strongSwan - Bug #812

bliss plugin cannot be cross-compiled

05.01.2015 14:04 - G. V.

Status:	Closed	**Start date:**	05.01.2015
Priority:	Normal	**Due date:**	
Assignee:	Tobias Brunner	**Estimated time:**	0.00 hour
Category:	build		
Target version:	5.3.0		
Affected version:	5.2.2	**Resolution:**	Fixed

Description

The bliss plugin cannot be cross-compiled since bliss_huffman_code_[1|3|4}.c are generated by running bliss_huffman executable. But since bliss_huffman is compiled against the target architecture not the host architecture it will fail to run.

```
OpenWrt-libtool: link: ccache_cc -rdynamic -Os -pipe -mips32r2 -mtune=mips32r2 -fno-caller-saves -fhonour-copts -Wno-error=unused-but-set-variable -msoft-float -include /home/build/openwrt/build_dir/target-mips_r2_uClibc-0.9.33.2/strongswan-5.2.2/config.h -Wl,-rpath-link -Wl,/home/build/openwrt/staging_dir/target-mips_r2_uClibc-0.9.33.2/usr/lib -o bliss_huffman bliss_huffman.o  -L/home/build/openwrt/staging_dir/target-mips_r2_uClibc-0.9.33.2/usr/lib -L/home/build/openwrt/staging_dir/target-mips_r2_uClibc-0.9.33.2/lib -L/home/build/openwrt/staging_dir/toolchain-mips_r2_gcc-4.6-linaro_uClibc-0.9.33.2/usr/lib -L/home/build/openwrt/staging_dir/toolchain-mips_r2_gcc-4.6-linaro_uClibc-0.9.33.2/lib -lm ./.libs/libbliss-params.a
\
./bliss_huffman 1 8 > bliss_huffman_code_1.c 2>/dev/null
make[8]: *** [bliss_huffman_code_1.c] Error 126
make[8]: Leaving directory `/home/build/openwrt/build_dir/target-mips_r2_uClibc-0.9.33.2/strongswan-5.2.2/src/libstrongswan/plugins/bliss'
make[7]: *** [all-recursive] Error 1
```

Related issues:

Related to Issue #798: MAC OSX Compile - libbliss error	**Closed**	**23.12.2014**

Associated revisions

Revision faafaf35 - 02.03.2015 12:05 - Tobias Brunner

bliss: Add generated Huffman codes to the repository

While these files are generated they don't really change and are not architecture dependant. The previous solution prevented cross-compilation from the repository as `bliss_huffman` was built for the target system but was then executed on the build host to create the source files, which naturally was bound to fail.

The `recreate-bliss-huffman` make target can be used inside the bliss directory to update the source files if needed.

Fixes #812.

History

#1 - 05.01.2015 16:25 - Tobias Brunner

- Tracker changed from Issue to Bug

- Category changed from libcharon to libstrongswan

- Status changed from New to Feedback

- Assignee set to Tobias Brunner

- Priority changed from High to Normal

Building from the tarball should work as that contains the bliss_huffman_code_*.c files. Building from the repository on the other hand does not because, as you point out, these files have to be built with bliss_huffman, which itself has to be built from sources provided in the repository. You could perhaps create the sources with e.g. make distdir (or any other dist* target) first and then build your cross-compiled binaries from that.

To enable building directly from the repository we might be able to use AX_PROG_CC_FOR_BUILD and then write our own rules to build bliss_huffman.

#2 - 05.01.2015 16:25 - Tobias Brunner

- Description updated

- Category changed from libstrongswan to build

#3 - 05.01.2015 18:43 - G. V.

> Building from the tarball should work as that contains the bliss_huffman_code_*.c files.

True, as long you don't have to run autoreconf.

> Building from the repository on the other hand does not because, as you point out,
> these files have to be built with bliss_huffman, which itself has to be built from
> sources provided in the repository. You could perhaps create the sources with e.g.
> make distdir (or any other dist* target) first and then build your cross-compiled
> binaries from that.

It will break the automated build.

My temporary solution was to manually save bliss_huffman_code_*.c files in a temporary directory
and restore them with cp -f ... instead of running bliss_huffman (patch for Makefile.am).

Still, I do not understand this:
why those files need to be generated if they are not target arch dependent?

Maybe a script (bash/perl/bison/yacc/etc.) to generate bliss_huffman_code_*.c files is a better idea.

G.V.

#4 - 17.02.2015 17:05 - Tobias Brunner

- Target version set to 5.3.0

> Still, I do not understand this:
> why those files need to be generated if they are not target arch dependent?

You are absolutely right. They don't even are expected to change after generating them once (unless e.g. the interface *bliss_huffman_code_t* changes).

I pushed a commit to the *bliss-cross-compile* branch that removes this build dependency. The files are now committed to the repository, but still can be recreated manually, if needed, with a special make target.

#5 - 02.03.2015 12:09 - Tobias Brunner

- Status changed from Feedback to Closed

- Resolution set to Fixed

#6 - 23.03.2015 17:51 - Tobias Brunner

- Related to Issue #798: MAC OSX Compile - libbliss error added