

# WLAN mit OpenWrt auf dem WRT54GL-DE



Für unser **WLAN** bedienen wir uns eines Linux-basierten **Wireless-G Linux-Broadband-Routers WRT54GL-DE**.

Hier nutzen wir aber nicht die originale [Firmware](#), sondern greifen auf [OpenWrt](#) - besser gesagt auf **Kamikaze**, einer Linux Distribution für embedded Devices, zurück.

## Download

Von der [OpenWrt Homepage](#) laden wir uns die aktuelle Version von **Kamikaze** <http://downloads.openwrt.org/kamikaze/8.09.2-RC2/brcm-2.4/> herunter.

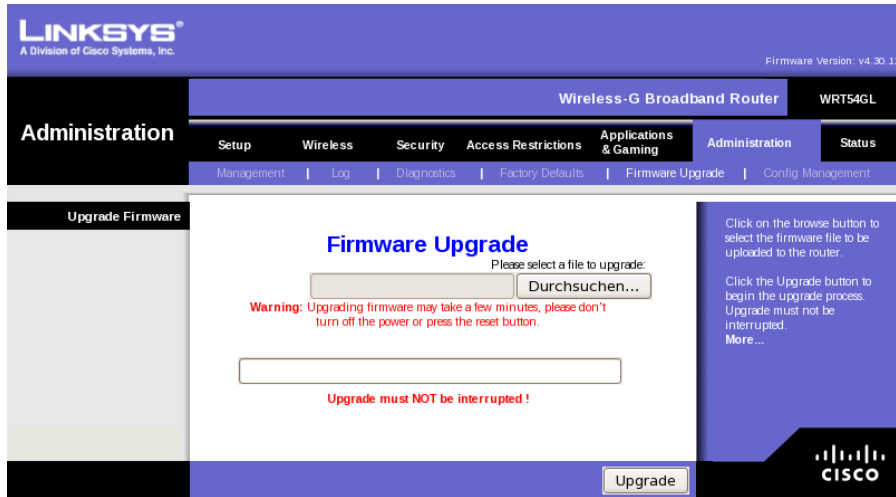
```
# wget
http://downloads.openwrt.org/kamikaze/8.09.2-RC2/brcm-2.4/openwrt-wrt54g-squashfs.bin
```

Diese Firmwarefile mit der Endung **.bin** beinhaltet die neue Firmware inkl. einiger Headerdaten, die für den Updatemechanismus der originalen Firmware benötigt werden. Möchte man OpenWrt selbst updaten, so z.B. von [Whiterussion](#) auf [Kamikaze](#) upgraden, greift man auf die **.trx** Version der Firmware zurück.

```
# wget
http://downloads.openwrt.org/kamikaze/8.09.2-RC2/brcm-2.4/openwrt-brcm-2.4-squashfs.trx
```

## Installation

Den Firmwareupdate stossen wir direkt über den Menüpunkt **Firmware Upgrade** unter **Administration**, erreichbar unter <http://192.168.1.1/Upgrade.asp> an.

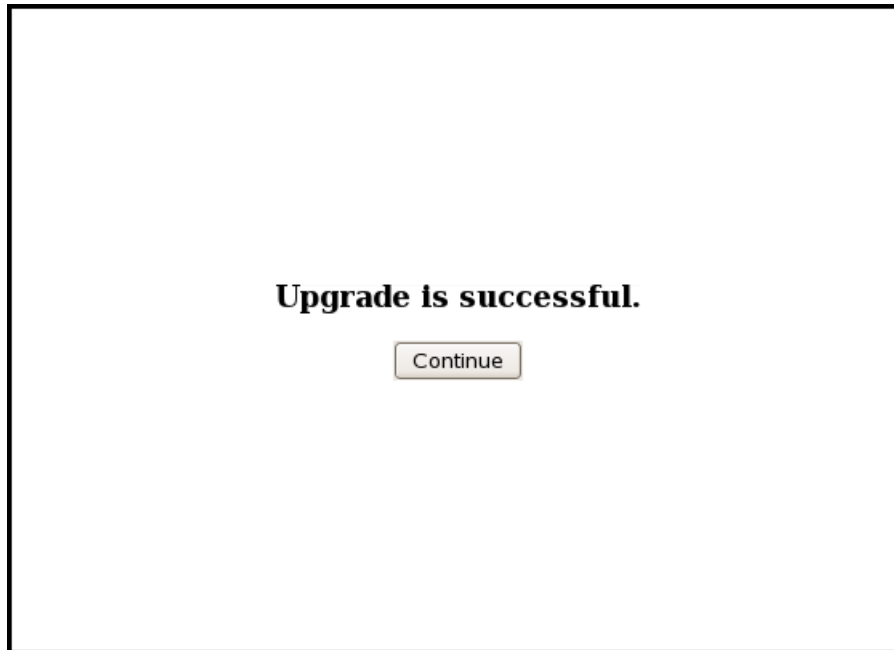


Nach Auswahl der Firmware-Datei klicken wir auf die Schaltfläche **Upgrade**.



Auf keinen Fall, die Stromversorgung des Router oder den Upgradevorgang unterbrechen! Als Ergebnis hätte man sonst einen *schmucken und wertlosen Briefbeschwerer*.

Am Ende des Firmware-Upgrades wir dies entsprechend mitgeteilt.



## Konfiguration

Nach dem erfolgreichen Upgrade können wir uns an unserem neue WLAN-Router unter der IP-Adresse **192.168.1.1** anmelden. Das initiale Rootpasswort lautet **admin**.



Dieses Passwort ändern wir gleich mal ab und tragen auch sonst unsere Konfigurationswünsche im **GUI<sup>1)</sup>** ein.

## Notfall

### Reset

Sollte bei der Konfiguration etwas schief gelaufen sein, kann man den Router zurücksetzen. Dazu drückt man den **Reset-Knopf** auf der Rückseite des Routers und führt einen **Power-Reset** (unterbrechen der Stromzuführung) durch. Den Reset-Taster halten wir solange gedrückt, bis das Blinken der LEDs aufgehört hat; i.d.R. sind dies ca. 10 bis 15 Sekunden.

## Firmwareflash mittels TFTP

Sollte es wider erwarten passieren, dass der Router weder via **SSH** noch ueber die Webconsolen ansprechbar ist, so hat man noch die Möglichkeit die Firmware via **TFTP** auf den Router zu spielen.

### TFTP-Server-Installation

Als erstes installieren wir uns den **TFTP-Client** via **yum**.

```
# yum install tftp
```

Das Paket **tftp** ist recht überschaubar, was uns ein **rpm -iql** auch entsprechend zeigt:

```
# rpm -iql tftp
Name           : tftp                Relocations: (not relocatable)
Version        : 0.49                Vendor: CentOS
Release        : 2.el5.centos        Build Date: Sa 26 Sep 2009
18:30:44 CEST
Install Date:  Do 29 Okt 2009 22:27:06 CET    Build Host:
builder16.centos.org
Group          : Applications/Internet    Source RPM:
tftp-0.49-2.el5.centos.src.rpm
Size           : 25236                License: BSD
Signature      : DSA/SHA1, Mo 28 Sep 2009 04:20:07 CEST, Key ID
a8a447dce8562897
Summary        : Der Client für das Trivial File Transfer Protocol (TFTP).
Description    :
The Trivial File Transfer Protocol (TFTP) is normally used only for
booting diskless workstations.  The tftp package provides the user
interface for TFTP, which allows users to transfer files to and from a
remote machine.  This program and TFTP provide very little security,
and should not be enabled unless it is expressly needed.
/usr/bin/tftp
/usr/share/man/man1/tftp.1.gz
```

### TFTP-Server-Konfiguration

An Konfiguration unseres TFTP-servers ist nicht's aufwändiges durchzuführen. Dei wichtigen Konfigurationsschritte beschränken sich auf die Konsole. Bevor wir von unserem Rechner den Datentransfer initiieren können, verpassen wir unserem Rechner noch eine feste IP-Adresse. Anschließend verbinden wir den Router direkt mit dem Rechner.

```
# ifconfig
eth0      Link encap:Ethernet  Hardware Adresse 00:11:25:2B:4A:66
          inet Adresse:192.168.1.100  Bcast:192.168.1.255
Maske:255.255.255.0
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
```

```
RX packets:10339 errors:0 dropped:0 overruns:0 frame:0
TX packets:10508 errors:0 dropped:0 overruns:0 carrier:0
Kollisionen:0 Sendewarteschlangenlänge:1000
RX bytes:1943762 (1.8 MiB) TX bytes:5934879 (5.6 MiB)
Basisadresse:0x8000 Speicher:c0220000-c0240000
```

Anschließend starten wir den **tftp-client** und geben mit der IP-adresse **192.168.1.1** die Hostadresse an, an die nachfolgend die daten übertragen werden sollen.

```
# tftp 192.168.1.1
```

Beim nun erscheinenden **Prompt** stoßen wir nun die Übertragung an:

```
# tftp 192.168.1.1
tftp> trace
Packet tracing on.
tftp> binary
tftp> put WRT54GL_v4.30.11_012_ETSI_EN_code.bin
```

Nun den Netzstecker schnell einstecken und der Router beginnt, automatisch die ihm angebotene Firmware selbstständig zu laden. Also einfach nur abwarten, bis der Router von selbst neu startet.

<sup>1)</sup>

**Grapfical User Interface**

From:  
<https://dokuwiki.nausch.org/> - **Linux - Wissensdatenbank**

Permanent link:  
<https://dokuwiki.nausch.org/doku.php/fun:openwrt>

Last update: **20.04.2018 09:06.**

