

An introduction



Steven Barth

Why bother?



Image: "[Internet Reboot](#)" by [Karl Baron](#); cropped; [CC BY 2.0](#)

At least 700,000 routers that ISPs gave to their customers are vulnerable to hacking security holes

Wi-Fi router has numerous

Remote access vulnerability
routers remain
unpatched

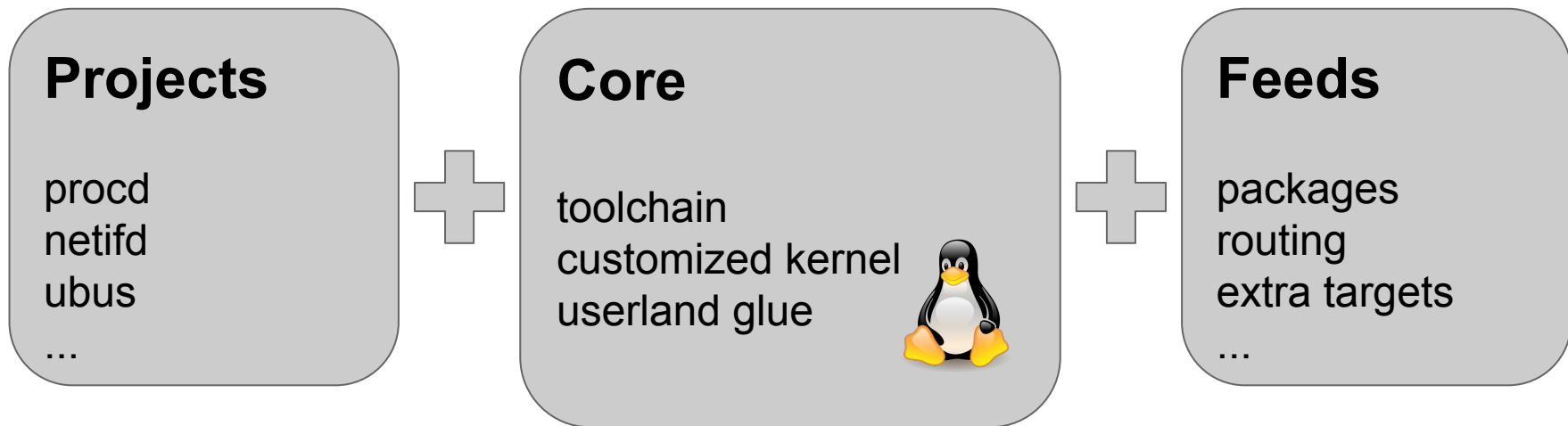
Linux.Wifatch: Routers hacked by 'white hat' virus that makes them more secure against malware

Hotel Router Vulnerability A Reminder Of Untrusted WiFi Risks

Stopping the race to the bottom

- code **quality** and long term **maintainability**
- sharing improvements and benefiting from others
- shift differentiation to where it really matters

OpenWrt/Linux in a nutshell



Core Developers

- loosely organized group of **volunteers**
- maintaining projects and core
- running wiki, forums, website
- managing the trademark

The Core

- `git.openwrt.org` + `dev.openwrt.org` + `openwrt-devel@openwrt.org`

Targets

Kernel Patches
DeviceTree files
Device Config
Default Profile



Base Packages

Kernel Modules
OpenWrt Userland
Networking / WiFi
Other
Das U-Boot branches



Toolchain

Shell, Make, Perl
GCC integration
~~uclibc~~ musl
Kconfig
Package templates
Package scripts

Build Configuration

```
.config - OpenWrt Configuration

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 (x86) --->
Subtarget (x86_64) ---->
Target Profile (Default) ---->
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 --->
Boot Loaders ---->
Development --->
Firmware --->
Kernel modules --->
Languages --->
Libraries --->
LuCI --->
Network --->
Utilities --->

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

Cross-compilation example

```
include $(TOPDIR)/rules.mk

PKG_NAME:=omcproxy
PKG_VERSION:=2015-08-24
PKG_RELEASE:=3

PKG_SOURCE:=$(PKG_NAME)-$(PKG_VERSION).tar.gz
PKG_SOURCE_SUBDIR:=$(PKG_NAME)-$(PKG_VERSION)
PKG_SOURCE_URL:=https://github.com/sbyx/omcproxy.git
PKG_SOURCE_PROTO:=git
PKG_SOURCE_VERSION:=8de9fa84e018e152e45c342f10b5b5140b63e4b1
PKG_MAINTAINER:=Steven Barth <cyrus@openwrt.org>
PKG_LICENSE:=APACHE-2.0

include $(INCLUDE_DIR)/package.mk
include $(INCLUDE_DIR)/cmake.mk

define Package/omcproxy
    SECTION:=net
    CATEGORY:=Network
    DEPENDS:=+libubox +libubus
    TITLE:=IGMPv3 and MLDv2 Multicast Proxy
endef

CMAKE_OPTIONS += -DWITH_LIBUBOX=1

define Package/omcproxy/install
    $(INSTALL_DIR) $(1)/etc/config
    $(INSTALL_CONF) ./files/omcproxy.config $(1)/etc/config/omcproxy
    $(INSTALL_DIR) $(1)/etc/init.d
    $(INSTALL_BIN) ./files/omcproxy.init $(1)/etc/init.d/omcproxy
    $(INSTALL_DIR) $(1)/usr/sbin
    $(INSTALL_BIN) $(PKG_BUILD_DIR)/omcproxy $(1)/usr/sbin/
endef

$(eval $(call BuildPackage,omcproxy))
```

1. Define name, version, revision
2. Define source and commit
3. Define distribution metadata
4. Enable build-tool integration (here: cmake)
5. Define package metadata (menu entry, dependencies)
6. Pass configure / cmake flags
7. Define installation instructions
8. Add package to build system

Userland Projects (examples)

netifd

event driven networking
device, vlan, tunnel setup
IPv4, IPv6, routes
protocol handlers

procd

process manager
monitoring, jailing
reload trigger
hotplug, syslog, init.d

LuCI & LuCI2

traditional Web-UI
&
JSON-RPC based
ubus-export

UCI

configuration interface
commit / rollback support
APIs to Lua and ubus

ubus

socket-based IPC bus
export & call methods
notifications
binary & JSON-API

libubox

small utility library
event loop
socket abstraction
common datastructs

The Feeds

- Developed on Github
- Community maintained

Routing

the original feed
routing + mesh

[github.com/
openwrt-routing](https://github.com/openwrt-routing)



Main Packages

formerly part of the core
self-governed guidelines

github.com/openwrt



Other Feeds

LuCI - WebUI
Telephony
Management
Outdated Targets
Abandoned Pkgs.

Releases

Cycle

Trunk (bleeding-edge) +
1 maintained stable release
Aiming for 6-12 month cycle



Stabilization Phase

Informal “soft-freeze” of **trunk**
RC1 built from **trunk**
Core and feeds branched off



Release Types

Core + Feeds release branch
SDK + Image-Builder
Binary images + Packages



Release Process

only semi-automated
lots of manual work and testing



OpenWrt in the Ecosystem

Communities

OpenWrt contributors
Bufferbloat.net
Meshing (e.g. Freifunk)
...

Engineers

ISP Laboratories
IETF Experiments
...

Companies

as SDK
as Firmware
as source for tools
...

Upstream

as source for patches
as source for feedback
as userbase

Areas for improvement

- Presentation?
- Communication?
- Documentation?
- Infrastructure?

Thank you for your attention! Questions?



Steven Barth
<cyrus@openwrt.org>