This file describes how to build and run Suricata on Windows. Currently

Windows XP and above are supported.

Preparing the build environment

===============================

**1. Setup MinGW environment from http://mingw.org**

Do not use the automatic installer as it is deprecated. Manually unpack

the following packages to c:\mingw (use newer versions if you like):

\* binutils

o **binutils-2.20–1-mingw32-bin.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/BaseSystem/GNU-Binutils/binutils-2.20/binutils-2.20-1-mingw32-bin.tar.gz/download>

\* mingw-runtime (dev and dll):

o **mingwrt-3.17-mingw32-dll.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/BaseSystem/RuntimeLibrary/MinGW-RT/mingwrt-3.17/mingwrt-3.17-mingw32-dll.tar.gz>

o **mingwrt-3.17-mingw32-dev.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/BaseSystem/RuntimeLibrary/MinGW-RT/mingwrt-3.17/mingwrt-3.17-mingw32-dev.tar.gz/download>

\* w32api

o **w32api-3.14-mingw32-dev.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/BaseSystem/RuntimeLibrary/Win32-API/w32api-3.14/w32api-3.14-mingw32-dev.tar.gz/download>

\* required runtime libraries for GCC (gmp, libiconv, MPFR and pthreads):

o **gmp-4.2.4-mingw32-dll.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/BaseSystem/GCC/Version4/Previous%20Release%20gcc-4.4.0/gmp-4.2.4-mingw32-dll.tar.gz>

o **libiconv-1.13.1–1-mingw32-dll-2.tar.lzma**

<http://sourceforge.net/projects/mingw/files/MinGW/libiconv/libiconv-1.13.1-1/libiconv-1.13.1-1-mingw32-dll-2.tar.lzma>

o **mpfr-2.4.1-mingw32-dll.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/BaseSystem/GCC/Version4/Previous%20Release%20gcc-4.4.0/mpfr-2.4.1-mingw32-dll.tar.gz/download>

o **pthreads-w32–2.8.0-mingw32-dll.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/BaseSystem/GCC/Version4/Previous%20Release%20gcc-4.4.0/pthreads-w32-2.8.0-mingw32-dll.tar.gz/download>

\* gcc-core (bin and dll):

o **gcc-core-4.4.0-mingw32-bin.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/BaseSystem/GCC/Version4/Previous%20Release%20gcc-4.4.0/gcc-core-4.4.0-mingw32-bin.tar.gz>

o **gcc-core-4.4.0-mingw32-dll.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/BaseSystem/GCC/Version4/Previous%20Release%20gcc-4.4.0/gcc-core-4.4.0-mingw32-dll.tar.gz/download?use_mirror=puzzle>

\* make

o **make-3.81–20090914-mingw32-bin.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/make/make-3.81-20090914-mingw32/make-3.81-20090914-mingw32-bin.tar.gz/download>

\* zlib

o **libz-1.2.3-1-mingw32-dll-1.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/zlib/zlib-1.2.3-1-mingw32/libz-1.2.3-1-mingw32-dll-1.tar.gz/download>

o **libz-1.2.3-1-mingw32-dev.tar.gz**

<http://sourceforge.net/projects/mingw/files/MinGW/zlib/zlib-1.2.3-1-mingw32/libz-1.2.3-1-mingw32-dev.tar.gz/download>

**2. Install MSYS**

http://sourceforge.net/projects/mingw/files/

**MSYS-1.0.11.exe (MSYS Base System) (double click this .exe)**

<http://sourceforge.net/projects/mingw/files/MSYS/BaseSystem/msys-core/msys-1.0.11/MSYS-1.0.11.exe/download>

**msysDTK-1.0.1.exe (MSYS Suplementary Tools)**

<http://sourceforge.net/projects/mingw/files/MSYS/Supplementary%20Tools/msysDTK-1.0.1/msysDTK-1.0.1.exe/download>

**autoconf-2.63–1-msys-1.0.11-bin.tar.lzma (unpack with 7zip then untar in the C:\msys\1.0\ directory)**

<http://sourceforge.net/projects/mingw/files/MSYS/autoconf/autoconf-2.63-1/autoconf-2.63-1-msys-1.0.11-bin.tar.lzma/download>

**automake-1.11–1-msys-1.0.11-bin.tar.lzma(unpack with 7zip then untar in the C:\msys\1.0\ directory)**

<http://sourceforge.net/projects/mingw/files/MSYS/automake/automake-1.11-1/automake-1.11-1-msys-1.0.11-bin.tar.lzma/download>

**libtool-2.2.7a-1-msys-1.0.11-bin.tar.lzma(unpack with 7zip then untar in the C:\msys\1.0\ directory)**

<http://sourceforge.net/projects/mingw/files/MSYS/libtool/libtool-2.2.7a-1/libtool-2.2.7a-1-msys-1.0.11-bin.tar.lzma/download>

MSYS will ask questions during the installation:

Accept Post Install: [y]

MinGW Installed? : [y]

path to MinGW: [c:/MinGW]

**3. Install pkg-config taken from**

http://wiki.videolan.org/Win32CompileMSYSNew#PKG-CONFIG

Download and extract the following into c:\Msys\1.0

<http://ftp.gnome.org/pub/GNOME/binaries/win32/glib/2.18/glib_2.18.2-1_win32.zip>

<ftp://ftp.gnome.org/pub/gnome/binaries/win32/dependencies/pkg-config_0.23-3_win32.zip>

<ftp://ftp.gnome.org/pub/gnome/binaries/win32/dependencies/pkg-config-dev_0.23-3_win32.zip>

Set PKG\_CONFIG\_PATH=/win32/lib/pkgconfig

(e.g. by adding the Windows enviroment variable PKG\_CONFIG\_PATH in "Control Panel"->"System"->"Advanced System Settings"->"Environment Variables" and setting the value to /win32/lib/pkgconfig)

**4. Get git**

Download portable GIT from this URL:

<http://code.google.com/p/msysgit/>

- unpack to /msys/1.0

- don't forget to edit your ~/.gitconfig to at least give youreself a name :-)

**5. Get libpcre**

Download and extract the pcre-8.02 into c:\Msys\1.0

<http://www.pcre.org/>

./configure --enable-utf8 --disable-cpp --prefix=/mingw

make

make install

**6. Get libyaml**

Download and extract the yaml-0.1.3 into c:\Msys\1.0

<http://pyyaml.org/wiki/LibYAML>

It does not support mingw compilation. However it works in static mode:

./configure --prefix=/mingw CFLAGS="-DYAML\_DECLARE\_STATIC"

make

make install

**7. Get libpcap**

Guide can be found here:

Download and install WinPcap, then download the same version’s developers package.

- Download Devlopers pack <http://www.winpcap.org/devel.htm>

- Download and install a coresponding installer package http://www.winpcap.org/install/default.htm (to have the driver in the system)

- Unpack the developers package. Copy includes to c:/mingw/include and libs (.a) to c:/mingw/lib

- Rename libwpcap to libpcap

**8. Get and compile Suricata**

git clone git://phalanx.openinfosecfoundation.org/oisf.git

cd oisf

Because of some weird autools port bug we do the following:

dos2unix.exe libhtp/configure.ac

dos2unix.exe libhtp/htp.pc.in

dos2unix.exe libhtp/Makefile.am

./autogen.sh

./configure CFLAGS="-DYAML\_DECLARE\_STATIC"

# add --enable-nfqueue as parameter to configure to enable inline mode

make

If everything goes well, you'll end up with suricata.exe in src/.lib. To test it

you will need libpcre-0.dll, libz-1.dll (**found in C:\mingw\bin\**), and pthreadGC2.dll(**found in C:\msys\1.0\bin\**) which you already have somewhere

under c:/mingw or c:/msys. To prepare the runtime environment:

- copy the executable and the DLLs to a dedicated directory

- get there classification.config (**C:\msys\1.0\oisf\classification.config**) and suricata.yaml (**C:\msys\1.0\oisf\suricata.yaml**)

- edit suricata.yaml (at least set the directories correctly)

PCAP Mode

=========

Make sure you have winpcap runtime and driver installed and then:

- determine your eth device UUID in the registry:

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\

- now cross your fingers and do:

suricata.exe -c suricata.yaml -i \DEVICE\{your device uuid}

Inline Mode

===========

You need to downoad, compile and install netfilterforwin (the netfilter.sys

driver and Windows port of the libnetfilter\_queue library):

1. Download and install Windows Driver Kit from Microsoft

http://www.microsoft.com/downloads/details.aspx?displaylang=en&FamilyID=36a2630f-5d56-43b5-b996-7633f2ec14ff

2. Download netfilterforwin

http://sourceforge.net/projects/netfilterforwin/

Unpack it so the netfilterforwin directory (omit the version from its name)

is beside the oisf directory.

3. Compile the driver

- Open the build environment from you Start menu:

Start->All Programs->windows Driver Kits->WDK xxxx.yyyy.z->Build Environments

->Windows Server 2003->x86 Free Build Environment

(or the one which is proper for your system)

- cd to netfilterforwin/netfilter

- enter command:

nmake

4. Install the driver

- copy inf/\* files and the freshly built netfilter.sys to a separate directory

- open network connecions

- right-click an interface, select properties

- click install...

- select service

- click add

- click 'have disk...'

- browse to the directory with the inf files and netfilter.sys, select netfilter.inf anc click ok

- confirm everything

You should have the driver installed now.

5. Run Suricata in inline mode:

suricata.exe -c suricata.yaml -q 0