

# Wiki / WLAN Router

## Dieser Artikel wurde für die folgenden Ubuntu-Versionen getestet:

Dieser Artikel ist größtenteils für alle Ubuntu-Versionen gültig.

## Zum Verständnis dieses Artikels sind folgende Seiten hilfreich:

1. **Einen Editor öffnen**
2. **Ein Terminal öffnen**
3. **root-Rechte**
4. **Installation von Programmen**

### Inhaltsverzeichnis

1. Voraussetzungen
  1. Hardware
  2. Software
2. Konfiguration
  1. hostapd
  2. Programmparameter
  3. interfaces
  4. dnsmasq
  5. Funktionsprüfung
3. Instant-AP Skript
  1. Koexistenz mit dem Network-Manager
  2. Variante für LAN auf WLAN
4. Netzwerkbrücke
5. Systemlog abfragen
6. Ereignisse auswerten
7. Zusatzinformationen
  1. Freie DNS
8. Problembehebung
  1. Fonera WLAN-Router
9. Links



[//media-cdn.ubuntu-de.org/wiki/attachments/34/28/network-wireless.png] Um einen Ubuntu-Rechner als WLAN-Access-Point und Router zu verwenden, wird nachfolgend die Konfiguration aufbauend auf den Artikeln **Router** [https://wiki.ubuntuusers.de/Router], **Dnsmasq** [https://wiki.ubuntuusers.de/Dnsmasq] und der Konfigurationsdatei **/etc/network/interfaces** [https://wiki.ubuntuusers.de/interfaces] beschrieben. Die Konfiguration kann sowohl auf Desktop-Systemen mit grafischer Oberfläche als auch mit einer Serverversion vorgenommen werden.

Wer eine einfache und schnelle Lösung für zwischendurch sucht, um z.B. ein Android-Smartphone über WLAN mit dem Internet zu verbinden, sollte den Abschnitt **Instant-AP-Skript - Koexistenz mit dem Network-Manager** beachten.

Ab **Ubuntu 14.04** [[https://wiki.ubuntuusers.de/Trusty\\_Tahr](https://wiki.ubuntuusers.de/Trusty_Tahr)] ist dies auch über den **Network-Manager** [<https://wiki.ubuntuusers.de/Network-Manager>] und mit WPA(2)-Verschlüsselung möglich. Es wird allerdings ein **Ad-Hoc-Netzwerk** [<http://de.wikipedia.org/wiki/Ad-hoc-Netz>] erzeugt. Entsprechende Konfigurationsmöglichkeiten und Hinweise sind im Artikel **Internetverbindungs freigabe** [<https://wiki.ubuntuusers.de/Internetverbindungs freigabe>] beschrieben.

Zusätzliche Konfigurationsbeispiele, unter anderem für einen WLAN-Repeater, einen WLAN-Router mit separatem Gast-Netzwerk oder auch einen **Onion-Router** [<https://wiki.ubuntuusers.de/Tor>], sind in **Skripte/WLAN-Accesspoint Konfigurationen** [[https://wiki.ubuntuusers.de/Skripte/WLAN-Accesspoint\\_Konfigurationen](https://wiki.ubuntuusers.de/Skripte/WLAN-Accesspoint_Konfigurationen)] beschrieben. Diese können unter Ubuntu oder z.B. auch auf dem RaspberryPI™ mit **Raspbian** [<http://www.raspbian.org/>]  eingesetzt werden.

## Voraussetzungen

### Hardware

Die WLAN-Hardware und der verfügbare Treiber müssen bestimmte Voraussetzungen erfüllen, um den benötigten Access-Point im Master-Modus überhaupt aktivieren zu können. Dazu zählen beispielsweise WLAN-Karten, die durch den Treiber **hostap** (für Prism2/2.5/3 Chipsatz basierende Karten) unterstützt werden oder solche, die mit dem aktuellen mac80211-Subsystem arbeiten.

Atheros b/g-Chipsatz basierende Karte nehmen eine Sonderstellung ein, da entweder der Madwifi-Treiber oder das Modul **ath5k** genutzt werden kann. **ath5k** arbeitet mit dem mac80211-Subsystem, der Madwifi-Treiber kann die Hardware direkt ansprechen und den gewünschten Modus aktivieren. Die Basiskonfiguration sieht dann jeweils etwas anders aus. Entsprechende Beschreibungen dazu findet man auf der **Madwifi Projektseite** [<http://madwifi-project.org/wiki/UserDocs/AccessPointInterface>] .

Eine Übersicht geeigneter Systemtreiber und WLAN-Karten findet man auf **Linuxwireless.org** [<http://www.linuxwireless.org/en/users/Drivers>]  und entsprechend in **WLAN/Karten** [<https://wiki.ubuntuusers.de/WLAN/Karten>].

Abfrage dazu mittels **iw** [<https://wiki.ubuntuusers.de/iw>]:

```
iw list
```

Erfolgt eine lange Ausgabe mit den Eigenschaften der erkannten WLAN-Karte, dann wird diese durch *hostapd* unterstützt. Wichtig sind noch die Angaben **AP** und **AP/VLAN** unter Supported interface modes: bei der Ausgabe.

```
...
```

```
Supported interface modes:
```

- \* IBSS
- \* managed
- \* AP

```

* AP / VLAN
* WDS
* monitor
* mesh point

software interface modes (can always be added):
* AP / VLAN
* monitor

```

Wird stattdessen Folgendes angezeigt, so unterstützt der verwendete Treiber *hostapd* nicht.

```
n180211 not found.
```

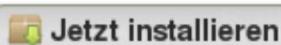
Aktuelle USB WLAN-Sticks mit Atheros-Chipsätzen, die z.B. mit den Treibern **ath9k\_htc** [[http://www.linuxwireless.org/en/users/Drivers/ath9k\\_htc](http://www.linuxwireless.org/en/users/Drivers/ath9k_htc)]  oder **carl9170** [<http://www.linuxwireless.org/en/users/Drivers/carl9170>]  arbeiten, erfordern eine Kernelversion 3.x und **hostapd** Version 0.7.3. oder höher, damit der Access-Point erstellt werden kann.

Windowstreiber unter **Ndiswrapper** [<https://wiki.ubuntuusers.de/WLAN/NdisWrapper>] unterstützen diese Funktion nicht. Mit Ndiswrapper kann nur ein Ad-Hoc-Netzwerk aufgebaut werden.

## Software

Zusätzlich benötigt werden [4]:

- **hostapd** (*universe*, der IEEE 802.11 AP/IEEE 802.1X **Authenticator** [[http://de.wikipedia.org/wiki/IEEE\\_802.1X](http://de.wikipedia.org/wiki/IEEE_802.1X)]. Erzeugt den WLAN Access-Point.)
- **dnsmasq** (*universe*, ein DHCP- und DNS-Cache, siehe auch **Dnsmasq** [<https://wiki.ubuntuusers.de/Dnsmasq>] und **Router** [<https://wiki.ubuntuusers.de/Router>])



[apt://hostapd,dnsmasq] mit **apturl** [<https://wiki.ubuntuusers.de/apturl>]

```
sudo apt-get install hostapd dnsmasq
```

## Konfiguration

Bei Desktop-Systemen mit grafischer Oberfläche muss die Netzwerkverwaltung über den **Network-Manager** [<https://wiki.ubuntuusers.de/Network-Manager>] (NM) deaktiviert werden. Über einen Rechtsklick  auf das NM-Symbol im Panel ist die Verwaltung der LAN- und WLAN-Verbindungen zu deaktivieren, um die Konfiguration über die **/etc/network/interfaces** [<https://wiki.ubuntuusers.de/interfaces>] zu ermöglichen. Der Manager sollte ansonsten vollständig deinstalliert werden, um Konflikte zu vermeiden.

Die Konfigurationsdateien können mit einem Editor [1] und Root-Rechten [3] bearbeitet werden. Hier im Beispiel wird die Internetverbindung über LAN (eth0) aufgebaut, die WLAN-Schnittstelle hat die Bezeichnung wlan0.

## hostapd

### Hinweis:

Ab Ubuntu 10.04 muss die Datei **/etc/init.d/hostapd** nicht mehr umgeschrieben werden. Der Dienst (Daemon) muss nur noch in **/etc/default/hostapd** aktiviert werden.

Auch die Datei **/etc/hostapd.conf** muss nicht mehr manuell anlegt werden. Die Konfigurationsdateien liegen alle unter **/etc/hostapd/**.

Nach der Installation aller erforderlichen Programm pakete muss zunächst die Datei **/etc/init.d/hostapd** bearbeitet werden. Nur zwei Einträge sind an beliebiger Stelle einzufügen:

```
DAEMON_CONF="/etc/hostapd.conf"      # Verweis auf die Konfigurationsdatei
RUN_DAEMON=yes                      # aktiviert hostapd bei Systemstart
...
...
```

**hostapd** bietet eine Vielzahl an Konfigurationsmöglichkeiten an, auf die hier im Detail nicht eingegangen werden kann. Eine komplette Beschreibung findet man in der Dokumentation unter **/usr/share/doc/hostapd/examples/hostapd.conf.gz**.

In der Beispielkonfiguration wird eine sichere **WPA2-AES** [<http://de.wikipedia.org/wiki/WPA2>]-Verschlüsselung verwendet. **WEP** [[http://de.wikipedia.org/wiki/Wired\\_Equivalent\\_Privacy](http://de.wikipedia.org/wiki/Wired_Equivalent_Privacy)], **WPA1-TKIP** [[http://de.wikipedia.org/wiki/Wi-Fi\\_Protected\\_Access](http://de.wikipedia.org/wiki/Wi-Fi_Protected_Access)] oder ein gemischter Modus WPA1/2 sind ebenfalls möglich. WEP ist veraltet, unsicher und sollte nicht mehr verwendet werden. WPA1-TKIP und gemischter Modus sind nur bedingt zu empfehlen. Nur WPA2-AES bietet eine sehr hohe Sicherheit.

Basiskonfiguration in der Datei **/etc/hostapd.conf** [1]:

```
# Schnittstelle und Treiber
interface=wlan0
driver=nl80211

# WLAN-Konfiguration
ssid=WLAN_AP-SSID
channel=1

# ESSID sichtbar
ignore_broadcast_ssid=0

# Ländereinstellungen
country_code=DE
ieee80211d=1

# Übertragungsmodus
hw_mode=g

# Optionale Einstellungen
# supported_rates=10 20 55 110 60 90 120 180 240 360 480 540
```

```
# Draft-N Modus aktivieren (optional, nur für entsprechende Karten)
# ieee80211n=1

# Übertragungsmodus / Bandbreite 40MHz
# ht_capab=[HT40+][SHORT-GI-40][DSSS_CCK-40]

# Beacons
beacon_int=100
dtim_period=2

# MAC-Authentifizierung
macaddr_acl=0

# max. Anzahl der Clients
max_num_sta=20

# Größe der Datenpakete/Begrenzung
rts_threshold=2347
fragm_threshold=2346

# hostapd Log Einstellungen
logger_syslog=-1
logger_syslog_level=2
logger_stdout=-1
logger_stdout_level=2

# temporäre Konfigurationsdateien
dump_file=/tmp/hostapd.dump
ctrl_interface=/var/run/hostapd
ctrl_interface_group=0

# Authentifizierungsoptionen
auth_algs=3

# wmm-Funktionalität
wmm_enabled=0

# Verschlüsselung / hier rein WPA2
wpa=2
rsn_preath=1
rsn_preath_interfaces=wlan0
wpa_key_mgmt=WPA-PSK
```

```
rsn_pairwise=CCMP
```

Das WPA-Kennwort muss mindestens 8 Zeichen und darf maximal 63 Zeichen (ASCII) bzw. 64 Zeichen (hex-Code) lang sein. **Sonderzeichen** [<https://wiki.ubuntuusers.de/WLAN/Sonderzeichen>] wie beispielsweise deutsche Umlaute sind problematisch, da die Kompatibilität zwischen verschiedenen Betriebssystemen nicht mehr gewährleistet ist und sollten vermieden werden.

Folgende Zeichen sind in der SSID und im Zugangsschlüssel erlaubt:

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

Hilfsmittel zur Erzeugung von Zugangsschlüsseln: **WPA-Schlüssel** [[https://wiki.ubuntuusers.de/WLAN/wpa\\_supplicant#Schluessel](https://wiki.ubuntuusers.de/WLAN/wpa_supplicant#Schluessel)] und **Schlüsselgenerator** [<http://darkvoice.dyndns.org/wlankeygen/>] .

## Programmparameter

Mögliche Startparameter zeigt:

```
hostapd -h
```

Konfiguration testen:

```
hostapd -dd /etc/hostapd.conf
```

## interfaces

Konfiguration der **/etc/network/interfaces**:

```
# loopback-Interface nicht löschen
auto lo
iface lo inet loopback

# LAN / automatische Konfiguration über DHCP
auto eth0
iface eth0 inet dhcp

# WLAN mit statischer Konfiguration
auto wlan0
iface wlan0 inet static
address 192.168.3.1
netmask 255.255.255.0
broadcast 192.168.3.255

# vorhandene Regeln und Ketten zuerst löschen (Restart-Funktionalität)
up /sbin/iptables -F
up /sbin/iptables -X
```

```
up /sbin/iptables -t nat -F

# Maskieren der LAN-Schnittstelle, Port-Forwarding & Nat aktivieren
up iptables -A FORWARD -o eth0 -i wlan0 -s 192.168.0.0/24 -m conntrack --ctstate NEW -j ACCEPT
up iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
up iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
up sysctl -w net.ipv4.ip_forward=1
```

Sollte die feste / statische IP für den Adapter wlan0 nicht vergeben werden, kann es hilfreich sein den Aufruf auto wlan0 aus der Konfiguration von **/etc/network/interfaces** zu entfernen.

## dnsmasq

Basiskonfiguration der **/etc/dnsmasq.conf**:

```
sudo cp /etc/dnsmasq.conf /etc/dnsmasq.conf.bak      # Originaldatei vorab sichern
```

Inhalt:

```
# DHCP-Server aktiv für Interface
interface=wlan0

# DHCP-Server nicht aktiv für Interface
no-dhcp-interface=eth0

# IP-Adressbereich / Lease-Time
dhcp-range=interface:wlan0,192.168.3.20,192.168.3.200,infinite
```

Feste IP für die angeschlossenen Clients vergeben (optional). Die Konfiguration erfolgt über die **/etc/dnsmasq.conf**

```
dhcp-host=<MAC-Adresse>,<Name>,<IP-Adresse>,infinite
dhcp-host=<MAC-Adresse>,<IP-Adresse>,infinite
dhcp-host=<Rechnername>,<IP-Adresse>,infinite
```

## Funktionsprüfung

Nun startet man das Netzwerk neu, beobachte die Terminalausgabe und überprüft anschließend die Konfiguration:

```
sudo /etc/init.d/networking restart
```

Fehlerfreie Ausgabe nach einem Neustart der Netzwerkkonfiguration:

```
# die automatisch konfigurierte LAN-Verbindung über DHCP ist hier weitestgehend
ausgeblendet
...
```

```
DHCPREQUEST of 192.168.178.24 on eth0 to 255.255.255.255 port 67
DHCPACK of 192.168.178.24 from 192.168.178.1
bound to 192.168.178.24 -- renewal in 416998 seconds.

net.ipv4.ip_forward = 1
 * Stopping advanced IEEE 802.11 management hostapd
 ...done.
 * Starting advanced IEEE 802.11 management hostapd
 ...done
```

Erfolgreiche Konfiguration der Schnittstelle für das hier gezeigte Beispiel:

```
ifconfig
iwconfig
```

Ausgabe:

```
...
mon.wlan0 Link encap:UNSPEC Hardware Adresse 00-01-36-0E-
44-C4-30-30-00-00-00-00-00-00-00-00
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metrik:1
      RX packets:3935 errors:0 dropped:0 overruns:0 frame:0
      TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
      Kollisionen:0 Sendewarteschlangenlänge:1000
      RX bytes:286055 (286.0 KB) TX bytes:0 (0.0 B)

wlan0      Link encap:Ethernet Hardware Adresse 00:01:36:0e:44:c4
      inet Adresse:192.168.3.1 Bcast:192.168.3.255 Maske:255.255.255.0
      inet6-Adresse: fe80::201:36ff:fe0e:44c4/64 Gültigkeitsbereich:Verbindung
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metrik:1
      RX packets:1018 errors:0 dropped:0 overruns:0 frame:0
      TX packets:1200 errors:0 dropped:0 overruns:0 carrier:0

wlan0      IEEE 802.11bg Mode:Master Frequency:2.412 GHz Tx-Power=20 dBm
      Retry long limit:7 RTS thr=2347 B Fragment thr=2346 B
      Power Management:off

mon.wlan0  IEEE 802.11bg Mode:Monitor Tx-Power=20 dBm
      Retry long limit:7 RTS thr=2347 B Fragment thr=2346 B
      Power Management:off
```

Die erzeugten Filtertabellen:

```
sudo iptables -t filter -n -L -v
```

Ausgabe für die Kette "*FORWARD*"

```
Chain FORWARD (policy ACCEPT 12 packets, 912 bytes)
pkts bytes target     prot opt in     out     source
```

```
destination
 0      0 ACCEPT      all  --  wlan0  eth0      192.168.0.0/24
0.0.0.0/0          ctstate NEW
 7    532 ACCEPT      all  --  *      *      0.0.0.0/0
0.0.0.0/0          ctstate RELATED,ESTABLISHED
```

```
sudo iptables -t nat -n -L -v
```

Ausgabe für die Kette "*POSTROUTING*":

```
Chain POSTROUTING (policy ACCEPT 180 packets, 17217 bytes)
 pkts bytes target      prot opt in      out      source
destination
 37  2734 MASQUERADE  all  --  *      eth0      0.0.0.0/0          0.0.0.0/0
```

Nach erfolgreicher Konfiguration sollte der Access Point bei einem manuellen Scan auf den Clients angezeigt werden:

```
sudo iwlist scan
```

Ausgabe:

```
Cell 01 - Address: 00:01:36:0E:44:C4
      ESSID: "WLAN_AP-SSID"
      Protocol: IEEE 802.11bg
      Mode: Master
      Frequency: 2.412 GHz (Channel 1)
      Quality: 87/100  Signal level: -40 dBm  Noise level: -96 dBm
      Encryption key: on
      Bit Rates: 1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 6 Mb/s
                  9 Mb/s; 12 Mb/s; 18 Mb/s; 24 Mb/s; 36 Mb/s
                  48 Mb/s; 54 Mb/s
      Extra: bcn_int=100
      Extra: atim=0
      IE: IEEE 802.11i/WPA2 Version 1
      Group Cipher : CCMP
      Pairwise Ciphers (1) : CCMP
```

Nun kann ein Verbindungsversuch gestartet werden.

[//media-cdn.ubuntu-de.org/wiki/attachments/18/40

/Netzwerk02\_Instant01.png]

## Instant-AP Skript

Über ein **Bash-Skript** [[https://wiki.ubuntuusers.de/Shell/Bash-Skripting-Guide\\_f%C3%BCr\\_Anf%C3%A4nger](https://wiki.ubuntuusers.de/Shell/Bash-Skripting-Guide_f%C3%BCr_Anf%C3%A4nger)] kann ein WLAN Access-Point erzeugt und die



vorhandene Internetverbindung über LAN

durchgereicht werden. Dies ist bei Desktopsystemen sehr praktisch, da die benötigte Konfiguration über den Network-Manager beibehalten werden kann. Die Datei **/etc/network/interfaces** muss dazu nicht verändert werden. **hostapd** ist wie beschrieben einzurichten. So kann "zwischendurch" schnell mal ein WLAN-AP erstellt werden.

Das Skript verwendet **dnsmasq-base** [https://wiki.ubuntuusers.de/Dnsmasq] zur Konfiguration. Manuelle Änderungen im Skript wie die verwendeten Schnittstellen und IP-Adressen werden für **dnsmasq** automatisch übernommen bzw. gemäß der getroffenen Einstellungen generiert.

Alle hier verwendeten **Dienste** [https://wiki.ubuntuusers.de/Dienste] wie hostapd werden durch das jeweilige Skript gestartet und auch wieder beendet. Werden diese Dienste nicht direkt bei Systemstart benötigt, so sollte man diese grundsätzlich deaktivieren, um ungewollte Systemfunktionen zu vermeiden.

Dienst bei Systemstart deaktivieren:

```
sudo update-rc.d hostapd disable
```

## Koexistenz mit dem Network-Manager

Eine zusätzliche vereinfachte **Variante** [https://forum.ubuntuusers.de/post/4876767/] des Skripts ermöglicht es, die Basisverbindung über den Network-Manager aufzubauen. So kann die Ethernet- oder z.B. auch UMTS/GPRS Verbindungen weiterhin über den NM verwaltet werden. Nur WLAN wird durch das Skript gesteuert.

## Variante für LAN auf WLAN

Dieses Skript bietet erweiterte Optionen für die Konfiguration speziell für Ethernet-Verbindungen und z.B. den Aufbau einer transparenten Netzwerkbrücke zwischen LAN und WLAN. Der Network-Manager wird allerdings deaktiviert, UMTS/GPRS Verbindungen können so nicht genutzt werden.

**Dnsmasq** [https://wiki.ubuntuusers.de/Dnsmasq] kann ebenfalls verwendet und muss dann wie bereits beschrieben nachinstalliert werden. Die erforderliche Startoption lautet -D. Eine Beispielkonfiguration der **/etc/dnsmasq.conf** ist im Skript hinterlegt und braucht nur übernommen zu werden. Manuelle Änderungen im Skript wie die verwendeten Schnittstellen und IP-Adressen sind zu berücksichtigen.

Skript mit einem Editor [1] oder über Terminal anlegen und ausführbar machen [2]:

```
touch instant_AP.sh
chmod +x instant_AP.sh
```

Folgenden Inhalt einfügen:

```

1 #!/bin/bash
2 ## Instant WLAN Access-Point
3 ## elektronenblitz63 ubuntuusers.de 2012
4 ## published under GPL v3
5 ##
6 ## Version 1.6.0 vom 31.Januar 2012
7  # Bridged-Modus möglich
```

```
8  # kleinere Fehler beseitigt
9  # Ausgaben verbessert
10 #
11 ## Version 1.5.2 vom 12.November 2011
12 # voreingestellte freie DNS geändert
13 # Restart des Network-Managers erst bei -stop
14 # entferne iptables-Filter bei -stop
15 #
16 ## Beispielkonfiguration der dnsmasq.conf
17 #
18 ## ## DHCP-Server dnsmasq aktiv für Interface
19 #
20 #
21 # interface=wlan0
22
23 ## ## DHCP-Server dnsmasq nicht aktiv für Interface
24 #
25 # no-dhcp-interface=eth0
26
27 ## ## IP-Adressbereich / Lease-Time
28 # dhcp-range=192.168.3.20,192.168.3.25,infinite
29 #
30 ## Ende Beispielkonfiguration dnsmasq.conf
31
32 ## freie Variablen
33
34 ## Konfiguration der Ethernet-Schnittstelle
35 ## LAN statisch (Standard ist automatisch über DHCP) / Startoption [-f]
36 laniface=eth0
37 laddress=192.168.178.6
38 lbroadcast=192.168.178.255
39 lnetmask=255.255.255.0
40 lgateway=192.168.178.1
41 lmacaddress=00:12:79:c0:49:ae
42 #
43 #
44 ## Konfiguration der WLAN-Schnittstelle
45 ## WLAN statisch
46 wlaniface=wlan0
47 waddress=192.168.3.1
48 wbroadcast=192.168.3.255
49 wnetmask=255.255.255.0
50 iptablemask=192.168.0.0/24
51
52 #### manuelle DNS (drei DNS Einträge, 1xDomain und 1xSearch sind möglich)
53 # Beispiel
54 # dns="nameserver 192.168.178.1 nameserver 192.168.178.1 nameserver 192.168.178.1 domain fritz.box search
55 # fritz.box"
56 dns="nameserver 8.8.4.4 nameserver 8.8.8.8 nameserver 213.73.91.35"
57
58 #
59 ## dnsmasq-base Konfiguration
60 # DHCP-Adresspool umfasst x-Adressen
61 ipaddresses=10
```

```
62
63  # Basisadresse DHCP-Adresspool (WLAN-IP + X)
64  wlanbaseip=1
65
66  # Lease-Time
67  leasetime=infinite
68
69  ## MAC-Adresse (optional) (Startoption [-m])
70  lmacaddress=00:12:79:c0:49:ae
71
72  ## Pause vor LAN-Verbindungstest
73  pause=4
74
75  ## Proxyserver (squid)
76  proxy="squid"
77
78  ## Proxy Server auf Port x (squid 3128 / tinyproxy 3128 / polipo 8123)
79  proxyport=3128
80
81  ## Bridge-Konfiguration
82  ## vor Ubuntu 11.x - /usr/sbin/brctl
83  ## ab Ubuntu 11.x - /sbin/brctl
84  br_util=/sbin/brctl
85  bridge0=br0
86  brdelay=5
87  brstp=0
88
89
90  ## Steuerung Dienste
91  ## alt
92  # hostapdrestart="/etc/init.d/hostapd"
93  ## neu
94  hostapdservice="service hostapd"
95
96
97  ## Ende freie Variablen
98  ##
99  # Skript
100 #
101 pingout=""
102
103 ## aut. Adressberechnung DHCP-Range für dnsmasq
104 ## gemäß Vorgabe WLAN-Schnittstelle
105 ipaddresses=$[$ipaddresses+$wlanbaseip]
106  baseendaddr="`echo $waddress | tr -s . \" \" | awk {'print $4'}`"
107  basestartaddr="`echo $waddress | tr -s . \" \" | awk {'print $1,$2,$3'} | tr -s \" \" .`"
108  endaddr="$basestartaddr"."[$startaddr+$ipaddresses]"
109  startaddr="$basestartaddr"."[$baseendaddr+$wlanbaseip]"
110
111 pingout=""
112 A=1
113 B=0
114 C=0
115 D=0
```

```
116  P=0
117  Br=0
118
119  if [ "$1" = "-h" ]; then
120  echo Verwendung: instant_AP.sh [-start] [-restart] [-stop] [-D] [-d] [-f] [-m] [-h]
121  [-B]
122  echo Syntax:
123  echo "sudo ./instant_AP.sh wie [-d] startet mit Standardparametern (DHCP)"
124  echo "sudo ./instant_AP.sh -f statische LAN-Konfiguration"
125  echo "sudo ./instant_AP.sh -f -m statische LAN-Konfiguration, MAC-Änderung"
126  echo "sudo ./instant_AP.sh -D verwendet dnsmasq.conf und nicht dnsmasq-base"
127  echo "sudo ./instant_AP.sh -start -f statische LAN-Konfiguration"
128  echo "sudo ./instant_AP.sh -restart -f statische LAN-Konfiguration"
129  echo "sudo ./instant_AP.sh -B Bridged-Modus ohne DHCP-Server. Dynamische und Statische
130  Schnittstellenparameter werden ignoriert"
131  echo "sudo ./instant_AP.sh -stop beendet den AP"
132  echo "Ende"
133  exit
134  fi
135
136
137  while getopts ":DdfmhPB" OPTION ; do
138  case $OPTION in
139    D) echo "wende dnsmasq.conf und nicht dnsmasq-base"; D=1 ;;
140    d) echo "konfiguriere LAN über DHCP"; A=1 ;;
141    f) echo "konfiguriere LAN statisch"; A=2 ;;
142    m) echo "MAC-Change LAN ein"; B=1 ;;
143    P) echo "Portumleitung für Proxy-Server Port" $proxyport "aktiviert"; P=1 ;;
144    B) echo "Bridge-Modus aktiviert"; Br=1 ;;
145
146  esac
147  done
148
149  echo "starte gewählte Konfiguration ..."
150  sleep 2
151
152  if [ "$Br" = "1" ]; then A=3
153  echo "Bridge-Mode aktiviert - Sonstige Parameter für Schnittstelleneinstellungen werden ignoriert"
154  fi
155
156  if [ "$1" != "-start" ]; then
157  echo "stoppe alle Dienste, und Verbindungen, lösche Itables-Filter ..."
158
159  # Konfiguration löschen
160  /sbin/iptables -F
161  /sbin/iptables -X
162  /sbin/iptables -t nat -F
163
164  defgw=`route -n | grep UG | awk {'print $2'}`
165  /sbin/route del default gw $defgw $laniface
166  echo " | tee /etc/resolv.conf
167
168  $hostapdservice stop
```

```
170  /sbin/ifconfig $wlaniface down
171  sleep 1
172  /sbin/iwconfig $wlaniface mode managed
173  sleep 1
174  /sbin/ifconfig $laniface down
175  sleep 1
176  /usr/bin/killall dnsmasq
177  /sbin/sysctl -w net.ipv4.ip_forward=0
178  /sbin/modprobe -rfv iptable_nat ipt_MASQUERADE xt_conntrack
179  iptable_filter
180
181  ## Bridge löschen
182  /sbin/ifconfig $bridge0 down
183  sleep 1
184  $br_util delif $bridge0 $laniface
185  $br_util delif $bridge0 $wlaniface
186  $br_util delbr $bridge0
187
188  if [ "$1" = "-stop" ]; then
189  echo
190  echo "aktiviere Network-Manager."
191  service network-manager start
192  service network-manager restart
193  echo "WLAN Access-Point Konfiguration beendet."
194  exit
195  fi
196  fi
197  fi
198
199  ## MAC-Adresse abgleichen
200  if [ "$B" = "1" ]; then
201  currentmac=`ifconfig $laniface | grep Adresse | awk {'print $6'}`
202  echo Schnittstelle $laniface, MAC-Adresse: $currentmac
203  echo Vorgabe: $lmacaddress
204
205  if [ "$currentmac" = "$lmacaddress" ]; then
206  echo Übereinstimmende MAC-Adresse
207  else
208  /sbin/ifconfig $laniface down
209  /sbin/ip link set dev $laniface addr $lmacaddress
210
211  currentmac=`ifconfig $laniface | grep Adresse | awk {'print $6'}`
212  echo versuche MAC-Adresse zu ändern ...
213  echo Schnittstelle $laniface, MAC-Adresse: $currentmac
214
215  if [ "$currentmac" = "$lmacaddress" ]; then
216  echo Änderung der MAC-Adresse erfolgreich!
217  else
218  echo Änderung der MAC-Adresse nicht erfolgreich!
219  echo fahre fort ...
220
221  fi
222  fi
223  fi
```

```
224
225 # Grundkonfiguration
226 echo beende Network-Manager
227 service network-manager stop
228 echo "starte alle Dienste, und Verbindungen ..."
229
230 # LAN aut.m über DHCP nur wenn kein Bridged-Mode gewählt
231 if [ "$Br" = "0" ] & [ "$A" = "1" ]; then
232 echo "starte automatische LAN-Verbindung ..."
233 /sbin/dhclient $laniface
234 fi
235
236 # LAN statisch nur wenn kein Bridged-Mode gewählt
237 if [ "$Br" = "0" ] & [ "$A" = "2" ]; then
238 echo "starte statische LAN-konfiguration ..."
239 /sbin/ifconfig $laniface down
240 sleep 2
241 /sbin/ifconfig $laniface $laddress broadcast $lbroadcast netmask $lnetmask
242 sleep 2
243 echo
244
245
246 echo setze Gateway und Route ...
247 /sbin/route add default gw $lgateway $laniface
248 sleep 1
249 echo
250
251 echo setze DNS
252 echo '# erzeugt durch instant_AdHoc.sh' | tee /etc/resolv.conf
253 echo $dns | awk {'print $1,$2'} | tee -a /etc/resolv.conf
254 echo $dns | awk {'print $3,$4'} | tee -a /etc/resolv.conf
255 echo $dns | awk {'print $5,$6'} | tee -a /etc/resolv.conf
256 echo $dns | awk {'print $7,$8'} | tee -a /etc/resolv.conf
257 echo $dns | awk {'print $9,$10'} | tee -a /etc/resolv.conf
258
259 fi
260 sleep $pause
261
262 ## Vorbereitung Bridge
263 if [ "$Br" = "1" ]; then
264
265 ## vorhandene Schnittstellenkonfiguration löschen
266 /sbin/ifconfig $wlaniface down
267 sleep 1
268 /sbin/ifconfig $wlaniface up
269 sleep 1
270 /sbin/ifconfig $wlaniface 0.0.0.0
271 sleep 1
272 /sbin/ifconfig $laniface down
273 sleep 1
274 /sbin/ifconfig $laniface up
275 sleep 1
276
277 ## Bridge anlegen
```

```

278  $br_util addbr $bridge0
279  $br_util addif $bridge0 $laniface
280  $br_util stp $bridge0 $brstp
281  $br_util setfd $bridge0 $brdelay
282  sleep 2
283
284  ## Dienste steuern
285  service dnsmasq stop
286  echo "fordere IP-Adresse für Bridge an ..."
287  /sbin/dhclient $bridge0
288  $hostapdservice restart
289  $br_util addif $bridge0 $wlaniface
290  else
291
292
293  ## ohne Bridge-Mode
294  ## WLAN-Schnittstelle statisch konfigurieren
295  echo "WLAN-Schnittstelle initialisieren ..."
296  /sbin/ifconfig $laniface up
297  sleep 1
298  /sbin/ifconfig $wlaniface $waddress broadcast $wbroadcast netmask $wnetmask
299  $hostapdservice restart
300  echo
301 echo "starte IP-Forward, Masquerading und NAT"
302 /sbin/iptables -A FORWARD -o $laniface -i $wlaniface -s $iptablemask -m conntrack
303 --ctstate NEW -j ACCEPT
304 /sbin/iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j
305 ACCEPT
306 /sbin/iptables -t nat -A POSTROUTING -o $laniface -j MASQUERADE
307 /sbin/sysctl -w net.ipv4.ip_forward=1
308 fi
309
310 ## dnsmasq-base starten
311 echo
312 echo starte dnsmasq-base
313 if [ "$D" = "0" ]; then

```

Die verwendeten Schnittstellen für LAN und WLAN, sowie die gewünschten IP-Adressen sind ggf. noch anzupassen. Alle erforderlichen Parameter können im Deklarationsblock des Skripts geändert werden.

Im Terminal [2] als **root** [3] kann das Skript nun mit verschiedenen Parametern aufgerufen werden:

```

sudo ./instant_AP.sh [-start|-restart|-stop]
sudo ./instant_AP.sh                               # startet mit automatischer Konfiguration
(wie [-start])
sudo ./instant_AP.sh -start                         # startet alle Verbindungen und Dienste
mit automatischer Konfiguration
sudo ./instant_AP.sh -restart                       # erneuert alle Verbindungen und Dienste
mit automatischer Konfiguration
sudo ./instant_AP.sh -stop                          # beendet alle Verbindungen und Dienste

```

Zusätzliche Optionen für eine statische Konfiguration der LAN-Schnittstelle und eine Änderung der

MAC-Adresse sind möglich, sollte dies für eine Verbindung erforderlich sein. Das Skript gibt entsprechende Rückmeldungen. Am Ende wird die Konfiguration der LAN- und WLAN-Schnittstelle sowie die Routingtabelle zur Funktionsprüfung ausgegeben.

```
sudo ./instant_AP.sh [-start|-restart|-stop] [-D] [-d] [-f] [-m] [-h] [-P] [-B]
sudo ./instant_AP.sh -h                                # gibt eine Hilfe zur Syntax aus
sudo ./instant_AP.sh -f                                # startet LAN mit statischer Konfiguration
sudo ./instant_AP.sh -P                                # startet Proxyserver Squid
sudo ./instant_AP.sh -start -d -m                      # startet LAN mit automatischer
Konfiguration und ändert die MAC-Adresse
sudo ./instant_AP.sh -start -m -f                      # startet LAN mit statischer Konfiguration
und ändert die MAC-Adresse
sudo ./instant_AP.sh -restart -m                      # erneuert alle Verbindungen mit
```

Die Optionen können beliebig kombiniert werden. Bei Option -df wird eine statische Konfiguration, und bei -fd eine dynamische Konfiguration der LAN-Schnittstelle ausgeführt, also immer die zuletzt angegebene Option verwendet. Parameter -stop erfordert keine Option, bzw. ignoriert diese.

Eine dazu passende Anleitung zur Konfiguration des Proxyservers Squid findet sich in **Internetverbindungs freigabe - Proxyserver** [<https://wiki.ubuntuusers.de/Internetverbindungs freigabe#Proxyserver>].

Nachdem das Skript und alle Dienste beendet, sowie die Einstellungen zurückgesetzt wurden, kann wieder der Network-Manager verwendet werden.

## Netzwerkbrücke

Sofern die WLAN-Hardware dies unterstützt, kann eine **Brücke** [<https://wiki.ubuntuusers.de/Netzwerkbrücke>] (sog. Bridging) zwischen LAN und WLAN hergestellt werden um einen gemeinsamen IP-Adressbereich für das lokale Netzwerk verwenden zu können. Die Basiskonfiguration ist in **Router** [<https://wiki.ubuntuusers.de/Router>] beschieben. Die Verbindung ist "transparent", d.h. alle angeschlossenen Geräte verhalten sich so, als wären sie direkt miteinander verbunden.

Die Hard- und Firmware der WLAN-Karte muss dies allerdings unterstützen, ansonsten funktioniert die Brücke nicht und der Adressbereich muss dann für LAN und WLAN getrennt werden. Atheros basierende Karten, z.B. AR5008 mit dem Kernelmodul **ath9k** [<http://www.linuxwireless.org/en/users/Drivers/ath9k>], unterstützen diese Funktion.

Neben der beschriebenen statischen Konfiguration für die Brücke in der **interfaces** [<https://wiki.ubuntuusers.de/interfaces>] muss **hostapd** und **dnsmasq** entsprechend konfiguriert werden. **dnsmasq** wird nur benötigt, wenn die Verbindung über DSL-Modem aufgebaut wird und kein anderer DHCP-Server vorgeschaltet ist (Router).

Folgendes Paket muss zunächst installiert [4] werden:

- **bridge-utils** (Dienstprogramme zur Konfiguration einer Netzwerkbrücke)



[apt://bridge-utils] mit **apturl** [<https://wiki.ubuntuusers.de/apturl>]

```
sudo apt-get install bridge-utils
```

Einstellungen für das oben verwendete Beispiel:

Konfiguration der Brücke in der **/etc/network/interfaces**

```
# loopback-Interface nicht löschen
auto lo
iface lo inet loopback

# Netzwerkbrücke
auto br0
iface br0 inet static
address 192.168.3.1
netmask 255.255.255.0
broadcast 192.168.3.255
bridge_ports eth1
bridge_fd 0          # Einstellungen testen, Verzögerung (forward-delay) ggf.
```

Die WLAN-Schnittstelle, hier *wlan0*, muss nicht in der Konfiguration der Bridge angegeben werden, dies erledigt *hostapd*. Das ist auch nur möglich, wenn sich die WLAN-Karte bereits im Master-Modus befindet.

Einstellung der **/etc/hostapd.conf**

```
...
interface=wlan0
bridge=br0
driver=n180211
...
```

Optional die Konfiguration der **/etc/dnsmasq.conf**

```
# DHCP-Server nicht aktiv für Interface
no-dhcp-interface=eth0          # Internetverbindung
no-dhcp-interface=eth1          # lokales Netzwerk
no-dhcp-interface=wlan0         # lokales WLAN

# DHCP-Server aktiv für Interface
interface=br0

# IP-Adressbereich / Lease-Time
dhcp-range=interface:br0,192.168.3.20,192.168.3.200,infinite
```

## Systemlog abfragen

Ereignisse zu **hostapd** und **dnsmasq** werden im Systemlogbuch unter **/var/log/syslog** abgelegt und können mit einem entsprechenden Filter ausgelesen werden:

```
cat /var/log/syslog | egrep 'hostapd|dnsmasq'
```

## Ereignisse auswerten

Um Ereignisse wie das An- oder Abmelden eines Clients auszuwerten, kann hostapd\_cli verwendet werden.

```
man hostapd_cli
```

hostapd\_cli als Daemon starten und bei entsprechenden Ereignissen ein Skript abarbeiten. Beispiel:

```
sudo hostapd_cli -p /var/run/hostapd -i [Interface] -a /Pfad/Skript -B
```

Das Controllinterface und WLAN-Interface wie z.B. wlan0 bitte entsprechend anpassen. Die Einstellung für das Controllinterface, hier der Standard `/var/run/hostapd`, wird in der Datei `/etc/hostapd.conf` festgelegt.

Einfaches Beispiel des aufgerufenen Skripts, um die Ereignisse in einer Protokolldatei abzulegen:

```
1 #!/bin/bash
2 ## Ereignisse in Textdatei speichern
3 echo $* >> /Pfad/Dateiname
```

Ereignisse, wenn sich z.B. ein Client verbindet oder trennt, werden mit Angabe der Schnittstelle und der MAC-Adresse des Clients übergeben:

```
[Interface] AP-STA-CONNECTED [MAC-Adresse]
[Interface] AP-STA-DISCONNECTED [MAC-Adresse]
```

## Zusatzinformationen

- **IP-Adressen** [[http://de.wikipedia.org/wiki/Private\\_IP-Adresse](http://de.wikipedia.org/wiki/Private_IP-Adresse)] in lokalen Netzwerken
- **Frequenzeinstellungen** [<http://de.wikipedia.org> /wiki/Wireless\_Local\_Area\_Network#Frequenzen\_und\_Kan.C3.A4le]
- **ISO Ländercodes** [[http://www.iso.org/iso/english\\_country\\_names\\_and\\_code\\_elements](http://www.iso.org/iso/english_country_names_and_code_elements)] 

## Freie DNS

- **DNS** [[http://de.wikipedia.org/wiki/Domain\\_Name\\_System](http://de.wikipedia.org/wiki/Domain_Name_System)]
- **Liste freier DNS auf www.friendly-board.de** [<http://www.friendly-board.de/forum/freie-dns-server-t552.html>] 
- **Liste freier DNS auf wiki.ak-zensur.de** [[http://wiki.ak-zensur.de/index.php/Unzensierte\\_DNS\\_Server](http://wiki.ak-zensur.de/index.php/Unzensierte_DNS_Server)] 

## Problembehebung

### Fonera WLAN-Router

Die erste Version des **Fonera-Routers** [[http://de.wikipedia.org/wiki/La\\_Fonera](http://de.wikipedia.org/wiki/La_Fonera)] verhindert, dass ein per WLAN

verbundener Client im (Heim-)Netzwerk erkannt wird. Das kann durch das Einspielen einer alternativen Firmware umgangen werden. Zuerst muss freeWLAN auf der Fonera installiert werden, dazu den Anweisungen des **freeWLAN-Wikis** [<http://trac.freewlan.info/wiki/Documentation/Installation.html#Howtoinstalltheadd-ons>]  folgen.

Wenn dies geschehen ist, im Webbrowser die Adresse <http://169.254.255.1> eingeben, zum Tab "*Advanced*" wechseln und den Punkt "*Network Settings*" auswählen. Dort findet man die Einstellung "*Private Wireless IP Settings*". Nun noch "*Yes*" auswählen und etwas warten.

## Links

- **hostapd** [<http://w1.fi/hostapd/>] Homepage 
- **hostapd Dokumentation** [<http://www.linuxwireless.org/en/users/Documentation/hostapd>]  auf [linuxwireless.org](http://www.linuxwireless.org)
- **Internet und Netzwerk** [[https://wiki.ubuntuusers.de/Internet\\_und\\_Netzwerk](https://wiki.ubuntuusers.de/Internet_und_Netzwerk)] und **Netzwerk** [<https://wiki.ubuntuusers.de/Netzwerk>]  - Übersichten zu verschiedenen Netzwerkthemen
- **Router** [<https://wiki.ubuntuusers.de/Router>]
- **Heimnetzwerk** [<https://wiki.ubuntuusers.de/Heimnetzwerk>] - Daten in einem lokalen Netzwerk (LAN) austauschen
- **WLAN** [<https://wiki.ubuntuusers.de/WLAN>]  Übersichtsartikel

---

**Diese Revision** [[https://wiki.ubuntuusers.de/WLAN\\_Router?rev=814963](https://wiki.ubuntuusers.de/WLAN_Router?rev=814963)] wurde am 6. Mai 2015 07:56 von **Chris1978** erstellt. Die folgenden Schlagworte wurden dem Artikel zugewiesen: **WLAN** [<https://wiki.ubuntuusers.de/Wiki/Tags?tag=WLAN>], **Netzwerk** [<https://wiki.ubuntuusers.de/Wiki/Tags?tag=Netzwerk>]

Inhalte von [ubuntuusers.de](http://ubuntuusers.de) lizenziert unter Creative Commons, siehe <http://ubuntuusers.de/lizenz/>.