jjOpenWrt User Guide

Index

1. OpenWrt Buildroot – Installation

OpenWrt Buildroot is the buildsystem for the OpenWrt Linux distribution. OpenWrt Buildroot works on Linux, BSD or MacOSX operating system. A case-sensitive filesystem is required.

It is recommended that you use a Linux distribution (Debian), either a standalone installation or one running in a virtual environment (VMware or Qemu).

Cygwin(Windows) will not be supported because of the lack of case sensitiveness in the file system.

Outdated information for old Buildroot versions, old Linux variants is archived at: buildroot.exigence.old

For common problems, benchmarks, common optimizations see development

1.1 Prerequisites

To generate an installable OpenWrt firmware image file with a size of e.g. 8MB:

- ca. 200 MB of hard disk space for OpenWrt Buildroot
- ca. 300 MB of hard disk space for OpenWrt Buildroot + OpenWrt Feeds
- ca. 2.1 GB of hard disk space for source packages downloaded during build from OpenWrt Feeds
- ca. 3-4 GB of available hard disk space to build (i.e. cross-compile) OpenWrt and generate the firmware file
- ca. 1-4 GB of RAM to build Openwrt.(build x86's img need 4GB RAM)

1.2 Install procedure on Linux

1. Do everything as non-root user!
2. Issue all OpenWrt Buildroot commands in the <buildsystem root> directory, e.g. ~/openwrt/trunk/
3. Do not build in a directory that has spaces in its full path
4. Change Ownership other than root user,of the directory where you Download the OpenWrt (sudo chown -R user:user /openwrt/)

a) Install git , to conveniently download the OpenWrt source code, and build tools to do the cross-compilation process:

```
sudo apt-get update
sudo apt-get install git-core build-essential libssl-dev libncurses5-dev unzip
```

Some feeds might not available over git but only via subversion (short: svn) or mercurial. If you want to obtain their source-code, you need to install svn and mercurial as well:

```
sudo apt-get install subversion mercurial
```

- for information about the build tools see make and build-essential
- for information about git see git(7)
- for information about the subversion tool see svn and subversion documentation (multiple languages)

b) Download the OpenWrt bleeding edge (trunk Version) with git:

```
git clone git://git.openwrt.org/openwrt.git
```

this creates a directory 'openwrt', which is the OpenWrt Buildroot build-directory
the OpenWrt toolchain "OpenWrt Buildroot" is included

c) (optional) Download and install all available "feeds" (see OpenWrt Feeds for more options!):

```
cd openwrt
./scripts/feeds update -a
./scripts/feeds install -a
```

d) Make OpenWrt Buildroot check for missing packages on your build-system using one of the following commands:

```
make defconfig
make prereq
make menuconfig
```

There you will need to select what you want to compile.

e) Proceed with build (i.e. cross-compile the downloaded sources to binaries)
After the cross-compilation process the ''trunk''-directory contained 244,451 files with a total size of 3.2GiB!

1.3 Corresponding packages Installations

Here is the package name for each prerequisite separated for different GNU/Linux and Unix like distributions.

- **Debian 7 Wheezy:**

```
apt-get install libncurses5-dev zlib1g-dev gawk
```

- **Fedora 20 – 64Bit**(Maybe also for lower versions, some packages seem to be missing above):

```
yum install -y subversion binutils bzip2 gcc gcc-c++ gawk gettext flex ncurses-devel
zlib-devel make patch unzip perl-ExtUtils-MakeMaker glibc glibc-devel glibc-static
quilt ncurses-lib sed sdcc intltool sharutils bison wget
```

- **openSUSE 13.2**

```
zypper install asciidoc bash bc binutils bzip2 fastjar flex git-core gcc-c++ gcc
util-linux gawk gtk2-devel intltool jikes zlib-devel mercurial make genisoimage
ncurses-devel libopenssl-devel patch perl-ExtUtils-MakeMaker python-devel rsync ruby
sdcc unzip wget gettext-tools libxslt-tools zlib-devel subversion
```

- **Ubuntu 12.04LTS:**

```
sudo apt-get install build-essential subversion git-core libncurses5-dev zlib1g-dev
gawk flex quilt libssl-dev xsltproc libxml-parser-perl mercurial bzr ecj cvs unzip
```

- **Ubuntu 64bit:**

```
sudo apt-get install build-essential subversion libncurses5-dev zlib1g-dev gawk
gcc-multilib flex git-core gettext libssl-dev
```

1.4 Downloading Sources

15.05 branch (Chaos Calmer)

Main repository

```
git clone git://git.openwrt.org/15.05/openwrt.git
```

Packages feed

```
git clone git://git.openwrt.org/15.05/packages.git
```

2. jjPlus OpenWrt for 802.11ac

The openWrt buildroot system of jjPlus version is based on 15.05 branch version (Chaos Calmer), which is cooked in Ubuntu 12.04LTS 64-bit OS environment. This system has included Chaos Calmer basic package and jjplus patches, which is for jjPlus platform modifications only.

2.1 Setup Ubuntu Build Environment

Require to install each prerequisite of package on uBuntu 12.04LTS 64-bit GNU/Linux distribution.

Ubuntu 12.04LTS 64-bit:

```
sudo apt-get install build-essential subversion libncurses5-dev zlib1g-dev gawk
gcc-multilib flex git-core gettext libssl-dev quilt xsltproc libxml-parser-perl mercurial
bzr ecj cvs unzip
```

2.2 Install jjPlus Buildroot System

There are two package files “jjPlus-MIPS-openWrt_rYYYYY_xxxxxxxx.tar.bz2” and “openWrt_rYYYYY-extra-tarballs.tar”.

- “jjPlus-MIPS-openWrt_rYYYYY_xxxxxxxx” file is buildroot system source base.
- “openWrt_rYYYYY-extra-tarballs” file is extra source tarballs.

Please follow the steps to install jjPlus Buildroot System.

a) Make a folder name in ubuntu $HOME path

```
$ mkdir –p ~/openwrt
```

b) Uncompress two package files into the created folder.

```
$ tar –xvf jjPlus-MIPS-openWrt_rYYYYY_xxxxxxxx.tar.bz2 –C ~/openwrt
$ tar –xvf openWrt_rYYYYY-extra-tarballs.tar –C ~/openwrt
```

c) Copy jjPlus platform config as buildroot system configuration file.

```
$ cd ~/openwrt
$ cp jwapXXX_defconfig .config
```

d) Build jjPlus platform images

```
$ make V=s
```

Note: If you got any inappropriate problem in your PC environment, please use the following steps to fix it.

```
$ cd ~/openwrt
$ make distclean
$ ./scripts/feeds update -a
$ ./scripts/feeds install -a
$ cp jwapXXX_defconfig .config
$ make V=s menuconfig    or    make V=s
```

2.3 jjPlus Platform Firmware Update

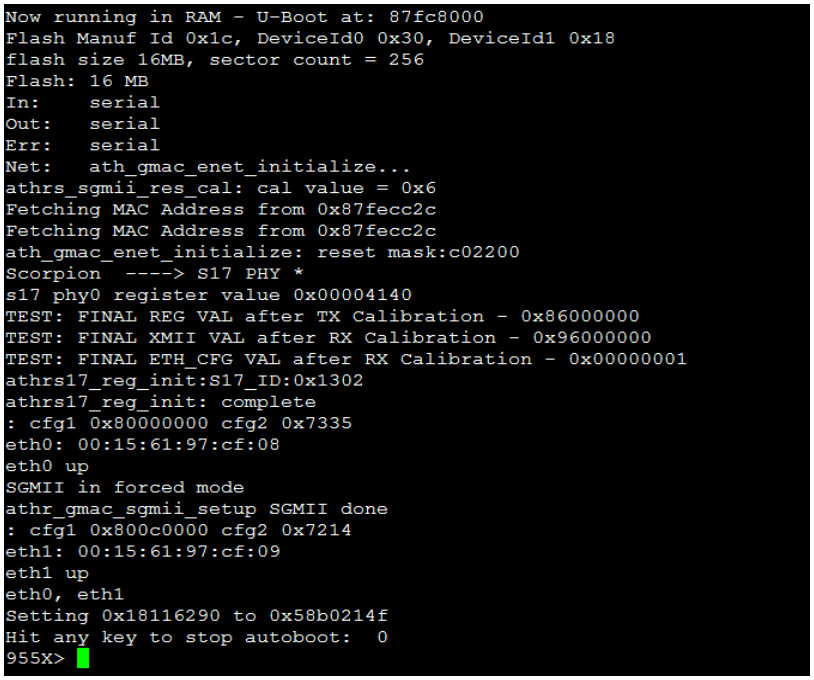
Three images will be generated, “openwrt-jwapXXX-kernel.bin”, “openwrt-jwapXXX-rootfs-squashfs.bin” and “openwrt-jwapXXX-squashfs-sysupgrade.bin”.

- openwrt-jwapXXX-kernel.bin: The lzma-compressed kernel image.
- openwrt-jwapXXX-rootfs-squashfs.bin: The root file system of squash format image.
- openwrt-jwapXXX-squashfs-sysupgrade.bin: This image is included kernel and root file system.

Firmware Update Procedures:

a) Setup a TFTP server on your PC, and copy the firmware "openwrt-jwapXXX-**squashfs-sysupgrade.bin”** in the directory of TFTP server program.

b) Enter boot-loader mode from the serial console; just press any key on keyboard when the boot-loader displays the countdown screen.



c) Set the TFTP server IP as PC's IP address.

```
> setenv serverip "PC's IP Address"
```

d) Update the firmware image by the following commands

```
> tftp 0x80060000 openwrt-jwapXXX-squashfs-sysupgrade.bin
> erase 0x9f050000 +0xfa0000 && cp.b $fileaddr 0x9f050000 $filesize
```

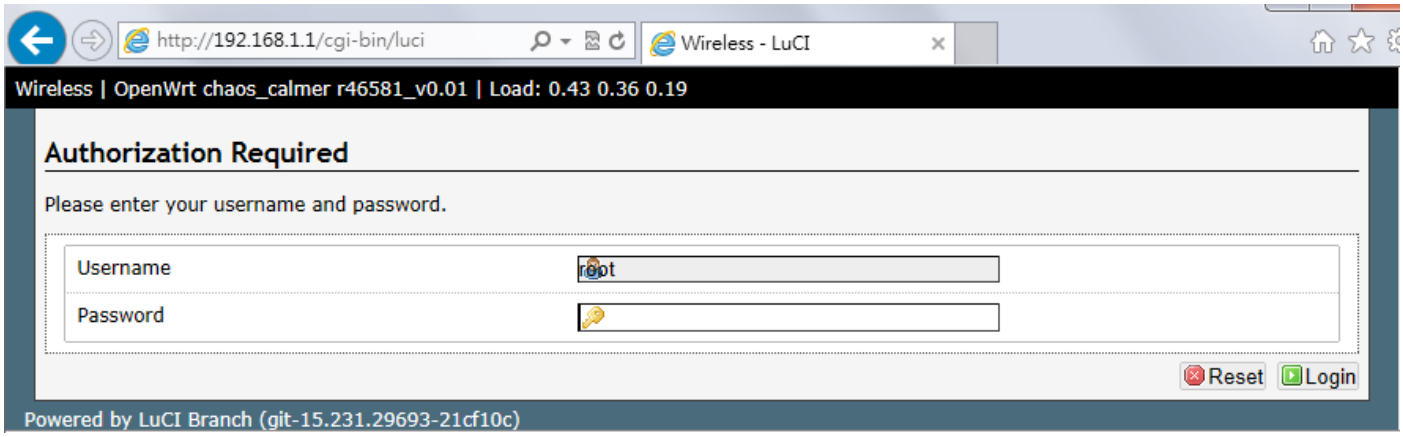
```
955X> print bootargs
bootargs=console=ttyS0,115200 root=31:02 rootfstype=jffs2 init=/sbin/init mtdpar
ts=ath-nor0:256k(u-boot),64k(u-boot-env),6336k(rootfs),1408k(uImage),8256k(mib0)
,64k(ART) board=jwap230
955X> tftp 0x80060000 openwrt-jwap230-squashfs-sysupgrade.bin
Trying eth0
eth0 link down
FAIL
Trying eth1
dup 1 speed 1000
Using eth1 device
TFTP from server 192.168.1.10; our IP address is 192.168.1.1
Filename 'openwrt-jwap230-squashfs-sysupgrade.bin'.
Load address: 0x80060000
Loading: ###################################################################
         ###################################################################
         ###################################################################
         ###################################################################
         ###################################################################
         ###################################################################
         ###################################################################
         ###################################################################
         ############
done
Bytes transferred = 16034238 (f4a9be hex)
955X> erase 0x9f050000 +0xfa0000&&cp.b $fileaddr 0x9f050000 $filesize
Erasing flash...
First 0x5 last 0xfe sector size 0x10000                                  254
Erased 250 sectors
Copy to Flash... write addr: 9f050000
done
955X>
```

e) Reboot the system when update firmware is done.

```
> reset
```

3. Software Features of jjOpenWrt

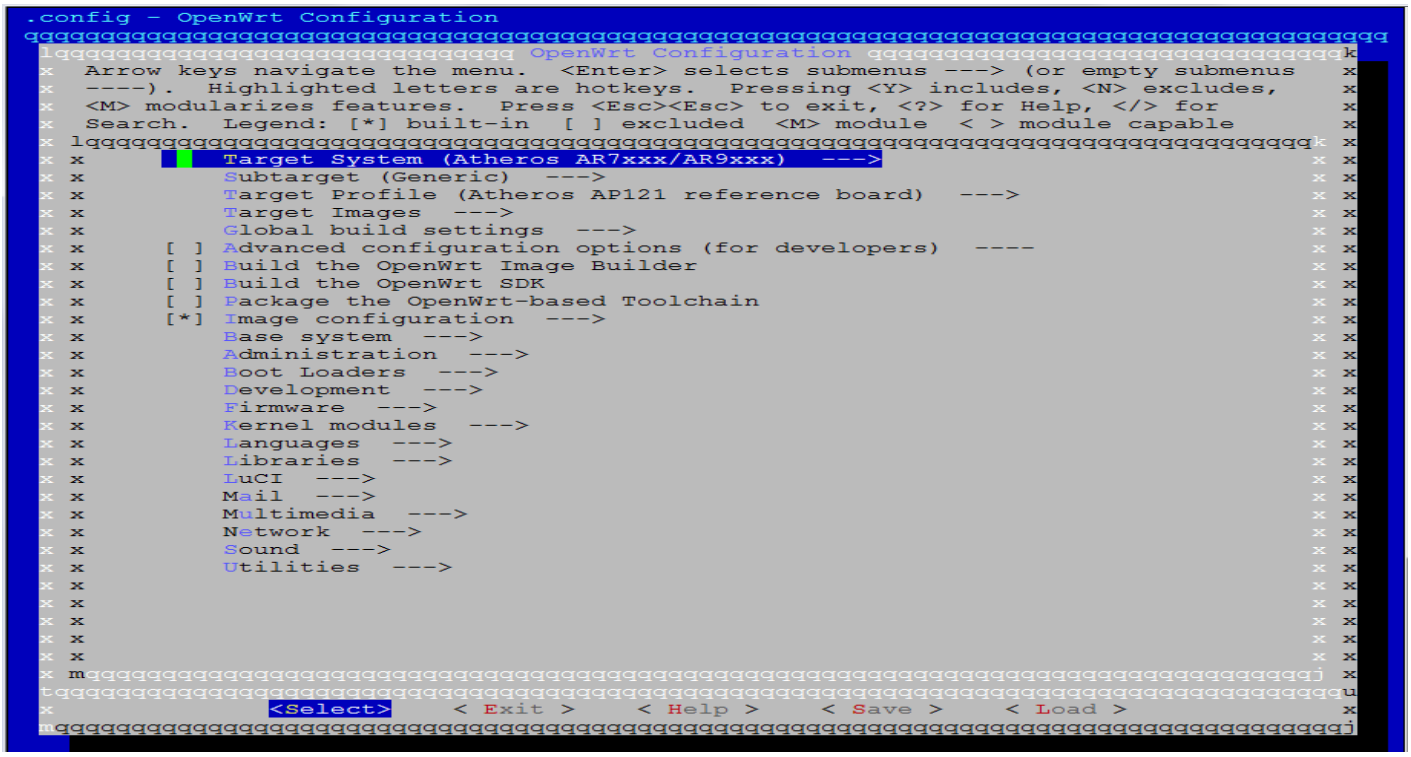
Launch a web browser and enter the default IP address “192.168.1.1” of the jjOpenWrt system.
The default account information is “Username: root / Password: admin”.



3.1 jjOpenWrt Features List

If you want to customize jjOpenWrt features/functions, you can use the “make menuconfig” command to customize it.

```
$ cd ~/openwrt
$ make V=s menuconfig
```



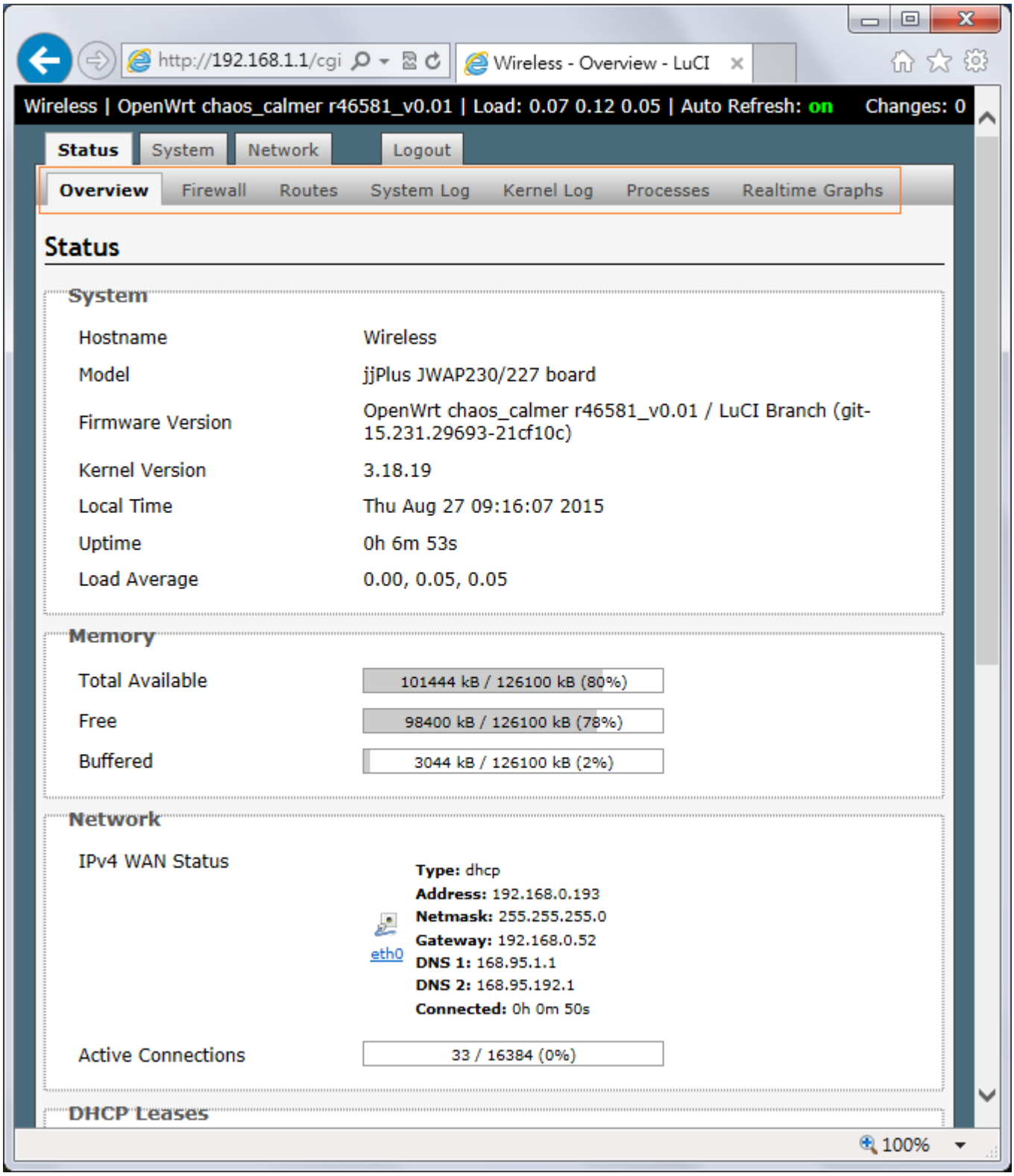
jjOpenWrt Features / Functions List:

Tab Name	Features	Sub-Features	Items Name	Sub-Items	Functions
Status					
	Overview				
	Firewall				
	Routes				
	System Log				
	Kernel Log				
	Processes				
	Realtime Graphs				
		Load			
		Traffic			
		Wireless			
		Connections			
System					
	System				
		System Properties			
		Time Sync.			
	Administration				
		Router Password			
		SSH Access			
	Startup				
	Scheduled Tasks				
	LED Configuration				
	Backup / Flash Firmware				
		Backup / Restore			
		Flash new firmware image			
	Reboot				
Network					
	Interfaces				
		Interface Configuration			
	Wifi				
		Wireless Configuration			

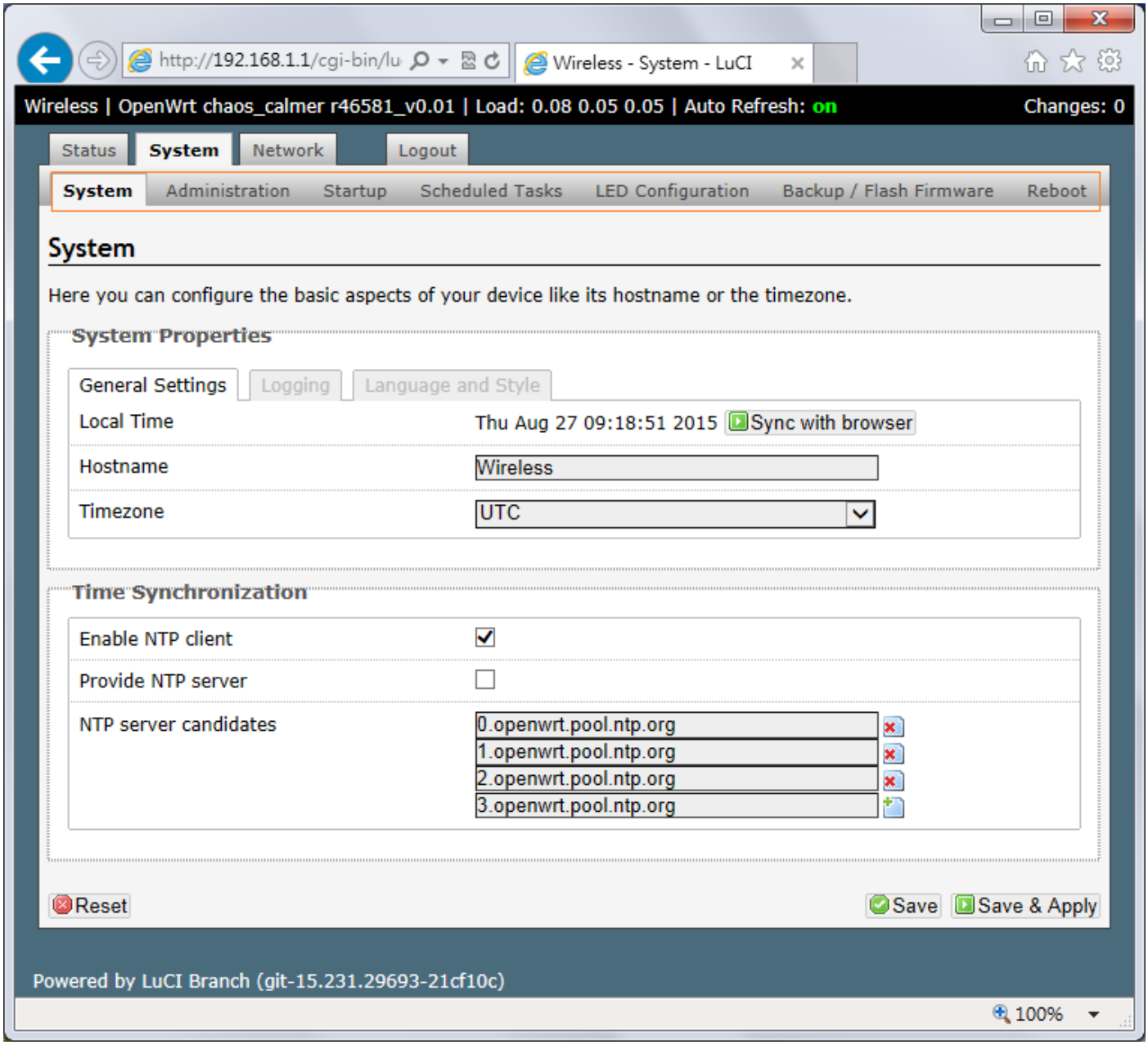
			Device Configuration		
				General Setup	
					Wireless network --- [Disabled / Enabled]
					Operating frequency --- [Legacy / N / AC]
					Transmit Power
				Advanced Settings	
					Country Code
					Distance Optimization
					Fragmentation Threshold
					RTS/CTS Threshold
			Interface Configuration		
				General Setup	
					ESSID
					Mode --- [Access Point / Client / Ad-Hoc / 802.11s / Pesudo Ad-Hoc / Monitor / Access Point (WDS) / Client (WDS)]
					Hide ESSID
					WMM Mode
				Wireless Security	
					Encryption --- [No Encry. / WEP Open System / WEP Shared Key / WPA-PSK / WPA2-PSK / WPA-PSK_WPA2-PSK Mixed]
				MAC-Filter	
					MAC-Address Filter --- [disable / Allow listed only / Allow all except listed]
		Associated Stations			
	Switch				
		Switch Configuration			
		VLANs Configuration			
	DHCP and DNS				
		Server Settings			
		Active DHCP Leases			
		Active DHCPv6 Leases			
		Static Leases			

	Hostnames				
	Static Routes				
	Firewall				
		General Settings			
		Port Forwards			
		Traffic Rules			
		Custom Rules			
	Diagnostics				
	QoS				
		QoS Configuration			
		Classification Rules			
Logout					

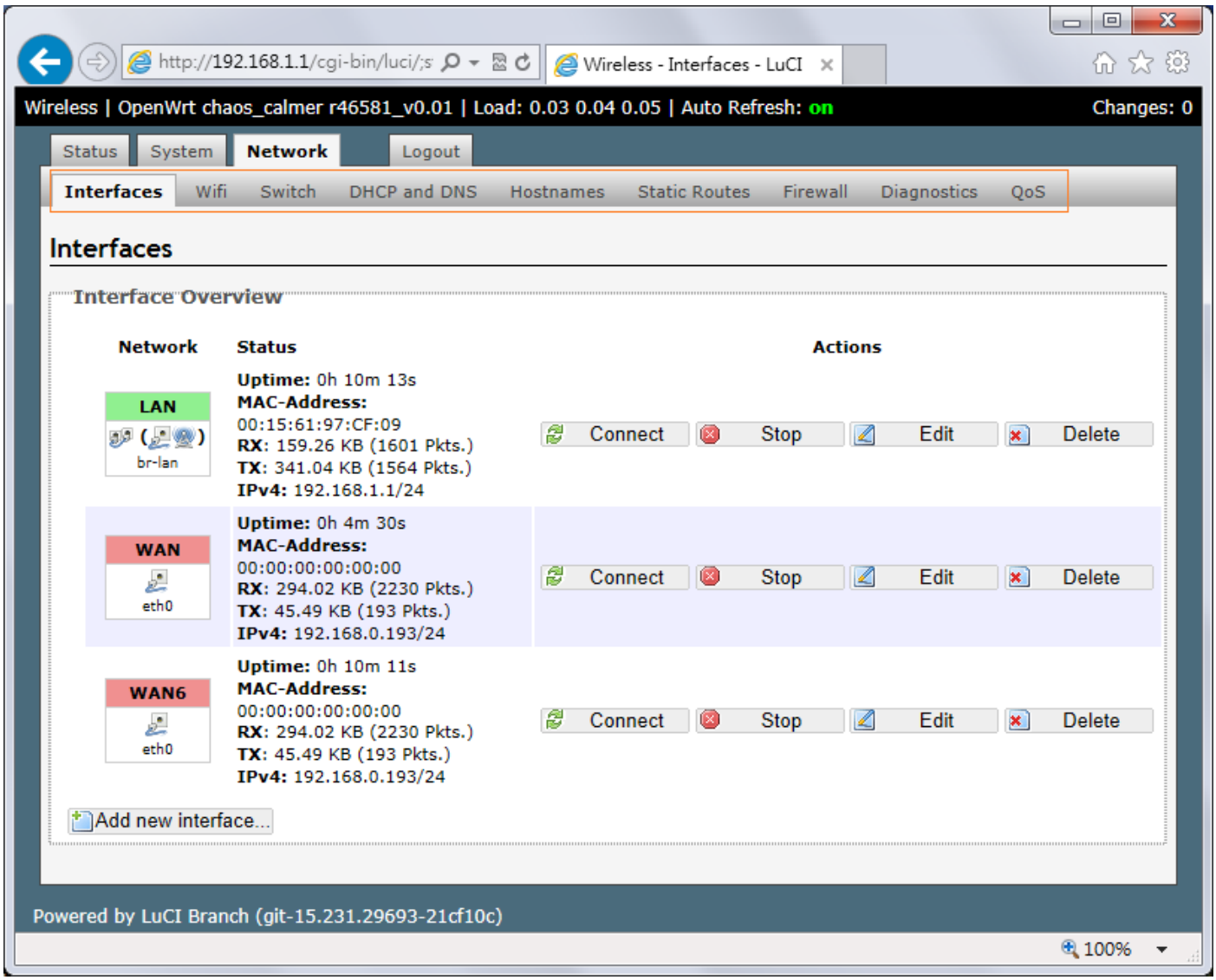
3.2 Status Tab Page



3.3 System Tab Page



3.4 Network Tab Page



3.4.1 WiFi Configuration Page