一、gsoap编译

下载gsoap2.8.17编译器：<http://sourceforge.net/projects/gsoap2/files/>

解压：unzip gsoap\_2.8.17.zip

配置：cd gsoap-2.8/

./configure

生成Makefile文件。

编译：make

安装：make install

在当前目录的gsoap目录下即可生成soapcpp2和wsdl2h两个工具。

验证：终端输入soap和wsdl，若能补齐为soapcpp2和wsdl2h，即表示以上两个已被安装到系统中。

二、onvif文件下载

1. remotediscovery.xml文件下载

浏览器打开：<http://www.onvif.org/onvif/ver20/util/operationIndex.html>

选择RemoteDiscovery下的任一选项，右键”目标另存为”，即可把remotediscovery.xml保存在本地目录下。

2. devicemgmt.xml文件下载

浏览器打开：<http://www.onvif.org/onvif/ver20/util/operationIndex.html>

选择DeviceMgmt下的任一选项，右键”目标另存为”，即可把devicemgmt.xml保存在本地目录下。

3. onvif.xsd.xml文件下载（可选）

onvif.xsd文件也可以不下载，在编译时，gsoap会自动上网查找这个文件。

使用chrome浏览器打开：<http://www.onvif.org/ver10/schema/onvif.xsd>

待完全展开之后，右键”另存为”，即可把onvif.xsd.xml文件保存到本地目录下。

三、onvif源文件编译

1. 编写onvif的C++源文件，并将它放在某个目录下，如OnvifTest。

2. 将remotediscovery.xml、onvif.xsd.xml和Makefile也放在OnvifTest目录下。

3. 修改.xml文件的后缀名，将remotediscovery.xml改为remotediscovery.wsdl，将onvif.xsd.xml修改为onvif.xsd，将devicemgmt.xml改为devicemgmt.wsdl。

注意：如果将修改了devicemgmt.xml的后缀名改成了devicemgmt.wsdl，则必须要定义soapStub.h中声明的函数，不然就会出现函数未定义的情况。

4. 修改devicemgmt.wsdl

schemaLocation="../../../ver10/schema/onvif.xsd"改为

schemaLocation="onvif.xsd"

不然就会编译错误：

create depend soapClient.d : soapClient.cpp ......................

g++: 错误： soapClient.cpp：没有那个文件或目录

g++: 致命错误： 没有输入文件

5. 修改gsoap的安装文件：

在gsoap-2.8目录下查找：find -name wsa5.h 得到wsa5.h的目录：gsoap/import/wsa5.h

vim gsoap/import/wsa5.h

L277：SOAP\_ENV\_\_Fault改为SOAP\_ENV\_\_Fault\_alxt

4. 修改Makefile：

GSOAP\_DIR=/home/gsoap-2.8

将GSOAP\_DIR指定为gsoap-2.8的绝对路径。

4. 编译：make

即可生成可执行文件。

错误集锦：

1. create depend soapClient.d : soapClient.cpp ......................

g++: 错误： soapClient.cpp：没有那个文件或目录

g++: 致命错误： 没有输入文件

2. undefined reference to `soap\_default\_xsd\_\_duration

这个问题是因为在gSOAP工具中提供的typemap.dat文件中有下面的类型转换规则：

# Uncomment the line below to use LONG64 int for xsd:duration

# (with high ms precision)

# Must compile and link custom/duration.c with the project

xsd\_\_duration = #import "custom/duration.h" | xsd\_\_duration 而你在工程中没有加入和编译链接gsoap目录下的custon/duration.c文件。因此解决办法有两个：

（1）需要将\custom文件夹下面的duration.h和duration.c导入工程中。

这样还不行。由于使用C++编译器 因此需要将duration.c文件改为.cpp文件。

（2）修改你的typemap.dat文件中的上述转换规则，不使用LONG64，只使用long类型。

# Uncomment the line below to use LONG64 int for xsd:duration

# (with high ms precision)

# Must compile and link custom/duration.c with the project

#xsd\_\_duration = #import "custom/duration.h" | xsd\_\_duration

3. make错误。

g++ -o myClient soapC.o stdsoap2.o soapClient.o myClient.o -lpthread -lm

soapC.o: In function `tt\_\_DurationRange::soap\_default(soap\*)':

soapC.cpp:(.text+0xa854): undefined reference to `soap\_default\_xsd\_\_duration(soap\*, long long\*)'

soapC.o: In function `tt\_\_RelayOutputSettings::soap\_default(soap\*)':

soapC.cpp:(.text+0xa891): undefined reference to `soap\_default\_xsd\_\_duration(soap\*, long long\*)'

soapC.o: In function `tt\_\_ReplayConfiguration::soap\_default(soap\*)':

soapC.cpp:(.text+0xa8ca): undefined reference to `soap\_default\_xsd\_\_duration(soap\*, long long\*)'

分析解决：如果将修改了devicemgmt.xml的后缀名改成了devicemgmt.wsdl，则必须要定义soapStub.h中声明的函数，不然就会出现函数未定义的情况。此时将devicemgmt.wsdl改成其他名称如devicemgmt.wsdl\_bk，执行make deepClean，然后再执行make就可以了。

4. make错误：make: \*\*\* 没有规则可以创建“libonvif”需要的目标“libonvif.o”。 停止。

分析解决：