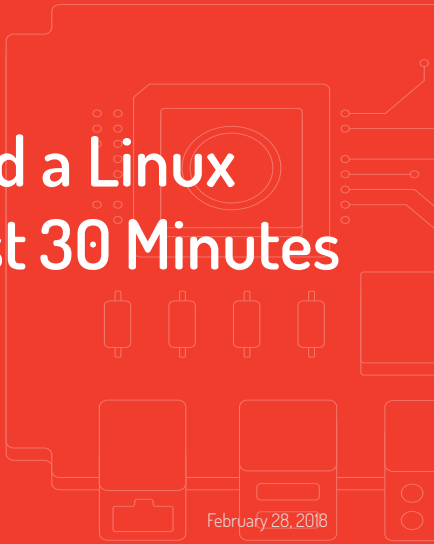
EMBEDDED WORLD 2018

OpenWrt 101: How to Build a Linux Embedded System in Just 30 Minutes

Luka Perkov, Managing Director, Sartura

Cesare Garlati, Chief Security Strategist, prpl Foundation

sartura



February 28, 2018

prpl Foundation

- An open source, community-driven, collaborative, non-profit foundation
- Enabling the security and interoperability of embedded devices for the IoT and smart society of the future
- **prpl Foundation & member companies**
  - 40+ member organizations
  - 200+ active participants

prpl Foundation members (partial member list)

- **prpl Foundation & OpenWrt**
  - Regular OpenWrt Summit hosts & sponsors:
    - Prague, Czech Republic, 2017
    - Berlin, Germany, 2016
    - Dublin, Ireland, 2015
- **prpl Foundation & Sartura**
  - Long-term collaboration
  - TR-069 integration project with member companies

Sartura

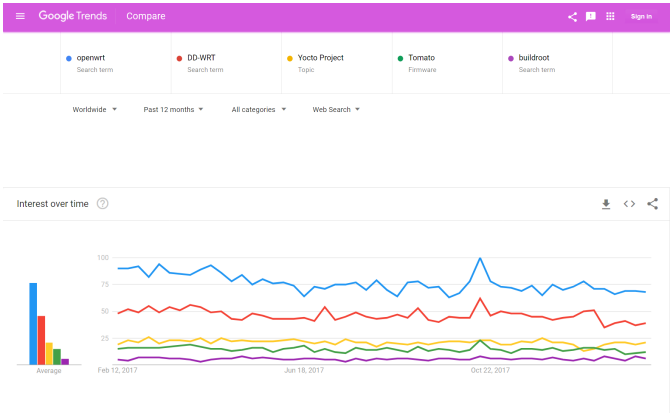
Services:

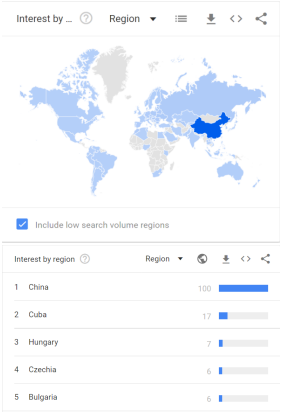
Development
Integration
Consulting
Training

Partners & Customers:

Embedded Device Manufacturers
Telecoms
Semiconductor Manufacturers
Open Source

OpenWrt Heat Map





Note: As of February 8, 2018 - This comparison contains both Search terms and Topics, which are measured differently.

What is OpenWrt?

- Linux distribution optimized for small footprint, portability and configurability
- Minimum requirements 4MB FLASH / 32MB RAM
- Open source licensed under GPLv2
- Broadest ISA support – 15+ SoCs
- Broadest hardware support - 680+ devices

- Used by many OEMs – Netgear, Linksys, TP-Link, D-Link
- Vibrant global community – non-profit & commercial
- Not the easiest to install and use
- Sometime tricky to configure
- Documentation not always available or up-to-date

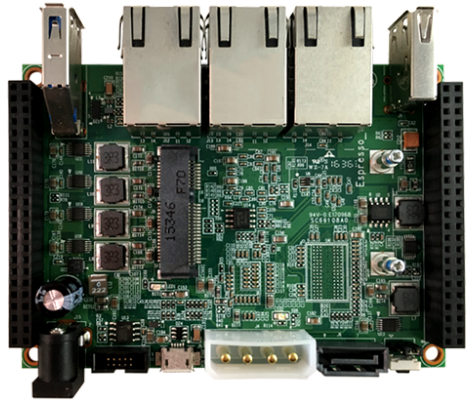
Comparative Table - OpenWrt, Buildroot, Yocto

Component	OpenWrt	Buildroot	Yocto
Menu config	Kconfig	Kconfig	Kconfig
C libraries	uClibc, glibc, musl	glibc, uClibc-ng, musl	EGLIBC
File Systems	OverlayFS, tmpfs, SquashFS, JFFS2, UBIFS, ext*	Cramfs, JFFS2, romfs, cloop, ISO 9660, cpio, UBI, UBIFS, SquashFS, ext*	Btrfs, cpio*, cramfs, ELF, ext*, ISO, JFFS2, multiubi, SquashFS, UBI, UBIFS
Root Necessary	Yes	No	Yes
Init Systems	Procd, BusyBox	systemV, BusyBox, systemd	SysVinit, systemd
Package Manager	opkg	none	smart

Hardware Requirements

Marvell's ESPRESSObin board

SoC	Marvell Armada 3700LP (88F3720) dual core ARM Cortex A53 processor up to 1.2GHz
System Memory	1 GB DDR3 or optional 2GB DDR3
Storage	1x SATA interface, 1x micro SD card slot (optional 4GB EMMC footprint)
Network Connectivity	1x Topaz Networking Switch, 2x GbE Ethernet LAN, 1x Ethernet WAN, 1x MiniPCIe slot for Wireless/BLE peripherals
USB	1x USB 3.0, 1x USB 2.0, 1x micro USB port
Expansion	2x 46-pin GPIO headers for accessories and shields with I2C, GPIOs, PWM, UART, SPI, MMC, etc.
Power Consumption	Less than 1W thermal dissipation at 1 GHz



Setting up Build Environment

- Docker image
  - Based on Ubuntu 16.04
  - Contains pre-compiled OpenWrt Build System environment
  - Size around 12 GB
- Pulling Docker image:

```
$ docker pull sartura/build_openwrt_ubuntu_16.04:espressobin
```

- Preparing local folder:

```
$ mkdir ~/espressobin
$ chmod 777 ~/espressobin
```

- Running Docker image:

```
$ docker run -it -v /home/<username>/espressobin:/opt/espressobin --name
    espressobin sartura/build_openwrt_ubuntu_16.04:espressobin
```

- Built images located in */home/build/openwrt-dd/bin/mvebu64/*

Modifying & rebuilding images

```
$ make menuconfig
```

- Configuring Target System and Target Profile
- Configuring Target Images
- Configuring additional features and packages (e.g. Web UI - LuCI)

OpenWrt Configuration

Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

```
    Target System (Marvell 64b Boards)  --->
    Target Profile (ESPRESSObin (Marvell Armada 3700 Community Board))  --->
    Target Images  --->
    Global build settings  --->
[*] Advanced configuration options (for developers)  --->
[ ] Build the OpenWrt Image Builder
[ ] Build the OpenWrt SDK
[ ] Package the OpenWrt-based Toolchain
[ ] Image configuration  --->
    Base system  --->
    Administration  --->
    Boot Loaders  ----
    Development  --->
    Extra packages  ----
    Firmware  --->
    Fonts  --->
    Kernel modules  --->
    Languages  --->
    Libraries  --->
    LuCI  --->
    Mail  --->
    Multimedia  --->
    Network  --->
    Sound  --->
    Utilities  --->
    Xorg  --->
```

<Select> < Exit > < Help > < Save > < Load >

```
Target System --->
    Marvell 64b Boards
Target Profile --->
    ESPRESSObin (Marvell Armada 3700 Community Board)
Target Images  --->
    [x] ramdisk  --->
    * Root filesystem archives *
    [x] tar.gz
    * Root filesystem images *
    [x] ext4  --->
[x] Advanced configuration options (for developers)  --->
    (/opt/kernel/openwrt-kernel) Use external kernel tree
LuCI  --->
    1. Collections  --->
        <*> luci
```

- Save configuration & exit
- Issue build:

```
$ make -j$(($(nproc)+1))
```

- Copy build artifacts (device tree file, OpenWrt image and root file system) to the local directory and exit container:

```
$ cp bin/mvebu64/*armada* /opt/espressobin/
$ exit
```

Transferring binaries & booting

- Transfer binaries to SD card and boot OpenWrt from it

  Link: Boot from removable storage - OpenWrt

- Accessing LuCI - entering ESPRESSObin's IP address in URL bar
  - By default 192.168.1.1

EMBEDDED WORLD 2018

OpenWrt 101: How to Build a Linux Embedded System in Just 30 Minutes

luka.perkov@sartura.hr · cesare@prplfoundation.org