Cross-compiling HOWTO

In a machine running Debian 7 (MiniPC 2)

Cross-compile simplemux.c with ROHC 1.7.0 in a Debian 7 machine, for running it in a TP-Link TL-WR1043ND (version 2) Access Point.

Some ideas here: http://wiki.openwrt.org/doc/devel/crosscompile

In the Debian machine (it must be 64 bits, as the TP-Link), download the Toolchain from OpenWRT.org

Go to `https://downloads.openwrt.org/snapshots/trunk/`

And download the version you need. In "binary releases" you have the latest version.

You can also download "historic releases". For example, this is the Barrier Breaker version:

https://downloads.openwrt.org/barrier_breaker/14.07/

I have downloaded from this one:

`https://downloads.openwrt.org/barrier_breaker/14.07/ar71xx/mikrotik/`

Download this Toolchain file:

```
https://downloads.openwrt.org/barrier_breaker/14.07/ar71xx/mikrotik/OpenWrt-Toolchain-ar71xx-for-mips_34kc-gcc-4.8-linaro_uClibc-0.9.33.2.tar.bz2
```

Extract it in

`/home/proyecto (if the username is "proyecto")`

In the Debian machine, modify the environment variable "CC"

Modify the CC variable in order to make the compiler be the MIPS one:

```
# export CC=/home/proyecto/OpenWrt-Toolchain-ar71xx-for-mips_34kc-gcc-4.8-linaro_uClibc-0.9.33.2/toolchain-mips_34kc_gcc-4.8-linaro_uClibc-0.9.33.2/bin/mips-openwrt-linux-gcc
```

Modify the STAGING_DIR variable in order to make the compiler be the MIPS one:

```
# export STAGING_DIR=/home/proyecto/OpenWrt-Toolchain-ar71xx-for-mips_34kc-gcc-4.8-linaro_uClibc-0.9.33.2/toolchain-mips_34kc_gcc-4.8-linaro_uClibc-0.9.33.2/
```

Use `#set | more` to confirm that `CC` has this value:

```
(…)
BASH_VERSION='4.2.37(1)-release'
CC=/home/proyecto/OpenWrt-Toolchain-ar71xx-for-mips_34kc-gcc-4.8-linaro_uClibc-0.9.33.2/toolchain-mips_34kc_gcc-4.8-linaro_uClibc-0.9.33.2/bin/mips-openwrt-linux-gcc
COLORTERM=gnome-terminal
(…)
```

In the Debian machine, download ROHC (file rohc-1.7.0.tar.xz from the ROHC web site)

Download the ROHC file from the web:

```
http://rohc-lib.org/download/rohc-1.7.x/1.7.0/rohc-1.7.0.tar.xz
```

Extract the file to the toolchain directory:

/home/proyecto/OpenWrt-Toolchain-ar71xx-for-mips_34kc-gcc-4.8-linaro_uClibc-0.9.33.2/toolchain-mips_34kc_gcc-4.8-linaro_uClibc-0.9.33.2/rohc-1.7.0

In the Debian machine, install the ROHC library in the toolchain

Go to this directory

/home/proyecto/OpenWrt-Toolchain-ar71xx-for-mips_34kc-gcc-4.8-linaro_uClibc-0.9.33.2/toolchain-mips_34kc_gcc-4.8-linaro_uClibc-0.9.33.2/rohc-1.7.0/

Run `configure`:

```
#./configure --disable-app-fuzzer --disable-app-performance --disable-app-sniffer --enable-app-tunnel --disable-app-stats --disable-linux-kernel-module --disable-doc --disable-doc-man --host=mips --prefix=/home/proyecto/OpenWrt-Toolchain-ar71xx-for-mips_34kc-gcc-4.8-linaro_uClibc-0.9.33.2/toolchain-mips_34kc_gcc-4.8-linaro_uClibc-0.9.33.2/
```

```
#make clean
```

```
#make
```

Note: `make all` may not work properly

`make check` does not work properly

```
#make install
```

After this, you have created in the `lib` folder of the toolchain the `.a` files required for compiling with static libraries (`librohc-common.a librohc-comp.a` y `librohc-decomp.a`)

Check if these files are in the folder:

```
~/OpenWrt-Toolchain-ar71xx-for-mips_34kc-gcc-4.8-linaro_uClibc-0.9.33.2/toolchain-mips_34kc_gcc-4.8-linaro_uClibc-0.9.33.2/lib
```

In the Debian machine, compile the .c file

This is the cross-compiling instruction:

The source file name is `/home/proyecto/simplemux/simplemux.c`

The created executable will be created in `/home/proyecto/simplemux-mips`

```
#/home/proyecto/OpenWrt-Toolchain-ar71xx-for-mips_34kc-gcc-4.8-linaro_uClibc-0.9.33.2/toolchain-mips_34kc_gcc-4.8-linaro_uClibc-0.9.33.2/bin/ ./mips-openwrt-linux-gcc -o /home/proyecto/simplemux-mips -g -Wall /home/proyecto/simplemux/simplemux.c -I ./include/ -L ./lib/ -lrohc_comp -lrohc -lrohc_common -lrohc_decomp -static
```

Now you can copy the executable file to the Access Point with MIPS architecture and run it there.