

Package OPT_FREIFUNK für FLI4L

Ulrich Wachtel

E-Mail: freifunk@wachtelnet.de

Inhaltsverzeichnis

Einleitung.....	4
Features.....	4
Installation.....	5
Hardware- Empfehlung.....	5
Abhängigkeiten:.....	5
Konfiguration.....	6
Schalter zur Paket- Aktivierung.....	6
Generelle OLSRD- Einstellungen.....	6
Debug Level:.....	6
Ip Version:.....	7
Start ohne Interface (AllowNoInt):	7
Pollrate:.....	7
TC redundancy.....	7
MPR coverage.....	7
LinkQualityFishEye.....	8
Link quality level.....	8
Link quality window size.....	8
LinkQualityDijkstraLimit optimization.....	8
OLSRD Willingness.....	9
Use Hysteresis.....	9
Hysteresis parameter.....	9
IPC Connect Werte.....	9
Max IPC Connections	9
IPC Hosts.....	10
OLSRD interface parameter.....	10
Anzahl der Interfaces.....	10
Name des Interfaces.....	10
BSSID.....	10
RTS.....	11
Fragmentierung.....	11
Sendeleistung.....	11
Hello interval	12
HELLO validity time.....	12
TC interval.....	12
TC validity time.....	12
MID interval.....	13
MID validity time.....	13
HNA interval.....	13
HNA validity time.....	13
LinkQualityMult.....	13
Zusätzliche HNA- Routen.....	14
Plugins.....	14
OLSRD dyn_gw_fli4l.....	14
OSLRD nameservice	14
OLSRD http-server.....	15
OLSRD dotdraw.....	16
OLSRD PGRAPH.....	17
„HORST“- Tool.....	17
Vorbereitung für den Start.....	17
Start des „HORST“- Tools:	17
Komfortabler HNA.....	18
ToDo – Was noch fehlt.....	19

Links/ Referenzen.....	19
Danksagung.....	19

Einleitung

Dieses Paket wurde von mir mit der freundlichen Unterstützung des Entwicklers der s.g. „Freifunk-Firmware“ für „WRT54“ Sven-Ola Tücke erstellt und enthält neben dem OLSR- daemon weitere für das Freifunk Projekt wichtige Optimierungen. Damit ist ein FLI4L mit diesem Paket nahezu softwarekompatibel mit einem anderen im Freifunk- Netz benutztem Gerät (z.B. WRT54).

So wurden wichtige das Netz stabilisierende Features standardmäßig eingebaut wie z.B. eine feste BSSID oder Dijkstralimit und Fisheye- Optimierung.

Weitere Optimierungen wie z.B. die bevorzugte Behandlung der OLSRD- Pakete können relativ einfach mit dem QOS- Paket von FLI4L realisiert werden. Einen kompletten Satz von Beispiel-Konfigurationsdateien für ein FLI4L- Freifunk- Gerät werden mit der Version 1.1 bereitgestellt. Für HNA aber auch für die anderen Knoten ist mittels „Traffic Shaping“ aus dem OPT_QOS- Paket auf einfache Art und Weise die vorhandene Bandbreite sinnvoll zu verteilen.

Die Software wurde bisher in einem Netz zusammen mit 1 WRT54GL, 1 Asus WL-HDD2.5 getestet. Ein FLI4L fungiert dabei als HNA, ein weiterer als normaler „Nicht- HNA“- Knoten. Besonders positiv machte sich hierbei die festgestellte BSSID auf die Stabilität der Verbindung zum FLI4L- HNA bemerkbar, da die verwendete WLAN- Karte ohne dieses Feature dazu neigte, ab und an ihre eigene BSSID vorzugeben. Längere Stabilitätstests stehen aus.

Features

- Dijkstralimit- Optimierung (Patch)
- Fisheye- Optimierung
- Fest einstellbare BSSID (z.B. „02:ca:ff:ee:ba:be“)
- Einstellbarer Sendepiegel (TXPOWER)
- Einstellbare FRAG und RTS- Parameter
- OLSRD- Defaultwerte (für Freifunk Projekt optimiert)
- LinkQualityMult (Abwertung einer IP)
- Olsrd-GUI (olsrd-httpinfo Plugin)
- Olsrd_dyngwfli4l- Plugin (Ein für FLI4L optimiertes olsrd_dyngwplain- Plugin)
- Plugin zum Internet- DNS announce und DNS auslesen (Olsrd_nameservice Plugin)
- Olsrd_dot_draw und Olsrd_pgraph Plugins zur Erstellung von Netzgraphen.
- HORST- Tool (WLAN- trace tool)
- Olsrd_secure Plugin (nicht benutzt für Freifunk)

Installation

Das OPT_FREIFUNK passt auf keine Disketten- Installation.

Daher sollte eine FLI4L- Installation auf HD, CD oder Flashkarte erfolgen.

Es werden folgende Ressourcen benötigt:

Memory: ca. 24 MB

HD/ Flashcard: ca. 7 MB

Gerechnet ist hier nicht die Minimalkonfiguration sondern alle unten genannten Pakete für den komfortablen HNA mit diversen Zusatzfunktionen.

Hardware- Empfehlung

Bisher wurde das Paket mit der folgenden WLAN- Karte getestet:

Level one - WNC-0300 (Atheros Chipset)
(S/N: 06011900551)

Diese Karte erreicht eine Sendefeldstärke von 18 dBm und hat eine gute Empfangsempfindlichkeit. Es werden Modi bis 108 Mbps unterstützt. Mit dem FLI4L- Treiber ist ein sicherer Betrieb im 54 Mbps- Mode möglich. Der Preis der Karte ist OK.

Bisher sehr stabile Zusammenarbeit mit dem alten MADWIFI- Treiber.

Abhängigkeiten:

Für die korrekte Funktion von OPT_FREIFUNK werden folgende OPT- Pakete benötigt:

<i>Packet</i>	<i>Beschreibung</i>
OPT_WLAN	Das Paket enthält auch die „wireless tools“, die von diesem Paket auch benutzt werden. Außerdem bindet es einen WLAN- Treiber Modul ein.
ATH_PCI	Dieser Treiber wird vorzugsweise benutzt, da

<i>Packet</i>	<i>Beschreibung</i>
	dieser es in Verbindung mit einer Karte von Atheros ermöglicht, die BSSID fest auf 02:ca:ff:ee:ba:be zu setzen. Natürlich arbeitet das System auch mit anderen von FLI4L angebotenen WLAN- Treibern, jedoch ist dann mit den Problemen zu rechnen, die bei nicht festgesetzter BSSID auftreten.
OPT_EASYCRON	Das Paket wird benötigt um regelmäßig die Domainnamen im OLSR- Netzwerk in Verbindung mit dem DNS- Plugin aufzufrischen. Ausserdem kann es benutzt werden, um den AP bei instabilem Verhalten des OLSR- Daemons regelmäßig zu rebooten bzw. den OLSRD neu zu starten.
OPT_HTTPD	Dieses Paket wird ab der Version 1.1 benötigt, um die Freifunk- GUI zu realisieren.
OPT_WGET aus dem Tools- Paket	Dieses Paket wird durch die in Version 1.1 realisierte Freifunk- GUI benutzt für den Fall, dass das httpinfo- plugin aktiviert wurde.

Konfiguration

In diesem Kapitel wird die Konfiguration des OPT_FREIFUNK erläutert.

Schalter zur Paket- Aktivierung

Gültige Werte: yes / no

OPT_FREIFUNK='yes'

Generelle OLSRD- Einstellungen

Debug Level:

Diese Einstellung ist nur für Entwickler geeignet und sollte im Normalfall nicht verändert werden.

Gültige Werte: 0-9.

Standardwert: 0

Mit dem Wert 0 läuft der OLSRD daemon im Hintergrund. Es wird kein Debug- Logfile geschrieben und es erfolgen keine Ausgaben auf die Konsole. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird. Der maximale Output wird im DebugLevel 9 erzielt.

FREIFUNK_DEBUGLEVEL=""

Ip Version:

Hier kann die zu verwendende IP- Version eingestellt werden. Bisher wird jedoch nur IP Version 4 benutzt. Daher den Wert bitte nicht ändern. Das Plugin dyngwfli4l unterstützt bisher auch nur IP V4. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Gültige Werte: 4 oder 6.

Standardwert: 4

FREIFUNK_IPVERSION=""

Start ohne Interface (AllowNoInt):

Diese Variable erlaubt es, den OLSRD auch dann zu starten, wenn noch kein Interface aktiv ist. Dieses Feature ist insbesondere bei Verwendung von „hotswap“ Geräten wie z.B. PCMCIA/ USB WLAN- Module. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Gültige Werte: yes/no.

Standardwert: yes

FREIFUNK_ALLOWNOINT=""

Pollrate:

Die Pollrate wird vom s.g. „event scheduler“ für die Ereignis- Abfrage benutzt. 0.2 bedeutet, dass alle 0,2 Sekunden nach neuen Ereignissen geprüft wird. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Gültige Werte: Sekunde (Positive Gleitpunktzahl)

Standardwert: 0.1

FREIFUNK_POLLRATE=""

TC redundancy

Dieser Parameter beschreibt wie viele Informationen über Nachbarn in der TC Meldung gesendet werden sollen. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Gültige Werte:

0: Sendet nur MPR selectors

1: Sendet MPR selectors und MPR's

2: Sendet alle Nachbarn

Standardwert: 2

FREIFUNK_TCREDUNDACY=""

MPR coverage

Dieser Parameter gibt an wie viele MPR's ein Knoten auswählen soll um einen 2 Hop Nachbarn zu erreichen. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 7

FREIFUNK_MPRCOVERAGE=""

LinkQualityFishEye

Dieser Parameter gibt an, ob der sogenannte „Fisheye- Mechanismus“ für TC- Meldungen benutzt werden soll. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Gültige Werte:

1: bedeutet: benutzen

2: bedeutet: nicht benutzen

Standardwert: 1

FREIFUNK_LINKQUALITYFISHEYE=""

Link quality level

Der Parameter entscheidet ob und wofür die Linkqualität zu verwenden ist. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Gültige Werte:

0: Verwende nicht link quality. OLSRD läuft im "RFC3626 mode"

1: Verwende link quality für die MPR- Selektion

2: Verwende link quality für die MPR- Selektion und Routing

Standardwert: 2

FREIFUNK_LINK_QUALITY_LEVEL=""

Link quality window size

Der Wert gibt die Fenstergröße für die Ermittlung der Linkqualität an. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 100

FREIFUNK_LINKQUALITYWINSIZE=""

LinkQualityDijkstraLimit optimization

Im Freifunk- Projekt wird die s.g . „LinkQualityDijkstraLimit“ Optimierung benutzt.

Gültige Werte:

Leer: Bei leerem Wert wird diese Optimierung nicht verwendet.

0 5.0: Optimierung wird verwendet.

Es ist kein Standardwert definiert. Im Freifunk- Netz sollte dieser Wert verwendet werden.

FREIFUNK_DIJKSTRALIMIT='0 5.0'

OLSRD Willingness

Dieser Parameter gibt an, ob der betreffende Knoten bereit ist, als Relay für seine Nachbarn zu arbeiten. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Gültige Werte: 0-7

0: Der Knoten gestattet überhaupt keine Relay- Funktion.

4: Neutrale Option

7: Der Knoten gestattet immer die Relay- Funktion.

Standardwert: 3

FREIFUNK_WILLINGNESS=""

Use Hysteresis

Der Parameter gibt an, ob Hysterese verwendet werden soll oder nicht. Im Freifunk Netz wird die Hysterese nicht verwendet.

Es ist kein Standardwert definiert.

FREIFUNK_USE_HYSTERESIS='no'

Hysteresis parameter

Diese Parameter haben nur eine Bedeutung, falls Hysterese benutzt wird. Bitte verändere die Werte nur, wenn Du weißt, was Du tust. Die Werte sind positive Gleitpunktzahlen im Intervall von 0.1.

„HystThrLow“ muss kleiner als der Wert „HystThrHigh“ sein. Es existieren keine Standardwerte.

Empfohlene Werte sind:

HystScaling: 0.50

HystThrHigh: 0.80

HystThrLow: 0.30

FREIFUNK_HYSTERESIS_SCALING='0.50'

FREIFUNK_HYSTERESIS_THRHIGH='0.80'

FREIFUNK_HYSTERESIS_THRLOW='0.30'

IPC Connect Werte

Max IPC Connections

Der Parameter definiert, wie viele simultane IPC Verbindungen aufgebaut werden können. Der Wert 0 schaltet den IPC- Mechanismus ab. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 0

FREIFUNK_IPC_MAX_CONNECTIONS=""

IPC Hosts

Der Parameter gibt an welche Hosts sich mit dem OLSRD per IPC verbinden dürfen. Standardmäßig darf sich nur der lokale Host (127.0.0.1) verbinden.

Es können hier weitere Rechner hinzugefügt werden. Mehrere Einträge sind erlaubt. Der lokale Rechner 127.0.0.1 muss nicht explizit definiert werden, da dieser bereits intern definiert wurde.

FREIFUNK_IPC_HOST_N='2'

FREIFUNK_IPC_HOST_1='192.168.250.4'

FREIFUNK_IPC_HOST_2='192.168.0.4'

Es können auch ganze Netze freigeschaltet werden. Mehrere Einträge sind erlaubt.

FREIFUNK_IPC_NET_N='2'

FREIFUNK_IPC_NET_1='192.168.250.0:255.255.255.0'

FREIFUNK_IPC_NET_2='192.168.1.0:255.255.255.0'

OLSRD interface parameter

Es können prinzipiell mehrere OLSRD Interfaces definiert werden. In der jetzigen Version werden die auf Schnittstellen bezogenen Interface Parameter für jedes Interface getrennt und die OLSRD-Parameter für alle Interfaces identisch eingerichtet.

Anzahl der Interfaces

Der Parameter gibt an, wieviele Interfaces (Schnittstellenkarten) existieren.

FREIFUNK_INTERFACE_N='2'

Name des Interfaces

Es sind hier gültige wireless Interfaces des FLI4L zu verwenden.

FREIFUNK_INTERFACE_1='eth1'

BSSID

Es hat sich im Freifunk- Projekt herausgestellt, dass eine fest definierte BSSID die Stabilität des Netzes deutlich erhöht und ein Netzsplit durch fehlerhafte Nodes ausgeschlossen werden kann. Es können sich nur noch Stationen mit gleicher BSSID verbinden und die festgesetzte BSSID kann von einer anderen Station nicht mehr verändert werden.

Dieser Modus wird vom OPT_FREIFUNK unterstützt, setzt jedoch zwingend den MADWIFI-

Treiber (old) von FLI4L und entsprechende Atheros- kompatible Hardware voraus. Wird ein anderer Treiber verwendet wird dieser Parameter keine Wirkung zeigen.

Im Berliner Freifunk Netz ist die BSSID „02:ca:ff:ee:ba:be“ einzustellen. Bitte informiere Dich, welche BSSID in Deinem Netz benutzt wird.

FREIFUNK_INTERFACE_1_BSSID='02:ca:ff:ee:ba:be'

RTS

RTS aktiviert einen s.g. Handshake- Mechanismus, der das Verhalten bei einer großen Anzahl von Nodes verbessert. Im Freifunk Netz wird der Wert 250 verwendet.

FREIFUNK_INTERFACE_1_RTS='250'

Fragmentierung

Fragmentierung verringert die Paketgröße und kann das Verhalten bei sehr diechten Netzen mit vielen Kollisionen verbessern. Falls der Wert im Freifunk Netz benutzt werden soll, wird der Wert 512 empfohlen.

Gültige Werte:

leer: keine Fragmentierung auf kleinere Pakete.

512: Dieser Wert wird im Freifunk Netz empfohlen.

FREIFUNK_INTERFACE_1_FRAG=""

Sendeleistung

Nicht immer muss der Sender mit der vollen Leistung senden. Z.B. wenn man den Gewinn der Antenne und damit den Gesamtgewinn erhöht kann man die Sendeleistung auf einen vom Gesetzgeber vorgeschrieben Wert drosseln. Es ist ein numerischer Wert, der die Leistung in mW angibt.

Gültige Werte:

leer: Standardwert der Karte wird nicht verändert.

xxx: Numerischer positiver Wert. Einheit muss nicht angegeben werden. Zahlenwert in mW.

FREIFUNK_INTERFACE_1_TXPOWER=""

Achtung: Mitunter sind die einstellbaren Werte der Sendeleistung an den Schnittstellenkarten nach oben hin begrenzt und können nicht beliebig erhöht werden.

Ein zweiter Interface- Eintrag sieht dann so aus:

```
FREIFUNK_INTERFACE_2='eth2'  
FREIFUNK_INTERFACE_2_BSSID='02:ca:ff:ee:ba:be'  
FREIFUNK_INTERFACE_2_RTS='250'  
FREIFUNK_INTERFACE_2_FRAG='512'  
FREIFUNK_INTERFACE_2_TXPOWER=''
```

Hello interval

Über das OLSRD Interface werden „Hello“ Meldungen gesendet. Dieser Parameter bestimmt das Sende Intervall dieser Pakete (in Sekunden/ Fliesskomma- Wert). Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 5.0 (= 5 Sekunden)

```
FREIFUNK_HELLO_INTERVAL=''
```

HELLO validity time

Der OLSRD gibt in der „Hello“ Meldung eine Gültigkeitsdauer (Hello validity time) an (in Sekunden/ Fliesskomma- Wert). Diese muss größer sein als das „Hello“ Intervall. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 90.0 (= 90 Sekunden)

```
FREIFUNK_HELLO_VALIDITY_TIME=''
```

TC interval

Über das OLSRD Interface werden „TC“ Meldungen gesendet. Dieser Parameter bestimmt das Sende Intervall dieser Pakete (in Sekunden/ Fliesskomma- Wert). Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 2.0 (= 2 Sekunden)

```
FREIFUNK_TC_INTERVAL=''
```

TC validity time

Der OLSRD gibt in der „TC“ Meldung eine Gültigkeitsdauer (TC validity time) an (in Sekunden/ Fliesskomma- Wert). Diese muss größer sein als das „TC“ Intervall. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 270.0 (= 270 Sekunden)

```
FREIFUNK_TC_VALIDITY_TIME=''
```

MID interval

Über das OLSRD Interface werden „MID“ Meldungen gesendet. Dieser Parameter bestimmt das Sende Intervall dieser Pakete (in Sekunden/ Fliesskomma- Wert). Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 15.0 (= 15 Sekunden)

FREIFUNK_MID_INTERVAL=""

MID validity time

Der OLSRD gibt in der „MID“ Meldung eine Gültigkeitsdauer (MID validity time) an (in Sekunden/ Fliesskomma- Wert). Diese muss größer sein als das „MID“ Intervall. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 90.0 (= 90 Sekunden)

FREIFUNK_MID_VALIDITY_TIME=""

HNA interval

Über das OLSRD Interface werden „HNA“ Meldungen gesendet. Dieser Parameter bestimmt das Sende Intervall dieser Pakete (in Sekunden/ Fliesskomma- Wert). Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 15.0 (= 15 Sekunden)

FREIFUNK_HNA_INTERVAL=""

HNA validity time

Der OLSRD gibt in der „HNA“ Meldung eine Gültigkeitsdauer (HNA validity time) an (in Sekunden/ Fliesskomma- Wert). Diese muss größer sein als das „HNA“ Intervall. Eine leere Variable bedeutet hier, dass der Standardwert benutzt wird.

Standardwert: 90.0 (= 90 Sekunden)

FREIFUNK_HNA_VALIDITY_TIME=""

LinkQualityMult

Dieses Flag dient dazu, einzelne Links über die IP abzuwerten. Dies geschieht mit einem Faktor, nachdem das Feature aktiviert worden ist. In der Version 1.0 ist derzeit nur eine IP- Adresse möglich. In Version 1.1 werden mehrere IP- Adressen möglich sein.

Dieser Schalter schaltet das Feature ein:

Gültige Werte: yes/ no

FREIFUNK_LINK_QUALITY_MULT='no'

Hier wird die betroffene IP- Adresse angegeben:

FREIFUNK_LINK_QUALITY_MULT_IPADDR='104.15.15.99'

Hier wird der Faktor für die betreffende IP- Adresse angegeben:

Gültige Werte: Fließkomma- Zahl.zwischen 0.1 und 0.9

FREIFUNK_LINK_QUALITY_MULT_QUALITY='0.5'

Zusätzliche HNA- Routen

Im Normalfall gibt FLI4L als HNA schon eine Default- Route vor, wenn FLI4L als Router arbeitet und mit dem Internet verbunden ist. Dann muss man hier keinen Eintrag angeben. Außerdem werden Default- Routen auch viel besser dynamisch durch das Plugin DYNGWFLI4L ermittelt.

Arbeitet FLI4L nicht als HNA, dann sorgt der OLSR- Daemon dafür, dass die richtigen Routen und auch die Default- Route richtig gesetzt wird.

Diese Option gestattet es darüber hinaus statische Routen anzugeben. Von mir wird dieses Feature derzeit nicht genutzt.

FREIFUNK_HNA4_ROUTE_N='0'

FREIFUNK_HNA4_ROUTE_1='192.168.250.1:255.255.255.0'

FREIFUNK_HNA4_ROUTE_2='192.168.1.1:255.255.255.0'

Plugins

OLSRD dyn_gw_fli4l

Dieses OLSRD- Plugin ist ein speziell für FLI4L angepasst worden. Es ermittelt automatisch den Zustand der pppoe- Verbindung und steuert damit das Senden der HNA- Pakete. Damit erhält das OLSR- Netzwerk automatisch eine Information darüber, ob der HNA noch zur Verfügung steht oder nicht. Es werden alle Dialmodi von FLI4L berücksichtigt.

Das Plugin arbeitet nur mit IP- Version 4.

Es wird nur PPPOE als Internet- Zugangsart unterstützt.

Schalte hier auf „yes“, wenn Dein FLI4L als HNA dienen soll. Schalte auf „no“, wenn Dein FLI4L kein HNA ist.

Gültige Werte: yes/ no

OPT_FREIFUNK_DYNGWFLI4L='yes'

OLSRD nameservice

Dieses OLSRD- Plugin dient dazu Hostnamen im OLSR- Netzwerk zu veröffentlichen als auch dem eigenen System verfügbar zu machen. Zusätzlich kann mit dem Plugin auch ein HNA- DNS- Name im OLSR- Netzwerk veröffentlicht werden.

Hintergrund: in jedem FLI4L werkelt ein „dnsmasq“ als Nameserver, der im lokalen Netzwerk in der Regel als Nameserver Cache dient und von den Clients als Nameserver angegeben wird.

Das Plugin liefert zum einen den eigenen Rechnernamen in das OLSR- Netzwerk aus.

Zum anderen werden die empfangenen Namen in einer Datei (var/run/hosts_olsr) eingetragen, die der dnsmasq mit Hilfe eines Cron- Jobs regelmäßig alle 55 Minuten ausliest. Damit sind auch im lokalen Netz die im OLSR- Netzwerk veröffentlichten Rechnernamen bekannt und können verwendet werden. Dabei findet der NS_Suffix Verwendung, um eindeutige Namen im OLSR- Netzwerk sicherzustellen.

Aktivieren des Plugins:

OPT_FREIFUNK_NAMESERVER='yes'

Gültige Werte: yes/ no

Suffix, der allen empfangenen Namen angehängt wird:

FREIFUNK_NS_SUFFIX='.olsr'

Prinzipiell kann man mit diesem Plugin auch DNS- Nameserver Adressen ausliefern, die zum Auflösen von Internet- Namen dienen können. Zum Beispiel die eigene Adresse, wenn man selbst HNA ist und der eigene dnsmasq Internet- Namen auflösen kann.

Dies kann man mit NS_ANNOUNCE_DNS und NS_DNS bewerkstelligen.

FREIFUNK_NS_ANNOUNCE_DNS ist der Schalter und aktiviert die Auslieferung einer DNS- Nameserver- Adresse und FREIFUNK_NS_DNS ist die entsprechende IP- Adresse.

Man sollte FREIFUNK_NS_ANNOUNCE_DNS immer auf „no“ setzen, wenn man kein HNA ist.

FREIFUNK_NS_ANNOUNCE_DNS='no'

FREIFUNK_NS_DNS='104.15.15.15'

Allerdings sollte man sich vergegenwärtigen, dass damit die empfangenden Stationen im OLSR- Netz, die ausschließlich diese dynamische Information benutzen, Probleme bekommen, wenn der entsprechende HNA einmal ausfällt. Dann wären zwar noch genügend andere Routen da, aber der DNS wäre nicht mehr verfügbar. Da das Feature optional ist kann man nicht davon ausgehen, dass alle HNA auch dieses verwenden und damit ist der Ausgang offen. Daher empfehle ich hier statische Einträge von Internet- DNS für die DNS- Server zu verwenden. Diese werden wie bisher im „base“- package eingetragen.

Aus all diesen Gründen habe ich auch in diesem Plugin darauf verzichtet die Option "resolv-file" zu verwenden, die das File /etc/resolv.dnsmasq angibt. Diese Datei enthält normalerweise die DNS- Server- Adressen und würde durch das Plugin überschrieben.

OLSRD http-server

Der OLSR Daemon ermöglicht auch die Verwendung einer GUI. Diese GUI entspricht jedoch nicht der in der „Freifunk- Firmware“ gebräuchlichen GUI.

Aus Aufwandsgründen wird es in Version 1.0 nur diese GUI geben.

In Version 1.1 wird es dann eine GUI geben, die mit der „Freifunk- Firmware“ kompatibel sein wird. Allerdings wird auf den Admin- Teil verzichtet, da FLI4L prinzipiell anders administriert wird als eine reguläre Linux- Distribution.

Dies hat auch den Vorteil, dass die Admin- GUI über einen anderen Port erreichbar ist als die Status- GUI. Somit kann mit „IPTABLES- Regeln“ die Admin- GUI in das Funknetz geblockt werden während die Status- GUI sichtbar bleibt.

GUI aktivieren:

Gültige Werte: yes/ no

OPT_FREIFUNK_HTTPINFO='yes'

Zugriffsport:

FREIFUNK_HTTPINFO_PORT='80'

Hier kannst Du die Rechner bestimmen, die auf das Admin- Interface zugreifen dürfen:

FREIFUNK_HTTPINFO_HOST_N='0'

FREIFUNK_HTTPINFO_HOST_1='192.168.252.10'

FREIFUNK_HTTPINFO_HOST_2='192.168.250.10'

FREIFUNK_HTTPINFO_HOST_3='192.168.250.4'

Aber auch ganze Netze wie z.B. das OLSR- Netz kannst Du freischalten. Natürlich ist es auch sinnvoll, die internen Netze freizuschalten:

FREIFUNK_HTTPINFO_NET_N='3'

FREIFUNK_HTTPINFO_NET_1='192.168.250.0:255.255.255.0'

FREIFUNK_HTTPINFO_NET_2='192.168.252.0:255.255.255.0'

FREIFUNK_HTTPINFO_NET_3='104.0.0.0:255.0.0.0'

ACHTUNG: Zu diesem Plugin ist noch zu bemerken, dass es noch einige Bugs enthält, die dazu führen können, dass sich der gesamte OLSRD aufhängt und dann kein Routing mehr funktioniert. Daher empfehle ich, das Plugin nur in der Testphase zu benutzen.

OLSRD Secure

Das Feature gestattet es Signatur- Meldungen im OLSR- Netz zu verschicken und somit gesicherte routing domains zu schaffen. Dafür muss ein entsprechendes Keyfile auf der Harddisk gespeichert werden.

Dieses Feature wird im Freifunk- Netz nicht verwendet!

Einschalten des Features:

Gültige Werte: yes/ no

OPT_FREIFUNK_SECURE='no'

Lage des Keyfiles:

FREIFUNK_SECURE_KEYFILE='/boot/data/keyfile'

OLSRD dotdraw

Das Plugin kann verwendet werden um Topologie Graphen zu erstellen. Mittels eines „Perl“-Scripts, „graphviz“ und „imagemagick“ können auf einem abgesetzten Rechner die Graphen erzeugt werden.

Das Script ist verfügbar unter:

[WWW] <http://meshcube.org/nylon/utils/olsr-topology-view.pl>

Aktivieren des Features:

Gültige Werte: yes/ no

OPT_FREIFUNK_DOTDRAW='yes'

Zugelassener Host:

FREIFUNK_DOTDRAW_HOST='192.168.250.4'

Zugelassener Port:

FREIFUNK_DOTDRAW_PORT='2004'

OLSRD PGRAPH

Dies ist ein weiteres Plugin zur Erstellung von Topologie Graphen.

Aktivieren des Features:

Gültige Werte: yes/ no

OPT_FREIFUNK_PGRAPH='yes'

„HORST“- Tool

Das so genannte „Horst“- Tool ist ein komfortables WLAN- Scan- Tool und kann z.B. gut verwendet werden, um die Sendeleistungen der Nachbarstationen zu ermitteln.

HORST bedeutet: „Horsts OLSR Radio Scanning Tool“.

Vorbereitung für den Start

Die Schnittstelle muss in den Monitormode gebracht werden, um das „HORST“- Tool aktivieren zu können.

Achtung: Der Monitormode funktioniert nicht mit jeder WLAN- Karte.

Die Aktivierung des Monitormodes erfolgt mit dem folgenden Befehl:

„iwconfig eth1 mode monitor “

ACHTUNG: Bei Atheros- Karten ist in Verbindung mit dem alten Atheros- Treiber kein normaler Traffic mehr möglich, wenn der Monitormode aktiv ist.

Daher muss nach Nutzung von „HORST“ der Modus wieder zurück auf „Ad-Hoc“ gestellt werden.

Dies geschieht mit: ***„iwconfig eth1 mode ad-hoc“***

Start des „HORST“- Tools:

Gib einfach ein: ***„horst“***.

Nun startet das „Horst“- Tool und kann verwendet werden.

Komfortabler HNA

Für einen komfortablen FLI4L- HNA kommen bei mir folgende Pakete zum Einsatz:

<i>Paket</i>	<i>Beschreibung</i>
OPT_FREIFUNK	Dieses Paket
OPT_HTTP_LEASES	Anzeige der DHCP- Leases in Mini HTTP
OPT_ACCOUNTING	Interface Accounting
OPT_WLAN	WLAN- Treiber und Tools
ATH_PCI	ATHEROS- Treiber
OPT_EASYCRON	CRON- Daemon
OPT_HTTPD	HTTP- GUI
OPT_WGET aus dem Tools- Paket	Zeilenorientierter Browser
OPT_DNS	DNS- Server (DNSMASQ)
OPT_DNSDHCP	DHCP- Server (DNSMASQ)
IMOND	Für PC- Tray- Tool (IMONC)
OPT_CGICONFIG2	Tool zum Blocken / Freischalten von Clients und Ports für GUI
OPT_CRONY	Timeserver
OPT_HD	OPT für eine HD/CD- Installation
OPT_MC_3	Midnight Commander
OPT_OPEN_VPN	Open VPN Paket
OPT_POPTOP	PPTP- VPN
OPT_PPP	PPP für DSL und Bluetooth- Verbindung
OPT_QOS	Quality Of Service
OPT_SSHD	SSH- Daemon
OPT_WOL	Wake On LAN- Paket
OPT_USB	USB- driver für Memorystick/ Bt- stick...
OPT_BLUEZ	Bluetooth – Server für Mobile
OPT_SYMBSPOT	Packet für Symbian Handys
OPT_DSL (Nur HNA)	Anschluss an DSL
OPT_DYNDNS (Nur HNA)	DYNDNS- Paket

ToDo – Was noch fehlt

Für die Version 1.1 sind einige Erweiterungen vorgesehen, die hier aufgelistet sind:

- Freifunk- GUI
- freifunk-setbssid Tool zum Abfragen/ Verändern der Uptime
- LinkQualityMult für mehrere IP's
- Dokumentation aller FLI4L- Konfigurations- Dateien
- Tauschbörsen Filter
- MADWIFI-NG Treiber (wird erst eingesetzt, wenn dieser als stabil eingestuft wird)
- WAS EUCH NOCH SO EINFÄLLT >>>> Mail an: freifunk@wachtelnet.de

Links/ Referenzen

<i>Link</i>	<i>Beschreibung</i>
http://freifunk.net	Die deutsche Freifunk- Seite
http://olsrexperiment.de/	Die Freifunkseite der Berliner
http://madwifi.org/	Die Seite des freien LINUX- Treibers für Atheros WLAN- Karten

Danksagung

- Als erstes vielen Dank an meine Familie, die immer wieder Verständnis für mein Hobby aufbringt.
- Recht herzlichen Dank an Sven-Ola Tuecke, dem Entwickler der Freifunk- Firmware für WRT54, der mir ein wenig zur Seite gestanden hat.
- Vielen Dank auch an Horst Krause, der mir freundlicherweise die Quellen des Horst- Tools zugesandt hat.