heimnetz) weitergegeben. Die VLAN-Markierung wird entfernt, damit es wieder normale Netzwerkpakete werden.

Auf Steckplatz Nr. 4 kann ein PC eingesteckt werden. Hier kommen die CLIENT-Pakete heraus. Das Endgerät hier bekommt eine Freifunk-IP-Adresse deiner Region. Auch hier werden die VLAN-Markierungen entfernt.

Der Steckplatz Nr. 3 ist für besondere Nutzung vorgesehen.

- a) **UBI für Backbone:** Dann werden die VLANs 3 und 4 benötigt und getaggt dort ausgeleitet.
- b) **Mesh-on-LAN:** Möchte man jedoch mit einem anderen Router per Kabel meshen (Option Mesh-on-LAN) wird nicht getaggt und nur das VLAN 4 ausgeleitet.



(1) Hängt am SP3 eine UBI dran, dann Port taggen, wird nur mesh_on_lan mit weiterem Router gemacht, dann KEIN "t" anhängen

Die Tabelle „Planung“ oben soll euch als Konfigurationshilfe dienen.

Wir sehen ganz oben die Beschriftung der Rückseite des Routers. In der gestrichelten Box zeigen die gelben Buchsen an, was wir da drauf stecken. Hier sind SP3 und SP4 als optional anzusehen und für dein einfachen „Offloader“-Betrieb nicht notwendig. Es schadet aber nicht, wenn wir die gleich mit konfigurieren.

Darunter sind die **internen Portnummern** ausgewählter Router bereits eingesetzt worden. Die Info kam wieder durch eine Abfrage beim Wiki von OpenWRT zum jeweiligen Modell.

Im unteren Teil braucht ihr jetzt nur noch die internen Portnummern übertragen in die gelben Felder (CPU ganz rechts nicht vergessen). Dieser ausgefüllte Block ist dann die Orientierungshilfe für das Anpassen der *network*-Datei im Futro und im Router.

(t)=getaggt/markiert = VLAN Nummer wird ins Datenpaket übernommen, sonst ungetaggt/unmarkiert = keine VLAN Nummer, also normales Datenpaket.

Zur Orientierung:

Von einem Gerät, dass kein VLAN erwartet/konfiguriert hat, werden VLAN Pakete ignoriert. Das gilt auch für den umgekehrten Fall. Nur wenn beide VLAN Informationen übereinstimmen wird das Paket weiterverarbeitet.

Also nun ran an die Kartoffeln. Wir sind per SSH verbunden oder sind live an der Konsole des Futros.
Hier geht es jetzt weiter...

Futro: Wir editieren die NETWORK Datei

Wir haben auf den Freifunk-Images nur den einfachen VIM Editor. Die Bedienung ist jetzt nicht Gegenstand dieser Anleitung. Das dürft ihr euch gerne hier anlesen:

<http://www.inf.fu-berlin.de/lehre/WS98/brueckenkurs/skript/node9.html>

Wir öffnen die network Datei im Editor:

```
vi /etc/config/network
```

Ganz vorsichtige Leute machen vorher eine Kopie der Datei mit

```
cp /etc/config/network /etc/config/network.org
```

Zum Verständnis: Ich erwähne jetzt nur die wichtigen Zeilen, die hinein müssen oder so angepasst werden sollten. Da steht noch viel mehr in den jeweiligen Blöcken drin, was auch so bleibt und von Version zu Version der FF-Images auch immer mal schwanken kann.

```
config interface 'wan'
    option ifname 'eth0.5'
    option auto '1'

config interface 'mesh_wan'
    option auto '0'

# den meshvlan4 Block neu anlegen
config interface 'meshvlan4'
    option auto '1'
    option ifname 'eth0.4'
    option proto 'gluon_mesh'
    option fixed_mtu '1'
    option macaddr '...übernehmen und eindeutig machen...'

config interface 'client'
    option type 'bridge'
    option macaddr '...beibehalten was hier steht...'
    list ifname 'bat0'
    list ifname 'eth0.3'
```

Erläuterungen:

option auto '0' oder **'1'** bedeuten, dass dieses Interface gerade 0=aus, 1=aktiv sein soll.

option ifname 'eth0.x' x = Nr. des VLANs mit dem die Pakete markiert werden, wenn sie über das Interface laufen. Umgekehrt werden hier auch nur Pakete angenommen, die sich mit der eigenen VLAN-Nr. ausweisen. Der Rest wird ignoriert.

[!] Die gesamte *network*-Datei findet ihr auch im [Appendix – 1 - Konfig vom Futro](#)

Datei nun Speichern (!) und Editor verlassen: <ESC> :wq <Enter> → **NOCH NICHT BOOTEN !!!**

Router: Wir editieren die NETWORK Datei

Die Änderungen die jetzt folgen sind bis auf die Port-Nummern auf fast jedes TP-Link-Modell anwendbar. Zumindest vom Sinn her, aber Mitdenken und -lesen sind unverzichtbar...

Annahme:

Der verwendete Router hat bereits ein aktuelles Freifunk-Image drauf. Es wird kein VPN über DSL benötigt, somit „mesht“ der Router erst mal nur per Antenne. MeshOnLAN und MeshOnWAN sind ausgeschaltet. Es sind keine Netzwerk-Kabel eingesteckt.

Erst Verkabelung ändern:

- Futro LAN Buchse verbinden mit Router Steckplatz Nr. 1 (siehe Bild vorher).
- Router Steckplatz Nr. 2 verbinden wir mit dem Hausnetz (Internet/DSL)
- Router einschalten
- Jetzt **Futro rebooten** (von der Console aus mit „reboot -f“ <Enter>) oder am Gerät einmal kurz auf den Knopf vorne drücken. Warten – Gerät fährt runter. Wenn aus, nochmal drücken für Start.

Mit Router per SSH verbinden:

- Deinen PC per Kabel mit Steckplatz Nr. 4 verbinden.
- Eine IpAdresse bekommt der Rechner nur, wenn der Router mit dem FF-Netz mesht, also weitere Router in Antennen-Reichweite sind. Sonst musst du dir eine feste IP-Adresse am PC temporär setzen. z.B: {Mainz} 10.37.5.123 {Wiesbaden} 10.56.5.123/ Maske 255.255.255.0.
- Testweise einen PING auf 10.37.0.1(MZ) oder 10.56.0.1(WI) machen. Das ist dein Freifunk-Router mit dem du per Kabel verbunden bist (Besser WiFi abschalten, fall PC das auch kann). Geht das, kannst du weitermachen, sonst musst du deine Einstellungen prüfen
- SSH / Putty benutzen und mit 10.37.0.1(MZ) oder 10.56.0.1(WI) verbinden. Benutzer = root, Passwort = deines, was du dir beim originären Einspielen des Freifunk-Images mal gesetzt hattest. Ups, kein Passwort? Dann 10 Sekunden Reset-Button gedrückt halten bis Gerät neu bootet. Dann bist du wieder im Gluon-Konfig-Modus und kannst dir eines setzen.
- Verbindung da, dann ...

Wir öffnen die network Datei im Editor:

```
vi /etc/config/network
```

Ganz vorsichtige Leute machen vorher wieder eine Kopie der Datei mit

```
cp /etc/config/network /etc/config/network.org
```

- weiter auf der nächsten Seite -

Zum Verständnis: Ich erwähne jetzt wieder nur die wichtigen Zeilen, die hineinmüssen oder so angepasst werden sollten. Da steht noch viel mehr in den jeweiligen Blöcken drin, was auch so bleibt und von Version zu Version der FF-Images auch immer mal schwanken kann.

Die Portnummern stammen aus der Konfighilfe weiter oben vom Modell *TP-Link 1043 ND Version 5*

Zur Erinnerung:

Interne Port-Nr. für						
Modell	Port	Port	Port	Port	Port	CPU
TP-1043 V2..V5	5	4	3	2	1	0

```

config switch_vlan 'vlan1'
    option vlan '1'
# Hier entfernen wir die Ports 1 bis 4
    option ports '0t'

# WAN Buchse: unbenutzt - trotzdem nicht entfernen
config switch_vlan 'vlan2'
    option vlan '2'
    option ports '0t 5'

config switch_vlan 'vlan3'
    option vlan '3'
# Optional für Zugang zu Geräten auf Port 1+2
    option ports '0t 1 2t'

config switch_vlan 'vlan4'
    option vlan '4'
# mesh vom Futro kommt auf Port 4 raus
    option ports '0t 4t'

config switch_vlan 'vlan5'
    option vlan '5'
# DSL an Port 3 zum Futro an Port 4 brücken
    option ports '0t 3 4t'

config interface 'wan'
    option ifname 'eth0.2'
    option auto '1'

config interface 'mesh_wan'
    option auto '0'

config interface 'mesh_vlan4'
    option auto '1'
    option ifname 'eth0.4'

config interface 'client'
    list ifname 'bat0'
    list ifname 'eth0.3'

config interface 'mesh_lan'
    option auto '0'

```

Erläuterungen:

MAC-Adressen, die in den Blöcken aufgeführt werden, deren Blocknamen mit „mesh..“ beginnen, **dürfen nicht doppelt** vorkommen. Verändert diese an einer Stelle in der Zahlenkolonne.

[!] Eine network-Datei findet ihr auch im [Appendix – 2 - Konfig vom Router \(Portnummern für TP-Link 841N\)](#). Bitte nicht blind benutzen, da die MacAdressen aus eurem Original bestand haben sollten.

Datei nun Speichern (!) und Editor verlassen: <ESC> :wq <Enter>

Bootet den Router nun neu: „reboot -f“ <Enter>

Ändert eure Netzwerk-Karte im PC wieder auf DHCP zurück (automatisch beziehen) und wartet sehnsüchtig in einer Kommando-Konsole (Dos-Box unter Windows) mit dem Kommando „ipconfig“ bis ihr eine Freifunk-Adresse bekommt. Ihr Steckt mit dem Kabel weiterhin auf Steckplatz Nr. 4

FERTIG. Beobachtet den Knoten auf der map und probiert auch mal ein Login per WLAN aus.



Anlagen: Futro S700 – Datenblatt [FutroS700-Datenblatt.pdf](#)

Appendix – 1 - Konfig vom Futro

Habt bitte ein wachsames Auge auf die Zeilen. Durch neue Gluon-Versionen werden auch hin und wieder weitere OPTIONEN eingefügt. Das hier ist ein Muster vom Stand Anfang 2018. Es kann Abweichungen geben!!!

```
config interface 'loopback'
    option ifname 'lo'
    option proto 'static'
    option ipaddr '127.0.0.1'
    option netmask '255.0.0.0'

config interface 'wan6'
    option proto 'dhcpv6'
    option sourcefilter '0'
    option ifname 'br-wan'
    option ip6table '1'
    option peerdns '0'

config globals 'globals'
    option ula_prefix 'XXXX:2186:e36b::/48'

config interface 'wan'
    option ifname 'eth0.5'
    option multicast_querier '0'
    option peerdns '0'
    option auto '1'
    option type 'bridge'
    option proto 'dhcp'
    option macaddr 'XX:XX:XX:XX:XX:XX'

config rule6 'wan6_lookup'
    option mark '0x01/0x01'
    option lookup '1'

config route6 'wan6_unreachable'
    option type 'unreachable'
    option table '1'
```

```
    option target '::/0'
    option metric '65535'
    option gateway '::'
    option interface 'loopback'

config interface 'mesh_wan'
    option ifname 'br-wan'
    option auto '0'
    option transitive '1'
    option fixed_mtu '1'
    option proto 'gluon_mesh'

config interface 'meshvlan4'
    option auto '1'
    option ifname 'eth0.4'
    option mesh 'bat0'
    option proto 'batadv'
    option macaddr 'XX:XX:XX:XX:XX:XX'

config interface 'client'
    option type 'bridge'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    list ifname 'bat0'
    list ifname 'eth0.3'
    option proto 'dhcpv6'
    option reqprefix 'no'
    option robustness '3'
    option query_interval '2000'
    option query_response_interval '500'
    option peerdns '1'
    option sourcefilter '0'
```

```

config interface 'bat0'
    option multicast_router '2'
    option ifname 'bat0'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option learning '0'
    option proto 'none'

config interface 'mesh_vpn'
    option ifname 'mesh-vpn'
    option transitive '1'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option fixed_mtu '1'
    option proto 'gluon_mesh'

config device 'local_node_dev'
    option macaddr 'XX:XX:XX:XX:XX:XX'

```

```

    option ifname 'br-client'
    option name 'local-node'
    option type 'macvlan'

config interface 'local_node'
    option ifname 'local-node'
    option ipaddr '10.XX.0.1'
    option ip6addr 'XXXX:b4dc:4b1e::1/128'
    option netmask '255.255.192.0'
    option proto 'static'

config route6 'local_node_route6'
    option target 'XXXX:b4dc:4b1e::/64'
    option gateway ':::'
    option interface 'client'

```

Vorsicht mit MACADRESSEN und standortbezogenen IP-Adressen (MZ/WI/ u.a.)

Appendix – 2 - Konfig vom Router (Portnummern für TP-Link 841N)

```

config interface 'loopback'
    option ifname 'lo'
    option proto 'static'
    option ipaddr '127.0.0.1'
    option netmask '255.0.0.0'

config globals 'globals'
    option ula_prefix 'XXXX:ff3f:3934::/48'

config interface 'wan6'
    option proto 'dhcpv6'
    option sourcefilter '0'
    option ifname 'br-wan'
    option ip6table '1'
    option peerdns '0'

config switch
    option name 'switch0'
    option reset '1'
    option enable_vlan '1'

config switch_vlan 'vlan1'
    option device 'switch0'
    option vlan '1'
    option ports '0t '

config switch_vlan 'vlan4'
    option device 'switch0'
    option vlan '4'
    option ports '0t 2t 4t'

config switch_vlan 'vlan3'
    option device 'switch0'
    option vlan '3'
    option ports '0t 1 2t'

config switch_vlan 'vlan5'
    option device 'switch0'
    option vlan '5'
    option ports '0t 3 4t'

config interface 'wan'
    option ifname 'eth1'
    option multicast_querier '0'
    option peerdns '0'
    option auto '1'
    option type 'bridge'
    option proto 'dhcp'
    option macaddr 'XX:XX:XX:XX:XX:XX'

```

```

config rule6 'wan6_lookup'
    option mark '0x01/0x01'
    option lookup '1'

config route6 'wan6_unreachable'
    option type 'unreachable'
    option table '1'
    option target ':::/0'
    option metric '65535'
    option gateway ':::'
    option interface 'loopback'

config interface 'mesh_wan'
    option ifname 'br-wan'
    option auto '0'
    option transitive '1'
    option fixed_mtu '1'
    option proto 'gluon_mesh'

config interface 'mesh_vlan4'
    option igmp_snooping '0'
    option ifname 'eth0.4'
    option transitive '1'
    option fixed_mtu '1'
    option proto 'gluon_mesh'
    option type 'bridge'
    option auto '1'
    option macaddr 'XX:XX:XX:XX:XX:XX'

config interface 'client'
    option type 'bridge'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option proto 'dhcpv6'
    option reqprefix 'no'
    option robustness '3'
    option query_interval '2000'
    option query_response_interval '500'
    option peerdns '1'
    option sourcefilter '0'
    list ifname 'eth0.3'
    list ifname 'bat0'

config interface 'mesh_vpn'
    option ifname 'mesh-vpn'
    option transitive '1'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option fixed_mtu '1'
    option proto 'gluon_mesh'

```

```

config interface 'mesh_lan'
    option igmp_snooping '0'
    option ifname 'eth0'
    option transitive '1'
    option fixed_mtu '1'
    option proto 'gluon_mesh'
    option type 'bridge'
    option auto '0'
    option macaddr 'XX:XX:XX:XX:XX:XX'

config interface 'mesh_radio0'
    option proto 'gluon_mesh'

config interface 'bat0'
    option multicast_router '2'
    option ifname 'bat0'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option learning '1'
    option proto 'none'

```

```

config device 'local_node_dev'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option ifname 'br-client'
    option name 'local-node'
    option type 'macvlan'

config interface 'local_node'
    option ifname 'local-node'
    option ipaddr '10.XX.0.1'
    option ip6addr 'XXXX:b4dc:4b1e::1/128'
    option netmask '255.255.192.0'
    option proto 'static'

config route6 'local_node_route6'
    option target 'XXXX:b4dc:4b1e::/64'
    option gateway '::'
    option interface 'client'

```

Vorsicht mit MACADRESSEN und standortbezogenen IP-Adressen (MZ/WI/ u.a.)

Bitte umblättern

Appendix – 4 - Konfig vom Router (Portnummern für TL-WDR3600)

```
config interface 'loopback'
    option ifname 'lo'
    option proto 'static'
    option ipaddr '127.0.0.1'
    option netmask '255.0.0.0'

config globals 'globals'
    option ula_prefix 'XXXX:ff3f:3934::/48'

config interface 'wan6'
    option proto 'dhcpv6'
    option sourcefilter '0'
    option ifname 'br-wan'
    option ip6table '1'
    option peerdns '0'

config switch
    option name 'switch0'
    option reset '1'
    option enable_vlan '1'

config switch_vlan 'vlan1'
    option device 'switch0'
    option vlan '1'
    option ports '0t '

config switch_vlan 'vlan4'
    option device 'switch0'
    option vlan '4'
    option ports '0t 2t 4t'

config switch_vlan 'vlan3'
    option device 'switch0'
    option vlan '3'
    option ports '0t 4t 5'

config switch_vlan 'vlan5'
    option device 'switch0'
    option vlan '2'
    option ports '0t 2t 3'

config switch_vlan 'vlan2'
    option device 'switch0'
    option vlan '2'
    option ports '0t 1'

config interface 'wan'
    option ifname 'eth0.5'
    option multicast_querier '0'
    option peerdns '0'
    option auto '1'
    option type 'bridge'
    option proto 'dhcp'
    option macaddr 'XX:XX:XX:XX:XX:XX'

config rule6 'wan6_lookup'
    option mark '0x01/0x01'
    option lookup '1'

config route6 'wan6_unreachable'
    option type 'unreachable'
    option table '1'
    option target '::/0'
    option metric '65535'
    option gateway '::'
    option interface 'loopback'

config interface 'mesh_wan'
    option ifname 'br-wan'
    option auto '0'
    option transitive '1'
    option fixed_mtu '1'
```

```
    option proto 'gluon_mesh'

config interface 'mesh_vlan4'
    option igmp_snooping '0'
    option ifname 'eth0.4'
    option transitive '1'
    option fixed_mtu '1'
    option proto 'gluon_mesh'
    option type 'bridge'
    option auto '1'
    option macaddr 'XX:XX:XX:XX:XX:XX'

config interface 'client'
    option type 'bridge'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option proto 'dhcpv6'
    option reqprefix 'no'
    option robustness '3'
    option query_interval '2000'
    option query_response_interval '500'
    option peerdns '1'
    option sourcefilter '0'
    list ifname 'eth0.3'
    list ifname 'bat0'

config interface 'mesh_vpn'
    option ifname 'mesh-vpn'
    option transitive '1'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option fixed_mtu '1'
    option proto 'gluon_mesh'

config interface 'mesh_lan'
    option igmp_snooping '0'
    option ifname 'eth0'
    option transitive '1'
    option fixed_mtu '1'
    option proto 'gluon_mesh'
    option type 'bridge'
    option auto '0'
    option macaddr 'XX:XX:XX:XX:XX:XX'

config interface 'mesh_radio0'
    option proto 'gluon_mesh'

config interface 'bat0'
    option multicast_router '2'
    option ifname 'bat0'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option learning '1'
    option proto 'none'

config device 'local_node_dev'
    option macaddr 'XX:XX:XX:XX:XX:XX'
    option ifname 'br-client'
    option name 'local-node'
    option type 'macvlan'

config interface 'local_node'
    option ifname 'local-node'
    option ipaddr '10.XX.0.1'
    option ip6addr 'XXXX:b4dc:4b1e::1/128'
    option netmask '255.255.192.0'
    option proto 'static'

config route6 'local_node_route6'
    option target 'XXXX:b4dc:4b1e::/64'
    option gateway '::'
    option interface 'client'
```

Vorsicht mit MACADRESSEN und standortbezogenen IP-Adressen (MZ/WI/ u.a.)

Appendix – 5 – Diagnosebefehle

Es gibt einige Befehle im Gluon, die sehr hilfreich sein können um etwas zu überprüfen. Ausgewählte Kommandos habe ich mal rausgezogen, eine Gesamtübersicht gibt's hier:

<https://github.com/freifunk-gluon/gluon/wiki/Commandline-administration>

Ich benutze im Zusammenhang mit der Konfiguration von VLANs gerne folgende Befehle, die nach einer SSH Anmeldung auf einem Router eingegeben werden müssen:

- Zustand der Ports und angelegte VLANs auf diesem Gerät

Befehl: **swconfig dev switch0 show**

Anzeige:

```
Global attributes:
  enable_vlan: 1
Port 0: → Das ist die CPU des Routers - die ist immer „up“
  pvid: 1
  link: port:0 link:up speed:1000baseT full-duplex txflow rxflow
Port 1:
  pvid: 3
  link: port:1 link:down
Port 2:
  pvid: 3
  link: port:2 link:down
Port 3:
  pvid: 0
  link: port:3 link:down
Port 4:
  pvid: 0
  link: port:4 link:up speed:100baseT full-duplex auto
VLAN 0:
  vid: 0
  ports: 3t 4t
→ Ab hier kommen die selber angelegten und gehegten VLAN-IDs
VLAN 1:
  vid: 1
  ports: 0t
VLAN 3:
  vid: 3
  ports: 0t 1 2 3t 4t
VLAN 6:
  vid: 6
  ports: 0t 3t 4t
```

- Anzahl der maximal möglichen VLANs auf diesem Gerät

Befehl: **swconfig dev switch0 help**

Anzeige:

```
→ Bei einem WDR 3500 sieht das so aus:
switch0: eth0(AR934X built-in switch), ports: 5 (cpu @ 0), vlans: 16
--switch
  Attribute 1 (int): enable_vlan (Enable VLAN mode)
  Attribute 2 (none): apply (Activate changes in the hardware)
  Attribute 3 (none): reset (Reset the switch)
--vlan
  Attribute 1 (int): vid (VLAN ID)
  Attribute 2 (ports): ports (VLAN port mapping)
--port
  Attribute 1 (int): pvid (Primary VLAN ID)
  Attribute 2 (unknown): link (Get port link information)
```

- Sind alle Mesh-Interfaces online (mesh-Blöcke in der network Datei)?

Befehl: **batctl if**

Anzeige:

```
br-mesh_vlan6: active    ← mesh über einen LAN Port
br-wan: active          ← mesh über den WAN Port
primary0: active
mesh0: active           ← mesh über Funkmodul 2,4 GHz (Radio 0)
mesh1: active           ← mesh über Funkmodul 5 GHz (Radio 1)
```