

This page is relevant only to ARM-based Broadcom routers such as the Netgear R7000 (and others compatible with the K3-AC-ARM build). The current state of Optware on these routers is that it is NOT OFFICIALLY SUPPORTED, meaning that neither BS nor Kong builds currently come with a working ipkg/opkg and there are NO REPOSITORIES yet that are formally confirmed as compatible with DD-WRT on K3-AC-ARM.

Contents

- 1 Current state of play
 - ◆ 1.1 History
 - ◆ 1.2 Developments
 - ◆ 1.3 Warning
- 2 Current build 277xx - August 2015
 - ◆ 2.1 Using the OpenWRT "chaos calmer" bcm53xx repository
 - ◇ 2.1.1 Prerequisites
 - ◇ 2.1.2 Bootstrap script
 - 2.1.2.1 For brave people [who examined the file first]
 - 2.1.2.2 For not so brave people
 - ◇ 2.1.3 Testing it out
 - ◇ 2.1.4 Package status of "chaos calmer" packages on DD-WRT
 - 2.1.4.1 Packages working
 - 2.1.4.2 Packages not fully working OOTB or requiring modifications
 - 2.1.4.3 Packages wishlisted (desired, not present as OpenWRT packages)
 - 2.1.4.4 To do list
- 3 OLD BUILDS (DEPRECATED)
 - ◆ 3.1 OpenWRT imx6 package repository
 - ◆ 3.2 Instructions
 - ◇ 3.2.1 Preparing for optware install
 - 3.2.1.1 Using USB storage
 - 3.2.1.2 Using JFFS storage
 - ◇ 3.2.2 Installing optware
 - 3.2.2.1 Use ipkg to install opkg
 - 3.2.2.2 Create opkg config file
 - 3.2.2.3 Update opkg lists
 - 3.2.2.4 Install a package
 - 3.2.2.5 Simplify opkg command line with an alias
 - ◆ 3.3 Package status on DD-WRT
 - ◇ 3.3.1 Packages working out of the box
 - ◇ 3.3.2 Packages working with minor changes
 - ◇ 3.3.3 Packages not working in DD-WRT
 - ◇ 3.3.4 Packages wishlist
 - ◆ 3.4 Build your own packages
 - ◇ 3.4.1 Fetch OpenWRT buildroot environment
 - ◇ 3.4.2 Build-time notes 5 Feb 2014

- ◆ 3.5 BUILDS OLDER THAN 23460M
 - ◇ 3.5.1 Why didn't it just work?
 - ◇ 3.5.2 mbwe-bluering repository [DEPRECATED]
 - 3.5.2.1 Packages not working

Current state of play

History

Early Kong builds shipped with the mbwe-bluering Optware repository from nslu2 (which didn't work too well) and were then by use of OpenWRT's imx6 repository. This hit a brick wall when the uClibc library was dropped by DD-WRT in favour of MUSL, rendering all previous packages that had been compiled against uClibc incompatible. A bcm53xx-musl repository was for a while maintained by Kong, but this now appears to have been removed from current builds and from his website.

Developments

As at August 2015, no current build supports Optware out of the box. However, meanwhile, back in June, the folks over at OpenWRT have themselves migrated to MUSL, effective as of the current "trunk" release. So, in theory at least, there's a chance that OpenWRT packages are again compatible (to a greater or lesser extent) with DD-WRT.

Warning

Use of OpenWRT (or any other third party) packages with DD-WRT may work, but it is not supported by the DD-WRT project. Proceed at your own risk.

Current build 277xx - August 2015

Using the OpenWRT "chaos calmer" bcm53xx repository

These instructions were put together on build 27715, deriving input from a number of sources including magick777's bootstrap script, Kong's bootstrap script, current OpenWRT documentation, etc.

Prerequisites

We assume that you have your /opt partition created, mounted, and available to your DD-WRT router. We don't care whether you've done that by preparing a USB stick and automounting it, clearing JFFS and bind-mounting it, setting up CIFS automount or whatever.

Bootstrap script

To make life slightly easier, magick777 has updated his previous [bootstrap script](#) to install opkg and enable the OpenWRT bcm53xx repository under DD-WRT. To give credit where it's due, the raw installation of opkg from the ipk is drawn from Kong's bootstrap script as was previously used with his repository on K3-AC-ARM builds.

For brave people [who examined the file first]

```
wget -q -O- http://debian.keithdunnett.net/ddwrt/optware\_setup | sh
```

Note: you should never trust a script like this without examining it. Go visit that file in your web browser and satisfy yourself it doesn't contain anything malicious before you run the above command. It doesn't, but it's a good practice to get into if you're going to start piping scripts off the web into a shell interpreter.

For not so brave people

```
mkdir -p /opt/usr/bin  
wget -q -O- http://debian.keithdunnett.net/ddwrt/optware\_setup > /opt/usr/bin/optware_setup  
vi /opt/usr/bin/optware_setup #or use your favourite editor, just have a look at it anyway  
chmod 700 /opt/usr/bin/optware_setup  
/opt/usr/bin/optware_setup
```

Testing it out

Try 'opkg update', then 'opkg install nano'. You'll get a few (minor) errors complaining about a failed dependency on libc, but don't worry - DD-WRT has libc built in and doesn't need a copy installed under /opt. GNU nano should work just fine, as will a number of other simple apps.

Package status of "chaos calmer" packages on DD-WRT

This section refers to the compatibility of packages from current OpenWRT trunk (chaos calmer) with current DD-WRT releases on ARM routers, such as the R7000.

Packages working

- nano - seems to be fine
- rsync - seems to be fine
- wget - seems to be fine

Packages not fully working OOTB or requiring modifications

- openssh-server - not working OOTB, requires specifying LD_LIBRARY_PATH=/opt/usr/lib (and not at the end of LD_LIBRARY_PATH, we need to prefer libcrypto.so.0 from /opt/usr/lib over the

built-in libcrypto). We can do this by default in root's .profile on startup, but this may break other packages and we may need to leave the defaults as they are and modify on a case by case basis. As of now, we do not yet have full support for startup scripts under /opt/etc/init.d (as is done in OpenWRT and as was done in OTRW2); that can hopefully be implemented later provided that a worthwhile number of packages are working.

Packages wishlisted (desired, not present as OpenWRT packages)

- rdiff-backup
- softether
- unionfs-fuse

To do list

- Ideally, support running native OpenWRT startup scripts automatically (or, indeed, at all) from /opt/etc/init.d
 - ◆ May require a script to 'optify' the startup scripts and make them work with the packages as installed
 - ◇ Or bolt-on an overlay filesystem to DD-WRT, pivot root to a RW union, de-optify the installation?
 - ◆ May require additional function scripts or compatibility tweaks
 - ◇ Requires a replacement of /etc/rc.common
 - ◇ May need to support OpenWRT's procd / ubusd architecture (which would need to be optified)
 - Redundant, can we use DD-WRT's existing architecture instead?
- Alternatively, find another way to produce DD-WRT compatible startup scripts for OpenWRT packages
 - ◆ Currently, individual command lines can be put manually into rc_startup for execution on boot
 - ◆ Our optware_boottime script could iterate over a directory of startup files
 - ◇ If unable to make native OpenWRT startup scripts work with DD-WRT, replace with working default scripts
 - ◇ Run each working startup script that is detected from optware_boottime on startup

OLD BUILDS (DEPRECATED)

OpenWRT imx6 package repository

Quoting <Kong>, "builds >23453 will use openwrt packages". The repository now in use is the OpenWRT imx6 repository, <http://downloads.openwrt.org/snapshots/trunk/imx6/packages/>

Instructions

Preparing for optware install

Using USB storage

Prepare a usb flash disk with 3 partitions 512MB swap + 2 partitions ext2/ext3 or ext4. Enable USB Storage and mount one partition to /jffs, the other to /opt, by using the uuids which are shown in disk info. **Note:** if you use this method, you do NOT need to enable JFFS2 support in the GUI. That's for putting the JFFS partition on the internal flash.

Using JFFS storage

If you don't have USB storage attached to your router you can resort to the time-honoured method of enabling JFFS2 (from the DD-WRT GUI), which will give you ~90MB of storage under /jffs. You can then

```
mkdir /jffs/opt
mount --bind /jffs/opt /opt
```

which will let you install smaller packages using JFFS alone. If you do this, you'll need to re-create the bind mount on startup.

Installing optware

Use ipkg to install opkg

```
ipkg update
ipkg install opkg
```

Create opkg config file

```
echo "src openwrtarm http://downloads.openwrt.org/snapshots/trunk/imx6/packages/" >/jffs/etc/opkg.conf
echo "dest root /opt" >> /jffs/etc/opkg.conf
echo "dest ram /tmp" >> /jffs/etc/opkg.conf
echo "lists_dir ext /jffs/var/opkg-lists" >> /jffs/etc/opkg.conf
echo "option overlay_root /jffs/overlay" >> /jffs/etc/opkg.conf
```

Update opkg lists

```
opkg -f /jffs/etc/opkg.conf update
```

Install a package

```
opkg -f /jffs/etc/opkg.conf --force-depends install <package>
```

Simplify opkg command line with an alias

```
alias opkg="opkg -f /jffs/etc/opkg.conf --force-depends"
```

will add those switches for you by default, so you can just use "opkg list", "opkg install <what>". If you want to make this permanent, you can add

```
echo 'alias opkg="opkg -f /jffs/etc/opkg.conf --force-depends"' >>/tmp/root/.profile
```

to your startup commands.

Package status on DD-WRT

Please add to / edit this section as appropriate.

Packages working out of the box

- **coreutils-sha512sum** - seems to install and run fine
- **nano** - seems to install and run fine
- **ncftp** - sometimes refuses to work interactively, logs in OK but promptly exits. Running it by installing "screen" and using "screen ncftp" works fine. So does logging out and logging in again, so moved to apparently working.
- **rsync** - seems to install and run fine
- **screen** - seems to install and run fine
- **transmission** - reported working by forum users: Magnetron1.1, tibismart

Packages working with minor changes

- **mrtg** - installs OK, but needs paths optifying in mrtg, indexmaker, cfgmaker scripts, also see note on perl, below. Use cfgmaker to produce config file in /jffs/etc. Customise to suit. Crontab accordingly, you'll need to specify PERL5LIB and LD_LIBRARY_PATH in order for the rateup executable to work. Obviously you'll also need snmpd enabled, and lighttpd set up.
- **perl** - doesn't work OOTB, try PERL5LIB=/opt/usr/lib/perl5/5.10/; export PERL5LIB. You can make this more permanent by adding "echo "export PERL5LIB=/opt/usr/lib/perl5/5.10/" >>/tmp/root/.profile" to your startup commands.
- **rdiff-backup** - doesn't work OOTB thanks to a hard-coded path to python in the rdiff-backup script. However, you can issue a sed -i "s#!/#!/opt/" /opt/usr/bin/rdiff-backup to fix the script, and rdiff-backup will then run correctly. Note that it has no onboard help and no default verbosity, so issue a -v5 as part of your command to see what is going on.
- **sshfs** - works, however you may need to run it in debug mode in order to accept the remote host's key, and you will need an "insmod fuse" first to load the requisite kernel module. Note that mounting very large filesystems (>4TB, I think) provokes an error from 'df': Value too large for defined data type; a

3.6TB filesystem is handled correctly but a 6.8TB filesystem is not. This does not seem to affect listing or accessing files, merely produces a bug in df output. [Magick777](#) 04:47, 2 February 2014 (CET)

Packages not working in DD-WRT

- **kmod-*** - kernel modules for OpenWRT, not for our DD-WRT kernel.
- **nfs-kernel-server** - depends on kernel modules not presently available in DD-WRT. Might yet be persuaded to work if we can have them added to DD-WRT or create a custom repository with compatible kernel modules.
- **nfs-server** - depends on OpenWRT basefiles / init scripts not presently available in DD-WRT.
- **aoetools** - requires kernel module aoe.ko
- **tor** - silently exits with no useful debug info. On disabling RunAsDaemon, produces a segfault.
- **iptables** - not compiled to find libraries under /opt, also needs kernel modules to be useful

Packages wishlist

Here we note stuff that is not available/supported via OpenWRT packages which there might be a case to add.

- **kmod-*** - would require modules compiled for the current DD-WRT kernel version installed to /opt and ideally a mechanism for loading them on startup (though this last could be done manually via the user.)
- **unionfs-fuse** - we have fuse support already, should not be too hard to add with a working SDK
- **webui-aria2** - we have aria2, just not the webif for it

Build your own packages

Disclaimer: this section is a work-in-progress.

Fetch OpenWRT buildroot environment

If you need to re-compile a package, or add a new one, you can download the OpenWRT source code and build it yourself. Read the OpenWRT documentation for this, starting with <http://http://wiki.dd-wrt.com/wiki.openwrt.org/doc/howto/buildroot.exigence>

Build-time notes 5 Feb 2014

- The build broke once or twice thanks to temporary network failures (e.g. git.kernel.org unreachable), resolved by simply trying again later. Suggest running with make V=s from the outset, you'll need debug info.

Optware_on_K3-AC-ARM

- Errors from 'cp' (flag -p not supported) are caused by gcp taking precedence over cp on your host system. Removing gcp should solve the problem.
- Compile of postgresql breaks for our architecture, see <https://forum.openwrt.org/viewtopic.php?id=45866> for a simple fix
- Problems with glib-genmarshal when compiling glibc under OpenSuse. Wasn't able to diagnose this and switched to Ubuntu 32-bit instead.
- Problems compiling ap51-flash, disabled the package.

BUILDS OLDER THAN 23460M

Long story short: do not attempt to use Optware in builds older than 23460M.

Why didn't it just work?

The [Optware Project](#) was not designed for use with ARM processors (most builds are for mipsel architecture), and nor was it intended for DD-WRT. It has been ported to a number of DD-WRT platforms, but with varying degrees of success. Also, the project has become somewhat dated, meaning that it relies on an old version of uClibc; this causes issues with the current DD-WRT codebase. So, we need an Optware feed compiled for modern DD-WRT, and then for that to work on the K3-AC-ARM routers we need it compiled for ARM architecture. There is presently no Optware feed that meets that descriptor; the mbwe-bluering repository is built for ARM, but it's not built for DD-WRT on kernel 3.x.

mbwe-bluering repository [DEPRECATED]

This page tracks the success (or failure) of using the Optware mbwe-bluering repository on newer 802.11ac routers with ARM architecture, running Kong's K3-AC-ARM builds (r23250 and later). Whilst this repository is supposedly compatible with an ARM CPU and uClibc, most of the binaries don't work on DD-WRT.

Packages not working

Most of them, by the look of it. Initially I got a bunch of the following errors:

```
hello: can't resolve symbol '__register_frame_info'
nano: can't resolve symbol '__register_frame_info'
ncftp: can't load library 'libresolv.so.0'
perl: can't resolve symbol '__register_frame_info'
whois: can't resolve symbol '__register_frame_info'
```

so I duly installed uClibc-opt; this got rid of the above errors but it still didn't persuade anything useful to work. Most things now exit silently, neither complaining nor producing any output.