KTH Computer Science
and Communication

Configuration and device identification on network gateways

Configuration and device identification on network gateways

SIMON KERS

Bachelor's Thesis at STH
Supervisor: Micael Lundvall
Examiner: Ibrahim Orhan

Abstract

Abstract in English.

Referat

Konfigurering och enhetsidentifiering på nätverksgateways

Abstract på svenska.

Contents

Chapter 1

Introduction

Inteno Broadband Technology is an international corporate group that supplies customer premises equipment for internet service providers. The headquarter as well as the research and development center is located in Stockholm, Sweden. Inteno Open Systems Platform is a Linux-based open source platform running on customer premises equipment. It is based off the OpenWrt distribution which targets embedded devices, specifically network gateways. [1]

The support costs for partners and resellers of Intenos customer premises equipment are constantly looking to reduce support costs. This creates costs for the businesses and by simplifying configuration by abstracting common tasks of the end-user, these costs can be cut.

Using automatic device identification and automating common tasks such as port forwarding, support costs can be reduced and end-user satisfaction is higher. Many common support issues could be automated by the software running on the customer premises equipment and by effectively communication with the end-user through the user interface.

The OpenWrt distribution provides a complete platform for compiling and deploying a gateway image. By building a extensible library of presets for common port forwarding rules and developing a simple selection dialog, partners can customize their preset rules to any new device heading for market.

Part I

Important Results

Chapter 2

First One

2.1 Preliminaries

2.1.1 OpenWrt

OpenWrt is a free and open-source GNU/Linux distribution, targeting embedded devices, specifically wireless routers, but can run on almost any set of hardware. Cross-compilation is enabled by OpenWrt Buildroot, which compiles the C code using uClibc, a lightweight C library focusing on embedded Linux systems. It intends to be a meta distribution and offers developers a framework on which to base their firmware on.

OpenWrt is generally compiled and linked using gcc and binutils, with the help of Makefiles and patches for the various gcc versions and target platforms. Allowing end users as well as service operators and hardware manufacturers to compile the firmware. It offers the BusyBox set of barebones UNIX tools, enabling advanced users to fully interact with their Linux system and providing developers with a familiar platform for debugging and testing their product. [2]

2.1.2 OPKG

The package management system used in OpenWrt is OPKG. It is based off the discontinued ipkg and operates similar to APT and dpkg of Debian-based distributions. There are currently over 2000 OPKG packages available for OpenWrt.

The OpenWrt system and its packages are built using GNU Autoconf.

2.1.3 Inteno Open Platform System

For Customer Premises Equipment[1] like wireless gateways, Inteno Open Platform System offers an open-source Linux distribution based on OpenWrt. It uses the OpenWrt build system including cross-compilation toolchain to ensure compatibility with the ecosystem and upstream.

[1] commonly abbreviated as CPE

2.1.4 Lua Configuration Interface

LuCI is an suite of programs and libraries for extending OpenWrt using the Lua programming language. It originated in the OpenWrt project but has since grown and is now it's own project.

The themes are accessed from the directory:

```
root@Inteno:/usr/lib/lua/luci/view/themes/
```

Rules for port forwarding are read from:

```
/etc/config/firewall
```

A port forwarding rule which forwards external HTTP traffic over port 80 to the local IP 192.168.1.214, looks like:

```
config redirect
        option target 'DNAT'
        option src 'wan'
        option dest 'lan'
        option proto 'tcp'
        option src_dport '80'
        option dest_ip '192.168.1.214'
        option dest_port '80'
        option name 'Web server'
```

The presentation markup for the current port forwarding page in the LuCI backend on the Gateway, is defined in the file:

```
luci-inteno/applications/luci-firewall/luasrc/view/firewall/cbi_addforward.htm

libs/core/luasrc/model/firewall.lua :555
```

in the functions:

```
function redirect.*
```

2.1.5 Remarks

2.1.6 Definitions

See figure 2.1 on page 7.

2.2 The Main Theorem

2.2.1 Problem Statement

2.2.2 The Proof

Automatic port forwarding

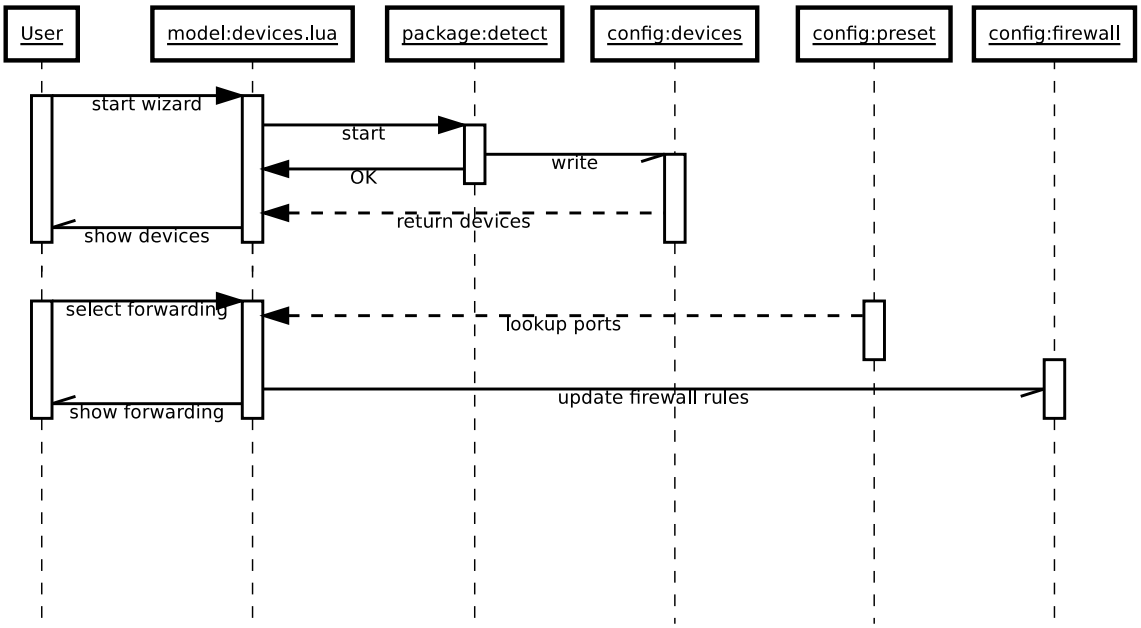

Figure 2.1. Sequential diagram of applying port forwarding rules

Appendix A

RDF

And here is a figure

Figure A.1. Several statements describing the same resource.

that we refer to here: A.1

Bibliography

[1] New business possibilities with Open Source software. http://www.inteno.se/Portals/0/IntenoFiles/ProductDocs/241/689/iopsys_white_paper.pdf_20121015135755.pdf. Accessed: 2013-04-29.

[2] OpenWrt structure and design. http://wiki.confine-project.eu/_media/soft:openwrt-talk-2012-06-01.pdf. Accessed: 2013-04-29.