OpenWrt Hacking

Felix Fietkau

December 27, 2005

Overview

1 Introduction
- Overview
- What is OpenWrt?
- New features

Overview

1. Introduction
   - Overview
   - What is OpenWrt?
   - New features
2. Building packages and images
   - Software Development Kit
   - Image Builder
   - Package directories
   - The package Makefile

Overview

1. Introduction
   - Overview
   - What is OpenWrt?
   - New features
2. Building packages and images
   - Software Development Kit
   - Image Builder
   - Package directories
   - The package Makefile
3. Structure of the Buildroot
   - Buildroot directories
   - The `toolchain/` directory
   - The `package/` directory
   - The `target/` directory

Introduction to OpenWrt

- Linux distribution for embedded wireless routers

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL
- Stable version:

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL
- Stable version:
  - Linux 2.4

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL
- Stable version:
  - Linux 2.4
  - Single platform: BCM947xx (or BCM953xx)

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL
- Stable version:
  - Linux 2.4
  - Single platform: BCM947xx (or BCM953xx)
- Development version:

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL
- Stable version:
  - Linux 2.4
  - Single platform: BCM947xx (or BCM953xx)
- Development version:
  - Linux 2.4 and Linux 2.6 (depending on target platform)

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL
- Stable version:
  - Linux 2.4
  - Single platform: BCM947xx (or BCM953xx)
- Development version:
  - Linux 2.4 and Linux 2.6 (depending on target platform)
  - Texas Instruments AR7

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL
- Stable version:
  - Linux 2.4
  - Single platform: BCM947xx (or BCM953xx)
- Development version:
  - Linux 2.4 and Linux 2.6 (depending on target platform)
  - Texas Instruments AR7
  - Embedded X86 (Soekris, WRAP, etc.)

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL
- Stable version:
  - Linux 2.4
  - Single platform: BCM947xx (or BCM953xx)
- Development version:
  - Linux 2.4 and Linux 2.6 (depending on target platform)
  - Texas Instruments AR7
  - Embedded X86 (Soekris, WRAP, etc.)
  - Atheros AR531x/231x (planned)

Introduction to OpenWrt

- Linux distribution for embedded wireless routers
- Runs on inexpensive hardware
- Free Software under the GNU GPL
- Stable version:
  - Linux 2.4
  - Single platform: BCM947xx (or BCM953xx)
- Development version:
  - Linux 2.4 and Linux 2.6 (depending on target platform)
  - Texas Instruments AR7
  - Embedded X86 (Soekris, WRAP, etc.)
  - Atheros AR531x/231x (planned)
  - RouterBoard 5xx (planned)

New Features

- Version 1.0, codename: 'White Russian'

New Features

- Version 1.0, codename: 'White Russian'
  - Lots of bug fixes

New Features

- Version 1.0, codename: 'White Russian'
  - Lots of bug fixes
  - A web interface

New Features

- Version 1.0, codename: 'White Russian'
  - Lots of bug fixes
  - A web interface
  - Scripts for handling system events

New Features

- Version 1.0, codename: 'White Russian'
  - Lots of bug fixes
  - A web interface
  - Scripts for handling system events
  - Better performance and faster boot time

New Features

- Version 1.0, codename: 'White Russian'
  - Lots of bug fixes
  - A web interface
  - Scripts for handling system events
  - Better performance and faster boot time
- Version 2.0, codename: 'Kamikaze'

New Features

- Version 1.0, codename: 'White Russian'
  - Lots of bug fixes
  - A web interface
  - Scripts for handling system events
  - Better performance and faster boot time
- Version 2.0, codename: 'Kamikaze'
  - New platforms

New Features

- Version 1.0, codename: 'White Russian'
  - Lots of bug fixes
  - A web interface
  - Scripts for handling system events
  - Better performance and faster boot time
- Version 2.0, codename: 'Kamikaze'
  - New platforms
  - Network scripts rewrite

New Features

- Version 1.0, codename: 'White Russian'
  - Lots of bug fixes
  - A web interface
  - Scripts for handling system events
  - Better performance and faster boot time
- Version 2.0, codename: 'Kamikaze'
  - New platforms
  - Network scripts rewrite
  - Multiple SSIDs for the Broadcom platform

Building packages and images

Software Development Kit

- Generated from the Buildroot

Software Development Kit

- Generated from the Buildroot
- Compiles packages

Software Development Kit

- Generated from the Buildroot
- Compiles packages
- Uses buildroot-compatible package directories

Software Development Kit

- Generated from the Buildroot
- Compiles packages
- Uses buildroot-compatible package directories
- Contains precompiled versions of the toolchain and all libraries

Software Development Kit

- Generated from the Buildroot
- Compiles packages
- Uses buildroot-compatible package directories
- Contains precompiled versions of the toolchain and all libraries
- Using the SDK:

Software Development Kit

- Generated from the Buildroot
- Compiles packages
- Uses buildroot-compatible package directories
- Contains precompiled versions of the toolchain and all libraries
- Using the SDK:
  - Copy the package directory in package/

Software Development Kit

- Generated from the Buildroot
- Compiles packages
- Uses buildroot-compatible package directories
- Contains precompiled versions of the toolchain and all libraries
- Using the SDK:
  - Copy the package directory in package/
  - (optional) add dependencies in package/depend.mk

Software Development Kit

- Generated from the Buildroot
- Compiles packages
- Uses buildroot-compatible package directories
- Contains precompiled versions of the toolchain and all libraries
- Using the SDK:
  - Copy the package directory in package/
  - (optional) add dependencies in package/depend.mk
  - run make

Image Builder

- Generated from the Buildroot

Image Builder

- Generated from the Buildroot
- Builds images from package lists

Image Builder

- Generated from the Buildroot
- Builds images from package lists
- Contains all binary packages and image building software

Image Builder

- Generated from the Buildroot
- Builds images from package lists
- Contains all binary packages and image building software
- Using the Image Builder:

Image Builder

- Generated from the Buildroot
- Builds images from package lists
- Contains all binary packages and image building software
- Using the Image Builder:
  - (optional) Add extra packages to `packages/`

Image Builder

- Generated from the Buildroot
- Builds images from package lists
- Contains all binary packages and image building software
- Using the Image Builder:
  - (optional) Add extra packages to `packages/`
  - (optional) Add extra files to `files/`

Image Builder

- Generated from the Buildroot
- Builds images from package lists
- Contains all binary packages and image building software
- Using the Image Builder:
  - (optional) Add extra packages to `packages/`
  - (optional) Add extra files to `files/`
  - (optional) Modify/add package lists in lists/

Image Builder

- Generated from the Buildroot
- Builds images from package lists
- Contains all binary packages and image building software
- Using the Image Builder:
  - (optional) Add extra packages to `packages/`
  - (optional) Add extra files to `files/`
  - (optional) Modify/add package lists in lists/
  - run make

Anatomy of a package directory

- `./Config.in`
  - Menuconfig configuration items for the package

Anatomy of a package directory

- `./Config.in`
  - Menuconfig configuration items for the package
- `./Makefile`
  - Main makefile containing all build instructions

Anatomy of a package directory

- `./Config.in`
  - Menuconfig configuration items for the package
- `./Makefile`
  - Main makefile containing all build instructions
- `./ipkg/`
  - `ipkg` control files and scripts

Anatomy of a package directory

- `./Config.in`
  - Menuconfig configuration items for the package
- `./Makefile`
  - Main makefile containing all build instructions
- `./ipkg/`
  - `ipkg` control files and scripts
- `./patches/`
  - Contains patches for the package

Anatomy of a package directory

- `./Config.in`
  - Menuconfig configuration items for the package
- `./Makefile`
  - Main makefile containing all build instructions
- `./ipkg/`
  - `ipkg` control files and scripts
- `./patches/`
  - Contains patches for the package
- `./files/`
  - Extra files for the package or the package build

The package menuconfig file

- Syntax: Linux 2.6 menuconfig

The package menuconfig file

- Syntax: Linux 2.6 menuconfig
- Example:

```
config BR2_PACKAGE_WEBIF
    select BR2_PACKAGE_HASERL
    tristate "webif - OpenWrt Administrative Console"
    default y
    help
      A web interface for configuring OpenWrt
```

The package Makefile

- 1. Header:

The package Makefile

- 1. Header:

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

PKG_NAME:=strace
PKG_VERSION:=4.5.11
PKG_RELEASE:=1
PKG_MD5SUM:=28335e15c83456a3db055a0a0efcb4fe
PKG_SOURCE_URL:=@SF/strace
PKG_SOURCE:=$(PKG_NAME)-$(PKG_VERSION).tar.bz2
PKG_CAT:=bzcat
PKG_BUILD_DIR:=$(BUILD_DIR)/$(PKG_NAME)-$(PKG_VERSION)

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

The package Makefile

- 2. Package declaration:

The package Makefile

- 2. Package declaration:

```
$(eval $(call PKG_template,STRACE,strace,\
    $(PKG_VERSION)-$(PKG_RELEASE),$(ARCH)))
```

The package Makefile

- 2. Package declaration:

```
$(eval $(call PKG_template,STRACE,strace,\
    $(PKG_VERSION)-$(PKG_RELEASE),$(ARCH)))
```

- 3. Important build targets:

The package Makefile

- 2. Package declaration:

```
$(eval $(call PKG_template,STRACE,strace,\
    $(PKG_VERSION)-$(PKG_RELEASE),$(ARCH)))
```

- 3. Important build targets:

```
$(PKG_BUILD_DIR)/.configured:
#   run the package ./configure script
    touch $@
```

The package Makefile

- 2. Package declaration:

```
$(eval $(call PKG_template,STRACE,strace,\
    $(PKG_VERSION)-$(PKG_RELEASE),$(ARCH)))
```

- 3. Important build targets:

```
$(PKG_BUILD_DIR)/.configured:
#   run the package ./configure script
    touch $@


$(PKG_BUILD_DIR)/.built:
#   run the package makefile
    touch $@
```

The package Makefile

- 4. Package targets:

The package Makefile

- 4. Package targets:

```
$(IPKG_STRACE):
#   copy all files into $(IDIR_STRACE)
    $(IPKG_BUILD) $(IDIR_STRACE) $(PACKAGE_DIR)
```

The package Makefile

- 4. Package targets:

```
$(IPKG_STRACE):
#   copy all files into $(IDIR_STRACE)
    $(IPKG_BUILD) $(IDIR_STRACE) $(PACKAGE_DIR)
```

- 5. Extra targets:

The package Makefile

- 4. Package targets:

```
$(IPKG_STRACE):
#   copy all files into $(IDIR_STRACE)
    $(IPKG_BUILD) $(IDIR_STRACE) $(PACKAGE_DIR)
```

- 5. Extra targets:

```
mostlyclean:
    # run make clean in the package directory
```

The package Makefile

- 5. Extra targets:

```
compile-targets: install-dev
clean-targets: uninstall-dev

install-dev:
#   run make install and install development
#   headers/libraries into $(STAGING_DIR)

uninstall-dev:
#   remove all development files of this
#   package from $(STAGING_DIR)
```

Buildroot directories

- Source directories

Buildroot directories

- Source directories
  - `toolchain/`

Buildroot directories

- Source directories
  - `toolchain/`
  - `package/`

Buildroot directories

- Source directories
  - `toolchain/`
  - `package/`
  - `target/`

Buildroot directories

- Source directories
  - `toolchain/`
  - `package/`
  - `target/`
  - `scripts/`

Buildroot directories

- Source directories
  - `toolchain/`
  - `package/`
  - `target/`
  - `scripts/`
- Build directories

Buildroot directories

- Source directories
  - `toolchain/`
  - `package/`
  - `target/`
  - `scripts/`
- Build directories
  - `toolchain_build_<arch>/`

Buildroot directories

- Source directories
  - `toolchain/`
  - `package/`
  - `target/`
  - `scripts/`
- Build directories
  - `toolchain_build_`*<arch>*`/`
  - `staging_dir_`*<arch>*`/`

Buildroot directories

- Source directories
  - toolchain/
  - package/
  - target/
  - scripts/
- Build directories
  - toolchain_build_<*arch*>/
  - staging_dir_<*arch*>/
  - build_<*arch*>/

The toolchain/ directory

- unpacks kernel headers

The toolchain/ directory

- unpacks kernel headers
- builds binutils

The toolchain/ directory

- unpacks kernel headers
- builds binutils
- builds initial gcc

The toolchain/ directory

- unpacks kernel headers
- builds binutils
- builds initial gcc
- uses initial gcc to build uClibc

The toolchain/ directory

- unpacks kernel headers
- builds binutils
- builds initial gcc
- uses initial gcc to build uClibc
- builds final gcc

The toolchain/ directory

- unpacks kernel headers
- builds binutils
- builds initial gcc
- uses initial gcc to build uClibc
- builds final gcc
- (optional) builds binutils

The package/ directory

- Build directories for all packages

The package/ directory

- Build directories for all packages
- Exception: kernel-modules and kernel-specific packages

The package/ directory

- Build directories for all packages
- Exception: kernel-modules and kernel-specific packages
- `Makefile`
  - handles package dependencies

The package/ directory

- Build directories for all packages
- Exception: kernel-modules and kernel-specific packages
- `Makefile`
  - handles package dependencies
- `Config.in`
  - includes all package menuconfig files

The target/ directory

- linux/linux-<2.4|2.6>

The target/ directory

- linux/linux-<2.4|2.6>
  - Compiles/installs the kernel

The target/ directory

- linux/linux-<2.4|2.6>
  - Compiles/installs the kernel
  - Calls target/linux/package/

The target/ directory

- linux/linux-<2.4|2.6>
  - Compiles/installs the kernel
  - Calls target/linux/package/
- linux/package/

The target/ directory

- linux/linux-<2.4|2.6>
  - Compiles/installs the kernel
  - Calls target/linux/package/
- linux/package/
  - Similar to the main package/ directory

The target/ directory

- linux/linux-<2.4|2.6>
  - Compiles/installs the kernel
  - Calls target/linux/package/
- linux/package/
  - Similar to the main package/ directory
  - Builds kernel-specific packages (e.g. modules)

The target/ directory

- linux/linux-<2.4|2.6>
  - Compiles/installs the kernel
  - Calls target/linux/package/
- linux/package/
  - Similar to the main package/ directory
  - Builds kernel-specific packages (e.g. modules)
- linux/image/

The target/ directory

- linux/linux-<2.4|2.6>
  - Compiles/installs the kernel
  - Calls target/linux/package/
- linux/package/
  - Similar to the main package/ directory
  - Builds kernel-specific packages (e.g. modules)
- linux/image/
  - Builds firmware images

The target/ directory

- linux/linux-<2.4|2.6>
  - Compiles/installs the kernel
  - Calls target/linux/package/
- linux/package/
  - Similar to the main package/ directory
  - Builds kernel-specific packages (e.g. modules)
- linux/image/
  - Builds firmware images
  - Subdirectories for filesystems and system types