

Paket WLAN

Version 3.10.4

Frank Meyer

E-Mail: frank@fli4l.de

Das fli4l-Team

E-Mail: team@fli4l.de

25. Oktober 2015

Inhaltsverzeichnis

1. Dokumentation des Paketes WLAN	3
1.1. WLAN - Wireless-LAN Unterstützung	3
1.1.1. WLAN-Konfiguration	3
1.1.2. Beispiele	8
1.1.3. Virtual Accesspoint (VAP)(Experimentell)	9
1.1.4. Zeitgesteuertes ein- und ausschalten mit easycron	9
1.1.5. Spendenhinweis	10
A. Anhang zum Paket WLAN	11
Abbildungsverzeichnis	12
Tabellenverzeichnis	13
Index	14

1. Dokumentation des Paketes WLAN

1.1. WLAN - Wireless-LAN Unterstützung

Achten Sie in jedem Fall darauf, dass Sie beim Einsatz von PCI Karten ein Mainboard benutzen, was mindestens die PCI 2.2 Spezifikationen erfüllt. Auf älteren Mainboard die nur PCI 2.1 oder älter unterstützen kann es zu den unterschiedlichsten Fehler kommen. Entweder startet der Computer gar nicht (er läßt sich nicht einmal einschalten), oder die WLAN-Karte wird beim PCI Scan nicht gefunden.

WLAN-Karten werden in der base.txt IP_NET_X_DEV mit wlanX angesprochen. Wenn nur eine WLAN-Karte im System ist, hat diese also den Namen wlan0.

1.1.1. WLAN-Konfiguration

OPT_WLAN Standard-Einstellung: OPT_WLAN='no'

Aktiviert das Wireless LAN Option Pack.

WLAN_WEBGUI Standard-Einstellung: WLAN_WEBGUI='yes'

Aktiviert das Webinterface für das Wireless LAN Option Pack.

WLAN_REGDOMAIN Mit dieser Variable kann man die Landesspezifischen Einstellungen anpassen. Gültige Werte sind ISO 3166-1 alpha-2 Ländercodes wie z.B. 'DE' In verschiedenen Ländern gelten verschiedene Vorgaben für die Kanalauswahl und Sendeleistungen.

WLAN_N Anzahl der voneinander unabhängigen WLAN-Konfigurationen. Steht hier eine '1' so ist das Verhalten wie in früheren Versionen von fl4l.

WLAN_x_MAC MAC-Adresse der WLAN-Karte in dieser Schreibweise:

XX:XX:XX:XX:XX:XX

Jedes X ist ein Hex-Digit der Mac-Adresse der Karte, für die diese Konfiguration gelten soll. Sollte keine der hier eingetragenen Mac-Adressen zu einer Karte passen, so wird die Konfiguration WLAN_1_* auf diese Karte angewandt und es wird eine Warnmeldung ausgegeben, die auf den Umstand hinweist. Die Warnmeldung enthält die festgestellte MAC-Adresse der Karte. Diese ist in der Konfiguration einzutragen, damit auch das Web-Interface problemlos funktionieren kann.

WLAN_x_MAC_OVERRIDE Ändert die MAC-Adresse der WLAN-Karte damit man als Client an ein WLAN mit MAC-Filter verbinden kann ohne dort den Filter anpassen zu müssen. Hilfreich bei WAN-Anbindungen, die z.B. auf die MAC-Adresse eines gelieferten WLAN-USB-Sticks gebunden sind.

WLAN_x_ESSID Die SSID ist der Name für das Funknetzwerk. Die auch "Network Name" genannte Zeichenfolge kann bis zu 32 Zeichen lang sein. Sie wird im AP eines WLAN

konfiguriert und von allen Clients, die darauf Zugriff haben sollen, eingestellt. Auch bei Ad-Hoc muß die SSID auf allen teilnehmenden Nodes identisch sein.

WLAN_x_MODE Stellt den zu verwendenden WLAN-Modus der Karte ein.

Standard-Einstellung: `WLAN_x_MODE='ad-hoc'`

Mögliche Werte:

ad-hoc	für ein Funknetz ohne Access-Point
managed	gemanagertes Funknetz mit mehreren Zellen
master	die WLAN-Karte arbeitet als Access-Point

`WLAN_x_MODE='master'` funktioniert nur mit einem geeigneten WLAN-Treiber.

WLAN_x_NOESSID Ermöglicht das Abschalten der ESSID in den Beacon Frames. Nur möglich mit Treiber `hostap_*` und Firmware `>= 1.6.3` im `WLAN_MODE='master'`

Dieses Feature ist optional und muß manuell zur `config/wlan.txt` hinzugefügt werden.

WLAN_x_CHANNEL Setzt den Übertragungskanal des Netzwerks.

Standard-Einstellung: `WLAN_x_CHANNEL='1'`

Mögliche Werte: 1-13 und 36,40,44,48,52,56,60,64,100,104,108,112,116,120,124,128,132,136,140

Bitte lesen sie die Dokumentation Ihrer WLAN-Karte um herauszufinden, welche Kanäle in Ihrem Land erlaubt sind. Sollten sie hier einen nicht erlaubten Kanal einstellen, so sind sie alleine dafür verantwortlich. In Deutschland sind die Kanäle 1-13 im Frequenzband 2,4 GHz (Modi: b und g) erlaubt. Die Kanäle im Bereich 36-140 (siehe oben) sind im 5 GHz zulässig.

Desweiteren ist der Wert '0' erlaubt, falls `WLAN_x_MODE='managed'` gesetzt ist. Dadurch wird kein expliziter Kanal eingestellt, sondern der AP auf allen verfügbaren Kanälen gesucht. Man kann dem Kanal-Wert auch einen Buchstaben a,b oder g anhängen (z.B. 5g), welcher dann den gewünschten Betriebsmodus/Frequenzband auswählt.

Ein angehängtes 'n' oder 'N' selektiert bei entsprechenden WLAN-Karten die Nutzung von 802.11n. Kleingeschrieben bedeutet: 20 MHz Kanalbreite, grossgeschrieben: 40 MHz Kanalbreite.

Großschreibung bei a/b/g sorgt bei einigen (aktuell nur `ath_pci`) Treibern dafür, dass proprietäre WLAN-Turbos aktiviert werden. Diese Option ist experimentell und kann auch wieder entfernt werden.

WLAN_x_RATE Setzt die Übertragungsgeschwindigkeit des Netzwerks.

Standard-Einstellung: `WLAN_x_RATE='auto'`

Mögliche Werte: 1,2,5.5,11,auto - Angaben in Megabit/s je nach Karte können auch noch diese Raten ausgewählt werden: 6,9,12,18,24,36,48 und 54. Bei manchen 54 MBit-Karten kann die Rate nicht angegeben werden. Hier ist dann 'auto' einzutragen.

WLAN_x_RTS Aktiviert RTS/CTS Handshake. Diese Option ist in grossen Wlans mit vielen sendenden Clients nuetzlich wenn sich die Clients gegenseitig nicht hoeren koennen sondern nur den AP. Ist diese Option aktiviert sendet der Client vor jedem Sendevorgang ein RTS mit der Bitte um Erlaubnis zum Senden und bekommt ein CTS, die Erlaubnis zum

Senden, vom AP zurueck. Damit weiss jeder Client dass ein Client sendet auch wenn er diesen Client nicht hoert. Hierdurch werden Kollisionen vermindert weil sicher gestellt ist dass immer nur ein Client sendet. Diese Option macht nur unter der oben beschriebenen Situation Sinn weil sie zusaetzlichen overhead hinzufuegt und somit die Gesamtbandbreite verringert. Durch die Verringerung von Kollisionen kann sich die Bandbreite jedoch wieder erhoehen.

Dieses Feature ist optional und muß manuell zur config/wlan.txt hinzugefügt werden.

WLAN_x_ENC_N (Überholt) Legt die Anzahl der Wireless Encryption Key's fest (WEP).

Mögliche Werte: 0-4

WLAN_x_ENC_x (Überholt) Setzt die Wireless Encryption Keys.

Mögliche Werte:

XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XX	128 Bit Hex-Key (X=0-F)
XXXX-XXXX-XX	64 Bit Hex-Key (X=0-F)
s:<5 Zeichen>	64 Bit
s:<6-13 Zeichen>	128 Bit
P:<1-64 Zeichen>	128 Bit

Das Verfahren der Key-Vergabe mit s:Text ist nicht mit der Passphrase der Windows-Treiber kompatibel. Hier bitte einen Hex-Key verwenden! Unter Windows wird der Hex-Key meist ohne die Bindestriche '-' verwendet. Die Angabe mittels P:<Text> ist kompatibel zur Passphrase der meisten Windows WLAN-Treiber (wenn nicht allen) aber nur im 128 Bit Modus. Linux erlaubt es, verschiedene Schlüssellängen zu mischen. Windows-Treiber jedoch in der Regel nicht!

WLAN_x_ENC_ACTIVE (Überholt) Legt den aktiven Wireless Encryption Key fest.

Mögliche Werte: 1-4

Diese Variable ist aufzunehmen, wenn WLAN_x_ENC_N > 0 gesetzt wird. Ansonsten optional.

WLAN_x_ENC_MODE (Überholt) Aktiviert den Encryption Mode.

Mögliche Werte:

on/off	mit oder ohne Verschlüsselung
open	nimmt auch unverschlüsselte Pakete an
restricted	nimmt nur verschlüsselte Pakete an

Sinnvoller Wert: 'restricted'

Dieses Feature ist optional und muß manuell zur config/wlan.txt hinzugefügt werden. Ist die Variable nicht vorhanden, so wird als Default 'off' angenommen, wenn kein WEP-Key definiert ist und 'restricted' wenn mindestens ein Key definiert ist.

WLAN_x_WPA_KEY_MGMT Will man statt WEP-Verschlüsselung WPA verwenden, stellt man hier den gewünschten WPA-Modus ein. Momentan wird nur WPA-PSK unterstützt, also WPA mit einem Client und Access-Point vorab bekannten Schlüssel. Dieser Schlüssel sollte sorgfältig gewählt werden und nicht zu kurz sein, da er ansonsten auch anfällig gegen Wörterbuchattacken ist.

Unterstützt werden im *managed*-Mode alle vom Wpa-Supplimenten (http://hostap.epitest.fi/wpa_supplicant/) und im *master*-Mode alle vom Hostapd (<http://hostap.epitest.fi/hostapd/>) unterstützten Karten.

Erfolgreich getestet wurden bereits Karten basierend auf den Chipsätzen von Atheros und vom hostap-Treiber unterstützte Karten (sowohl im managed als auch im master mode). Theoretisch ist auch noch Unterstützung für atmel-Karten und einige andere möglich. Hier müssen die Ersteller der entsprechenden Opts aber ihre Opt-Pakete noch entsprechend anpassen.

WLAN_x_WPA_PSK Hier wird der Schlüssel angegeben, der zur Kommunikation zwischen Client und Access-Point verwendet werden soll. Dieser Schlüssel wird in Form einer Passphrase (eines Satzes) angegeben, die mindestens 16 Zeichen lang sein muß und bis zu 63 Zeichen lang sein kann. Folgende Zeichen werden unterstützt:

a-z A-Z 0-9 ! # \$ % & () * + , - . / : ; < = > ? @ [\] ^ _ ` { | } ~

WLAN_x_WPA_TYPE Zur Auswahl stehen hier 1 für WPA1, 2 für den WPA2 (IEEE 802.11i) Modus und 3 für beide - der Client kann dann entscheiden ob er WPA1 oder WPA2 nutzen möchte. Wenn die WLAN Hardware den Standard unterstützt, so ist dem sicheren WPA2 Verfahren den Vorzug zu geben.

WLAN_x_WPA_ENCRYPTION Die Verschlüsselungsprotokolle TKIP und die erweiterte Version CCMP (AES-CTR/CBC-MAC Protocol, manchmal auch nur AES genannt) stehen hier zur Verfügung. CCMP wird eventuell nicht von älterer WLAN-Hardware unterstützt. Es können auch beide gemeinsam angegeben werden.

WLAN_x_WPA_DEBUG (Experimentell) Bei Problemen mit der WPA-Anbindung kann man diese Variable auf 'yes' setzen, um den zuständigen daemon zu umfangreicheren Ausgaben zu veranlassen. Diese kann man dann zur Diagnose der Probleme verwenden.

WLAN_x_AP Registriert diese Node bei einem Access-Point.

Hier ist die MAC-Adresse des Access-Points anzugeben. Wenn man bereits den WLAN-Mode "master" ausgewählt hat, ist dieser Eintrag leer zu lassen. Diese Option ist nur dann sinnvoll, wenn der fli4l den AP nicht von selber finden kann oder an einen bevorzugten Access-Point gebunden werden soll. Nur zum Einsatz in WLAN-Mode "managed" gedacht.

Dieses Feature ist optional und muß manuell zur config/wlan.txt hinzugefügt werden.

WLAN_x_ACL_POLICY Policy der ACL.

Standard-Einstellung: `WLAN_x_ACL_POLICY='allow'`

Beschreibt eine Aktion, der die angegebenen MAC-Adressen unterliegen:

deny	Keine der aufgelisteten MAC-Adressen erhält Zugang
allow	Nur aufgelistete MAC-Adressen erhalten Zugang
open	Alle MAC-Adressen erhalten unabhängig vom Filter Zugang

Leider werden WLAN_ACL's aktuell nur von einem Treiber sauber unterstützt: hostap_* Als Alternative bieten sich die in 3.0.x deutlich erweiterten Firewall-Möglichkeiten an.

WLAN_x_ACL_MAC_N AP-ACLs - Einschränkung der erlaubten WLAN-Stationen.

Standard-Einstellung: `WLAN_x_ACL_MAC_N='0'`

Eine Zahl größer 0 aktiviert die Access Control List (der MAC-Adressenfilter) und gibt die Anzahl der ACL-Einträge an. Die Access Control List ist eine Liste von MAC-Adressen, denen der Zugang zum Access Point (AP) erlaubt/verboten wird. Anzahl der Mac-Adressen, die definiert werden.

WLAN_x_ACL_MAC_x Mac-Adressen in der Form: 00:00:E8:83:72:92

WLAN_x_DIVERSITY Hiermit kann man einstellen, ob manuelle Antennen-Diversity aktiviert wird.

Standard-Einstellung: `WLAN_x_DIVERSITY='no'` (automatische Wahl)

WLAN_x_DIVERSITY_RX Auswahl der Empfangsantenne.

Standard-Einstellung: `WLAN_x_DIVERSITY_RX='1'`

0 = Automatische Auswahl

1 = Antenne 1

2 = Antenne 2

WLAN_x_DIVERSITY_TX Auswahl der Sendeantenne.

Standard-Einstellung: `WLAN_x_DIVERSITY_TX='1'`

WLAN_x_WPS Aktiviert den WPS-Support. Push-Button und PIN ist möglich. Es ist sinnvoll, `WLAN_WEBGUI` zu aktivieren, es sei denn es ist nur die Steuerung per Commandline gewünscht.

Standard-Einstellung: `WLAN_x_WPS='no'`

WLAN_x_PSKFILE Bei aktiviertem `PSKFILE` können neben dem unter `WLAN_x_WPA_PSK` konfigurierten Preshared Key auch weitere Client-bezogene Keys genutzt werden. Aktuell nutzt die Funktion `WLAN_x_WPS` dieses File um darüber konfigurierten Clients individuelle Keys zu geben.

Wird das File abgeschaltet sind auch bisher mit aktiviertem File verbundene WPS-Clients nicht mehr in der Lage mit dem AccessPoint zu verbinden.

WPS-Clients, die mit abgeschaltetem File verbunden wurden, sind davon nicht betroffen.

Standard-Einstellung: `WLAN_x_PSKFILE='yes'`

WLAN_x_BRIDGE Alternativ zur Angabe im Paket `ADVANCED_NETWORKING` kann hier umgekehrt angegeben werden an welche Bridge das WLAN gebunden werden soll.

Beispiel: `WLAN_x_BRIDGE='br0'`

Achtung: Entweder in Advanced-Network oder hier angeben! Nicht an beiden Stellen!

1.1.2. Beispiele

Anbindung an einen Access Point via WPA

```
OPT_WLAN='yes'
WLAN_N='1'
WLAN_1_MAC='00:0F:A3:xx:xx:xx'
WLAN_1_ESSID='foo'
WLAN_1_MODE='managed'           # Anbindung an Access Point
WLAN_1_CHANNEL='1'
WLAN_1_RATE='auto'
#
# WPA Konfiguration
#
WLAN_1_ENC_N='0'                # kein WEP
WLAN_1_WPA_KEY_MGMT='WPA-PSK'   # WPA pre shared key
WLAN_1_WPA_TYPE='1'            # WPA 1
WLAN_1_WPA_ENCRYPTION='TKIP'
WLAN_1_WPA_PSK='Deine gute Passphrase (16-63 Zeichen)'
#
# irrelevant im WPA Kontext
#
WLAN_1_ENC_N='0'
WLAN_1_ENC_ACTIVE='1'
WLAN_1_ACL_POLICY='allow'
WLAN_1_ACL_MAC_N='0'
```

Access Point mit WPA2 Verschlüsselung

```
OPT_WLAN='yes'
WLAN_N='1'
WLAN_1_MAC='00:0F:A3:xx:xx:xx'
WLAN_1_ESSID='foo'
WLAN_1_MODE='master'           # Access Point
WLAN_1_CHANNEL='1g'            # Channel 1, Modus 'g' auf einer
                                # Atheros-Karte
WLAN_1_RATE='auto'
#
# WPA Konfiguration
#
WLAN_1_ENC_N='0'                # kein WEP
WLAN_1_WPA_KEY_MGMT='WPA-PSK'   # WPA pre shared key
WLAN_1_WPA_TYPE='2'            # WPA 2
WLAN_1_WPA_ENCRYPTION='CCMP'
WLAN_1_WPA_PSK='Deine gute Passphrase (16-63 Zeichen)'
#
# MAC basierte Zugriffskontrolle auf AP
#
WLAN_1_ACL_POLICY='allow'
WLAN_1_ACL_MAC_N='0'
#
# irrelevant im WPA Kontext
#
```



```
WLAN_1_ENC_ACTIVE='1'
```

Access Point mit WEP Verschlüsselung

```
OPT_WLAN='yes'
WLAN_N='1'
WLAN_1_MAC='00:0F:A3:xx:xx:xx'
WLAN_1_ESSID='foo'
WLAN_1_MODE='master'           # Access Point
WLAN_1_CHANNEL='1'
WLAN_1_RATE='auto'
#
# WEP Konfiguration
#
WLAN_1_WPA_KEY_MGMT=''         # kein WPA
WLAN_1_ENC_N='4'               # 4 WEP-Keys
WLAN_1_ENC_1='...'
WLAN_1_ENC_2='...'
WLAN_1_ENC_3='...'
WLAN_1_ENC_4='...'
WLAN_1_ENC_ACTIVE='1'         # erster Schlüssel ist aktiv
#
# MAC basierte Zugriffskontrolle auf AP
#
WLAN_1_ACL_POLICY='allow'
WLAN_1_ACL_MAC_N='0'
#
# irrelevant für WEP Konfiguration
#
WLAN_1_WPA_TYPE='2'
WLAN_1_WPA_ENCRYPTION='CCMP'
WLAN_1_WPA_PSK='...'
```

1.1.3. Virtual Accesspoint (VAP)(Experimentell)

Bestimmte WLAN-Karten (Treiber: ath_pci, ath5k, ath9k, ath9k_htc) können auf bis zu 4 virtuelle WLAN-Karten aufgeteilt werden. (VAP)

Die WLAN-Konfiguration der virtuellen AP kann beliebig sein bis auf folgende Bedingung: Gleich sein muss: Kanal und MAC-Adresse. Anhand der mehrfach verwendeten MAC-Adresse, wird die Karte identifiziert, die aufgesplittet werden soll. Bei mehreren verbauten Karten kann dies auch mehrfach gemacht werden.

Das Basis-Device wird weiterhin wlan0 heißen (bei einer WLAN-Karte). Bei VAP dann wlan0v2 usw. Zum binden an eine Bridge bitte hier WLAN_x_BRIDGE='br0' usw. verwenden!.

Das aktuelle Maximum ist: je nach Karte und Treiber bis zu 8x Master.

1.1.4. Zeitgesteuertes ein- und ausschalten mit easycron

Mittels *easycron* (Seite ??), einem anderen Paket, kann das WLAN ein- und ausgeschaltet werden.

1. Dokumentation des Paketes WLAN

```
EASYCRON_N='2'
EASYCRON_1_CUSTOM = ''      # Jeden Abend um 24 Uhr ausschalten
EASYCRON_1_COMMAND = '/usr/sbin/wlanconfig.sh wlan0 down'
EASYCRON_1_TIME    = '* 24 * * *'

EASYCRON_2_CUSTOM = ''      # und um 8 Uhr wieder an.
EASYCRON_2_COMMAND = '/usr/sbin/wlanconfig.sh wlan0'
EASYCRON_2_TIME    = '* 8 * * *'
```

1.1.5. Spendenhinweis

Durch die großzügige Spende von 2 Ralink 2500 basierten WLAN-Karten können WLAN-Karten mit dem RT25xx Chipsatz mit fli4l in den Modi ad-hoc und managed verwendet werden. Als Treiber ist in der base.txt hierzu rt2500 auszuwählen. Die Karten wurden gespendet von:
Computer Contor, Pilgrimstein 24a, 35037 Marburg

A. Anhang zum Paket WLAN

Abbildungsverzeichnis

Tabellenverzeichnis

Index

OPT_WLAN, [3](#)

WLAN_N, [3](#)

WLAN_REGDOMAIN, [3](#)

WLAN_WEBGUI, [3](#)

WLAN_x_ACL_MAC_N, [6](#)

WLAN_x_ACL_MAC_x, [7](#)

WLAN_x_ACL_POLICY, [6](#)

WLAN_x_AP, [6](#)

WLAN_x_BRIDGE, [7](#)

WLAN_x_CHANNEL, [4](#)

WLAN_x_DIVERSITY, [7](#)

WLAN_x_DIVERSITY_RX, [7](#)

WLAN_x_DIVERSITY_TX, [7](#)

WLAN_x_ENC_ACTIVE, [5](#)

WLAN_x_ENC_MODE, [5](#)

WLAN_x_ENC_N, [5](#)

WLAN_x_ENC_x, [5](#)

WLAN_x_ESSID, [3](#)

WLAN_x_MAC, [3](#)

WLAN_x_MAC_OVERRIDE, [3](#)

WLAN_x_MODE, [4](#)

WLAN_x_NOESSID, [4](#)

WLAN_x_PSKFILE, [7](#)

WLAN_x_RATE, [4](#)

WLAN_x_RTS, [4](#)

WLAN_x_WPA_DEBUG, [6](#)

WLAN_x_WPA_ENCRYPTION, [6](#)

WLAN_x_WPA_KEY_MGMT, [5](#)

WLAN_x_WPA_PSK, [6](#)

WLAN_x_WPA_TYPE, [6](#)

WLAN_x_WPS, [7](#)