Instructions: How to Compile Wireshark with IDLs in a Linux Environment

CONTENTS

LIST OF TABLES

FOREWORD

This document describes how to compile and install the Open Source Wireshark network debugging tool with CORBA IDL files. These instructions include a software list that contains the SW used in the test environment.

B.1 SCOPE

The intent of this document is to describe how to compile the Open Source Wireshark network debugging tool with CORBA IDL files on the Linux Operating System.

B.2 SOFTWARE REQUIRED

2.1 Tested Operating Systems

Operating System (OS)	Kernel Version	Manufacturer
SUSE Linux Enterprise Server 10 Service Pack 2	2.6.16.60-0.21-default (ia64)	Novell
Red Hat Linux Enterprise Server 5.1	2.6.18-53.e15 (i686)	Red Hat
Fedora 17	3.6.11-5.fc17.x86_64 x86_64	Open Source

Table : Tested Operating Systems List

B.3 HOW TO BUILD AND INSTALL A CUSTOM VERSION OF WIRESHARK

This section provides direction on how to build and install a custom version of Wireshark. If this section has already been completed on the target machine, and the only thing changing for the entire build are the CORBA IDL files then you can continue to section B.4.

3.1 Create Development and Installation Folders

Create the folders */home/user/wireshark/dev* and */home/user/wireshark/inst* with user privileges allowing the builder to read and write to files within them. Create a folder called "*bin*" inside of the */home/user/wireshark/inst* folder.

3.2 Unpack Supporting Software Packages

Unpack the following software packages to the */home/user/wireshark/dev* folder:

- m4-x.y.z.tar.bz2
- autoconf-x.y.tar.gz
- Python-x.y.z.tar
- omniORB-x.y.z.tar.gz
- omniORBpy-x.y.tar.gz
- wireshark-x.y.z.tar.bz2

3.3 Install ConvertIDLs Tool

Copy the convertidls file to the */home/user/wireshark/inst/bin* folder.

3.4 Link Supporting Software Packages

Create soft-links to the unpacked supporting software package folders in the */home/user/wireshark/dev* folder with the names listed below:

- m4-1.4.15.tar.bz2 m4
- autoconf-2.68.tar.gz autoconf
- Python-2.6.6.tar Python
- omniORB-4.1.4.tar.gz omniORB
- omniORBpy-3.4.tar.gz omniORBpy
- wireshark-1.4.2.tar.bz2 wireshark

3.5 Modify PATH System Variable

Within a BASH shell execute the following command to allow building within user defined areas:

#> export PATH=/home/user/wireshark/inst/bin:$PATH:/home/user/wireshark/inst/bin

3.6 Build M4 Software

Move to the */home/user/wireshark/dev/m4 folder* within the same BASH shell used in 3.5. Execute the following commands:

a) ./configure --prefix=*/home/user/wireshark/inst*

b) make

c) make install

3.7 Build Autoconf Software

Move to the */home/user/wireshark/dev/autoconf* folder within the same BASH shell used in 3.5. Execute the following commands:

a) ./configure --prefix=*/home/user/wireshark/inst*

b) make

c) make install

3.8 Build Python Software

Move to the */home/user/wireshark/dev/Python* folder within the same BASH shell used in 3.5. Execute the following commands:

a) ./configure --prefix=*/home/user/wireshark/inst*

b) make

c) make install

3.9 Build omniORB Software

Move to the */home/user/wireshark/dev/omniORB* folder within the same BASH shell used in 3.5. Execute the following commands:

a) ./configure --prefix=*/home/user/wireshark/inst*

b) make

c) make install

3.10 Build omniORBpy Software

Move to the */home/user/wireshark/dev/omniORBpy* folder within the same BASH shell used in 3.5. Execute the following commands:

a) ./configure --prefix=*/home/user/wireshark/inst*

b) make

c) make install

3.11 Build Wireshark Software

Move to the */home/user/wireshark/dev/wireshark* folder within the same BASH shell used in 3.5. Execute the following commands:

a) ./autogen.sh

b) you will need here to install other libraries:

   a. sudo yum install bison

   b. sudo yum install flex

   c. sudo yum install libpcap-devel

c) ./configure --prefix=*/home/user/wireshark/inst*

d) make

e) make install

B.4 BUILD AND INSTALL WIRESHARK WITH CUSTOM IDL FILE DISSECTORS

This section provides direction on how to build and install a custom version of wireshark with custom CORBA IDL file dissectors. **SECTION** B.3 **MUST BE COMPLETED PRIOR TO STARTING THIS SECTION OF THE PROCEDURE.**

4.1 Modify 'wireshark_gen.py' File

Using any text editor open the file *'/home/user/wireshark/dev/wireshark/tools/wireshark_gen.py'*. Perform the steps below to comment out certain sections of the *wireshark_gen.py* file that create problems with custom dissector builds.

a) Search for the string *'template_plugin_register ='*. Add *'//'* to the beginning of each line starting from the line that contains *'#ifndef ENABLE_STATIC'* to the line that contains *'#endif'.*

b) Search for the string *'version[]'*. Add *'//'* to the beginning of the current line, the line before it starting with *'#ifndef'* and the line after it starting with *'#endif'.*

c) Create and Populate Developmental IDL Folder

Create the folder */home/user/wireshark/dev/idls* with user privileges allowing the builder to read and write to files within them. Place all IDL files that are to be used in wireshark into the */home/user/wireshark/dev/idls* folder keeping subdirectory structures.

***NOTE: Make sure all folder names match the content of the #includes found within the IDL files or there will be compilation issues!**

Error Example (Bad Case Match):

IDL File: #include "MEADS**t**ypes/MEADSCompositeTypes/MEADSBaseTypes/tnIDNumber.idl"

Dir Structure: ~\MEADS**T**ypes\MEADSCompositeTypes\MEADSBaseTypes

4.2 Copy Wireshark IDL to C/C++ Generation Scripts

Copy the files */home/user/wireshark/dev/wireshark/tools/wireshark_be.py* and */home/user/wireshark/dev/wireshark/tools/wireshark_gen.py* to the folder */home/user/wireshark/inst/lib/python2.6/site-packages*.

4.3 Modify PATH System Variable

If the BASH shell from 3.5 is still available move to section 4.4. If it is not still available please redo section 3.5 before continuing to section 4.4.

4.4 Convert IDL Files to C Files

Use the command omniidl to convert idl to c files that will be used after.

This command will be like this:

```
../inst/bin/omniidl -p ../dev/wireshark/tools -b wireshark_be -I . Messenger.idl > Messenger.c
```

4.5 Copy Custom Wireshark Plugin Descriptor Files

Copy the following files to the '*/home/user/wireshark/dev/wireshark/plugins/NewMessages'* folder:

- Makefile.am
- Makefile.common
- Makefile.nmake
- moduleinfo.h

- moduleinfo.nmake
- plugin.rc.in
- CMakeLists.txt

4.6 Modify Custom Plugin 'CMakeList.txt' File

Using any text editor open the file *'/home/user/wireshark/dev/wireshark/plugins/NewMessages/CMakeList.txt'*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Starting with the line after '`set(DISSECTOR_SRC`' add all converted .c file names from the resulting .c files from section 4.4. Each file name should be on its own line without any commas after it.

b) Change the text right after '`add_library(`' and before the space to *NewMessages.*

c) Change the text right after '`set_target_properties(`' and before the space to *NewMessages. There are three (3) of lines that contain this, and all three (3) need to be changed.*

d) Change the text right after '`target_link_libraries( `' and before the space to *NewMessages.*

e) Change the text right after '`install(TARGETS `' to *NewMessages.*

4.7 Modify Custom Plugin 'Makefile.am' File

Using any text editor open the file *'/home/user/wireshark/dev/wireshark/plugins/NewMessages/Makefile.am'*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Set "`plugin_LTLIBRARIES`" = NewMessages.la

b) Replace the text on the line after '`CLEANFILES = \`' with *'NewMessages \'*.

c) Replace the text just in front of '`_la_SOURCES =`' with *'NewMessages'*.

d) Replace the text just in front of '`_la_LDFLAGS =`' with *'NewMessages'*.

e) Replace the text just in front of '`_la_LIBADD =`' with *'NewMessages'*.

4.8 Modify Custom Plugin 'Makefile.common' File

Using any text editor open the file *'/home/user/wireshark/dev/wireshark/plugins/NewMessages/Makefile.common*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Replace the text on the line after '`PLUGIN_NAME =`' with '`NewMessages \`'.

b) Starting with the line after '`DISSECTOR_SRC = \`' add all converted .c file names from the resulting .c files from section 4.4. Each file name should be on its own line with '` \`' after it except for the last item in the list.

4.9 Modify Custom Plugin 'moduleinfo.h' File

Using any text editor open the file *'/home/user/wireshark/dev/wireshark/plugins/NewMessages/moduleinfo.h'*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Replace the text on the line after ‘`#define PACKAGE`’ with ‘`NewMessages`’. Keep the quotes around `NewMessages`.

b) Replace the text on the line after ‘`#define VERSION` with whatever version number you want to use. Keep the quotes around the version number.

4.10 Modify Wireshark ‘configure.in’ File

Using any text editor open the file *‘/home/user/wireshark/dev/wireshark/configure.in’*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Search for ‘`AC_OUTPUT`’ and add ‘`plugins/NewMessages/Makefile`’ to the list on its own line.

4.11 Modify Wireshark ‘CMakeLists.txt’ File

Using any text editor open the file *‘/home/user/wireshark/dev/wireshark/CMakeLists.txt’*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Search for ‘`PLUGIN_SRC_DIRS`’ and add ‘`plugins/NewMessages`’ to the list on its own line.

4.12 Modify Wireshark ‘Makefile.am’ File

Using any text editor open the file *‘/home/user/wireshark/dev/wireshark/Makefile.am’*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Search for ‘`plugin_ldadd =`’ and add ‘`-dlopen plugins/NewMessages/NewMessages.la`’ to the list on its own line.

4.13 Modify Wireshark Plugin Folder ‘Makefile.am’ File

Using any text editor open the file *‘/home/user/wireshark/dev/wireshark/plugins/Makefile.am’*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Search for ‘`SUBDIRS =`’ and add ‘`NewMessages`’ to the list on its own line with ‘ `\`’ after it unless it is the last item in the list.

4.14 Modify Wireshark Plugin Folder ‘Makefile.in’ File

Using any text editor open the file *‘/home/user/wireshark/dev/wireshark/plugins/Makefile.in’*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Search for ‘`SUBDIRS =`’ and add ‘`NewMessages`’ to the list on its own line with ‘ `\`’ after it unless it is the last item in the list.

1.1 Modify Wireshark EPAN ‘Makefile.am’ File

Using any text editor open the file *‘/home/user/wireshark/dev/wireshark/epan/Makefile.am*. Perform the steps below to allow the custom Wireshark Plugin to compile correctly.

a) Starting with the line after ‘`plugin_src = \`’ add all converted .c file names from the resulting .c files from section 4.4. Each file name should be on its own line with ‘`../plugins/NewMessages/`’ before it, and each new line should should have ‘ `\`’ after it except for the last item in the entire list.

4.15 Rebuild Wireshark Software with Custom Plugin

Move to the */home/user/wireshark/dev/wireshark* folder within the same BASH shell used in 3.5. Execute the following commands:

a) ./autogen.sh **(NOTE: FIX ANY PROBLEMS THAT THIS SCRIPT MAY COMPLAIN ABOUT!)**

b) ./configure --prefix=*/home/user/wireshark/inst*

c) make

d) make install

APPENDIX A: Custom Logo

This appendix contains the steps to use a custom graphic for the built Wireshark resulting from this procedure.

B.5 Software Required

1. GIMP 2.6.11 (Image Editing Software)

B.6 Create Custom Logo

1. Create a copy of the following file with a custom name still containing the extension “.xpm”:
   “@WSS \image\wssplash-dev.xpm” ▯ ***“@WSS\image\wssplash-custom.xpm”***

2. Open the new “.xpm” file with the GIMP image editing software.

3. Edit and Save the Image

B.7 Update Wireshark Build Files

1. Open the file ***“@WSS\gtk\main_welcome.c”***, and search for ***“xpm”***. The first result should be in the “Include” section.

2. Change the resulting line to: ***#include*** **"*../image/ wssplash-custom.xpm* "**

3. Open the file ***“@WSS\gtk\about_dlg.c”***, and search for ***“xpm”***. The first result should be in the “Include” section.

4. Change the resulting line to: ***#include*** **"*../image/ wssplash-custom.xpm* "**

APPENDIX B: Custom Version Information

This appendix contains the steps to use custom version information for the built Wireshark resulting from this procedure.

C.1 Update Wireshark Build Files

1. Open the file ***“@WSS\config.h”*** and modify the following:
   a. Search for ***“VERSION”*** and modify the value to represent the current build.
      i. *Example:*

```
VERSION_EXTRA="VA.VB.VC-PROJECTNAME-VERSION-DESCRIPTION"
```