



Windows Installation Guide for Suricata IDS/IPS

This is a Suircata Windows Installation Guide -

Compilation from scratch.

Tested on Win XP, Windows Vista, Windows 7, Windows Server 2003, Windows Server 2008R2 64 bit.

Date: 26 May 2012 Document Version: 1.3

Author: Peter Manev









SECTION I - ADVANCED USERS	4
Step 1 – You need to download and install Cygwin	4
Step 2 – Make sure you have WinPcap installed.	5
Step 3 Add some paths to the system path.	6
Step 4 Get libyaml	6
Step 5 Get and compile Suricata. Suricata from git: Suricata Stable (at the moment of this writing the stable version is 1.2.1):	6 7 7
Ste 5.1 – MD5 support For stable - For git –	8 8 8
Step 6 Configure and run Suricata	10
SECTION II - STEP BY STEP FOR NEWBIE	13
Step 1 Download Cygwin	13
Step 2 Install extra packages	24
Step 3 Add paths to system variables	30
Step 4 Get libyaml	33
Step 5 Get libpcap – for windows	33
Step 6 Start Cygwin and compile yaml	34
Step 7 Compile Suricata Step 7.1 – Suricata from git – latest version Step 7.2 Suricata stable	39 39 43
Step 8 Set up Suricata for Windows	45
Step 9 Runing Suricata	50









MORE INFO AND DOCUMENTATION

52









This is a guide of how to compile and come up with your own executable/binary of Suricata on Windows. If you do not want to do that – there is a auto installation (MSI) windows native package here:

http://www.openinfosecfoundation.org/index.php/download-suricata

just run it and it will install Suricata for you on your Windows system.

This guide consists of two sections.

Section I – is for advanced users, a quick overview of what needs to be done. If you have had experience with Cygwin and Suricata before, this section should be enough. Should you feel you need a bit more details, please jump to Section II.

Section II – is a step by step guide and detailed instructions on how to install and configure Suricata on Windows OS, for newbies.

SECTION I - Advanced users

Step 1 – You need to download and install Cygwin Download Cygwin – <u>http://cygwin.com/setup.exe</u>

After the installation is done you would need to add the packages below to your Cygwin installation - needed for Suricata to run.









w32api, mpfr, pthreads, gcc-core , gcc4-core , make , zlib , autoconf , automake , libtool , glib , pkg-config , pkg-config , git .

Or in a bit more detail:

earch	Clear					🔘 Keep 🔘 Prev 💿 Curr 💿 Exp 🛛 Wew Pending
Current	New	Bin?	Src?	Categories	Size	Package
	10-1	\times		Devel	4k	autoconf: Wrapper scripts for autoconf commands
	2.68-1	\times		Devel	955k	autoconf2.5: Development version of the automatic configure script builder
	1.9.6-10	\times		Devel	556k	automake 1.9: (1.9) a tool for generating GNU-compliant Makefiles
	3.4.4-999	\times		Devel	3,630k	gcc-core: C compiler
	4.5.3-3	\times		Devel	10,508k	gcc4-core: Core C compiler subpackage
	4.5.3-3	\times		Devel	6,896k	gcc4-objc: Objective C and Objective C++ subpackage
	1.7.5.1-1	\times		Devel	2,725k	git: Fast Version Control System - core files
	0 2.22.4-2	\times		Gnome, Libs	1,734k	libglib2.0-devel: GNOME C function library (development)
	0 2.24.1-1	\times		Gnome, Libs	1,306k	libglib2.0_0: GNOME C function library (runtime)
	3.0.1-1	\times		Libs, Math	20k	libmpfr-devel: A library for multiple-precision floating-point arithmetic with exact rounding
	3.0.1-1	\times		Libs, Math	354k	libmpfr4: A library for multiple-precision floating-point arithmetic with exact rounding
	0.2-1	\times		Libs	3k	libpthread-stubs: Pthread stubs pkg-config metadata
	0 2.4-1	\times		Devel	768k	libtool: A shared library generation tool
	3.81-2	\times		Devel	350k	make: The GNU version of the 'make' utility
	3.0.1-1	\times		Libs, Math	70k	mpfr: A library for multiple-precision floating-point arithmetic with exact rounding
	0.23b-10	\times		Devel	68k	pkg-config: A utility used to retrieve information about installed libraries
	3.17-2	\times		Libs	1,274k	w32api: Win32 API header and library import files
	€ 1.2.5-1	\boxtimes		Devel, Libs	69k	zlib-devel: The zlib compression/decompression library (development)

Step 2 – Make sure you have WinPcap installed. http://www.winpcap.org/install/default.htm

You would also need to download and unzip (anywhere you like) the devs pkgs of WinPcap

http://www.winpcap.org/devel.htm

Copy libraries (from the unpacked directory) like this:









- ✓ Copy all the content of WpdPack\Lib\ to cygwin\lib\
- ✓ Copy all headers (all the content)from WpdPack\Include\ to C:\cygwin\usr\include\
- ✓ Rename "libwpcap" to "libpcap" (in your cygwin\lib\ directory)

Step 3 Add some paths to the system path.

Add to system path (Win 7, 2008 - Control Panel\System and Security\System\Advanced system settings\Environment Variables), select "path" under "system variables", click "edit", append the following to the end:

C:\cygwin\bin;C:\cygwin\lib\pkgconfig;

Add the above to environment system variables in your windows system!!

Step 4 Get libyaml

Download the yaml package (at the time of this writing the current version is yaml-0.1.4.tar.gz)

http://pyyaml.org/download/libyaml/yaml-0.1.4.tar.gz

Unpack it in (for example in your Cygwin tmp directory) - C:\cygwin\tmp

Start Cygwin, go to the yaml directory then execute -

./configure --prefix=/usr && make && make install

Step 5 Get and compile Suricata.

As you are still in the CYGWIN environment -









Suricata from git: If you want to install Suricata from git – <u>latest version</u>

go to a tmp dir. Type in :

- a) git clone git://phalanx.openinfosecfoundation.org/oisf.git
- b) cd oisf
- c) dos2unix.exe libhtp/configure.ac && dos2unix.exe libhtp/htp.pc.in && dos2unix.exe libhtp/Makefile.am
- d) ./autogen.sh && ./configure && make

Suricata Stable (at the moment of this writing the stable version is 1.2.1): If you want to install Suricata stable – <u>latest stable version (production)</u>

(You can find it here - http://www.openinfosecfoundation.org/index.php/download-suricata)

go to a tmp dir. Type in :

- a) wget http://www.openinfosecfoundation.org/download/suricata-1.2.1.tar.gz
- b) tar –zxf suricata-1.2.1.tar.gz
- c) cd suricata-1.2.1
- d) dos2unix.exe libhtp/configure.ac && dos2unix.exe libhtp/htp.pc.in && dos2unix.exe libhtp/Makefile.am
- e) libtoolize -c && autoreconf -fv --install && ./configure && make









Ste 5.1 – MD5 support

OPTIONALLY – if you would like to compile Suricata with MD5s support ()to be able to log MD5s on opened/downloaded/transferred files coming through the wire- you must compile like this –

Make sure you add the following for Cygwin:



Then -

For stable -

libtoolize -c && autoreconf -fv --install && ./configure --with-libnss-libraries=/usr/lib --withlibnss-includes=/usr/include/nss/ --with-libnspr-libraries=/usr/lib --with-libnsprincludes=/usr/include/nspr && make

For git -

./autogen.sh && ./configure --with-libnss-libraries=/usr/lib --with-libnssincludes=/usr/include/nss/ --with-libnspr-libraries=/usr/lib --with-libnsprincludes=/usr/include/nspr && make

Make sure you add the following DLLs to the directory where you will run suricata (copy them from the Cygwin\bin directory):

cygfreebl3.dll









cygnspr4.dll

cygnss3.dll

cygnssckbi.dll

cygnssdbm3.dll

cygnssutil3.dll

cygplc4.dll

cygplds4.dll

cygsmime3.dll

cygsoftokn3.dll

cygssl3.dll

Then continue with the instructions below (the bellow mentioned DLLs are also needed), just substitute the **oisf** directory with **suricata-1.2.1** directory!

After it is done, go to your /oisf/src/.lib (or /suricata-1.2.1/src/.lib for Suricata stable) directory and copy the Suricata.exe file to a dedicated directory, for example –C:\Suricata

Also - copy classification.config , reference.config and suricata.yaml (form your oisf/ directory) to (your dedicated directory) C:\Suricata

NOTE: If you would like to make a standalone installation, copy (from C:\cygwin\bin)

cygz.dll

cygwin1.dll

cygpcre-0.dll

cygmagic-1.dll







cyggcc_s-1.dll

cygnspr4.dll

cygnss3.dll

to your C:\Suricata directory

Also copy C:\cygwin\usr\share\misc\magic.mgc to your C:\Suricata directory

Step 6 Configure and run Suricata ...run intruders...run...

Download some rule sets and copy them to your rules directory.

Edit your suricata.yaml - (for example, at least change these lines and create the necessary folders respectively):

"default-log-dir: C:\Suricata\log

•••••

- file:

enabled: yes

filename: C:\Suricata\suricata.log

.....

default-rule-path: C:\Suricata\rules\









classification-file: C:\Suricata\classification.config"

Open a cmd.

cd to your Suricata directory , execute -

suricata.exe -c suricata.yaml -i 192.168.1.71

change 192.168.1.71 with your respective IP and you are done.

NOTE:

If you need to run Suricata on an un-ip'd interfaces(thanks to Rich Rumble for pointing that out):

You can get the NIC UUID in a variety of ways, the simplest is using a single command for WMIC: (from cmd prompt paste in the following)

wmic nicconfig get ipaddress,SettingID

If you know your NIC's IP you can filter the results with findstr:

wmic nicconfig get ipaddress,SettingID | findstr 1.2.3.4

(replace 1.2.3.4 with your NIC's IP)

Then use that as your interface argument:

suricata.exe -c suricata.yaml –i <u>\\DEVICE\\NPF \{EE7B2A76-9343-449F-B3D8-3CB0F37DCA49\}</u> Make sure the double slashes are used, and a backslash is placed before the curly braces!

















SECTION II - Step By Step for newbie

The following installations instructions were executed on Windows Server 2008R2 64 bit. About 500 -600MB of space needed in total with all the necessary prerequisites installed.

Step 1 Download Cygwin http://cygwin.com/setup.exe

Then double click the stup.exe to install

Go ahead and install it with the default options (basically just click next and ok)















Choose A Do	wnload Source
Choose wh a local dire	ether to install or download from the internet, or install from files in ctory.
	 Install from Internet (downloaded files will be kept for future re-use)
	Ownload Without Installing
	Install from Local Directory









i	elect the directory where you want to install Cygwin. Also choose a few stallation parameters.
Roo	t Directory
C	cygwin Browse
Inst	all For
0	All Users (RECOMMENDED)
	Cygwin will be available to all users of the system.
0	Just Me
	Cygwin will still be available to all users, but Desktop Icons, Cygwin Menu Entries, and important Installer information are only available to the current user. Only select this if you lack Administrator privileges or if you have specific needs.









Cygwin Setup - Select Local Pa		
Select Local Package Direct Select a directory where you w downloads. The directory will	t ory want Setup to store the installation files it be created if it does not already exist.	E
Local Package Directory		
E:\work\Downloads		Browse







e ct Your Ini Setup needs	ternet Connection to know how you want it to connect to the internet. Choose
he appropria	te settings below.
(Direct Connection
(Use Internet Explorer Proxy Settings
0	Use HTTP/FTP Proxy:
	Proxy Host
	Pat 00
	FOIL

Here , select any mirror you want:









Choose A Do Choose a s	wnload Site te from this list, or add your own sites to the list	E
	Available Download Sites:	
	http://mirrors.163.com http://cygwin.mirrors.hoobly.com ftp://cygwin.mirrors.pair.com http://cygwin.mirrors.pair.com http://cygwin.parentingamerica.com http://cygwin.skazkaforyou.com http://cygwin.skazkaforyou.com http://mirrors.ymnds.com http://tweedo.com ftp://tweedo.com ftp://mirrors.xmission.com http://lug.mtu.edu http://lug.mtu.edu http://lug.mtu.edu	
User URL:		Add

Pic6

Then you are going to see a progress bar:









Progress This page displays the progress of the download or installation.				
Downloading setup.bz2 fro	 http://mirrors.163.com/cvawin/			
11 % (30k/2	′4k) 0.5 k B/s			
Progress:				

You might get a warning if you already have installed CYGWIN – this is a guide for an installation from scratch :









Setup Alert		23
8	This is the first time you've installed Cygwin 1.7.x. Please be advised that this is a major release. If you have not done so already, please check out the documentation at http://cygwin.com/ to see how the upgrade could potentially affect any existing Cygwin installation.	N
	If this is the first time you've installed Cygwin on this system then you can ignore this message.	
	OK	:el

Click next:







Then CYGWIN will start downloading and installing the necessary packages:







Progress This pag	ge displays the p	progress of the do	wnload or installat	ion.	C
	Downloading				
	bash-4.1.10-4.ta	ar.bz2 from http://	cygwin.mirrors.hoo	obly.com//re	
	Connecting				
	Package:				
	Total:				
	Disk:				

It will probably take 10-15 min.

After it is finished - click finish 🙂 :









Create Icons Tell setup if you want it to create a few icons for convenient access to the Cygwin environment.				
cygwin chvironinent.				
	Create icon on Desktop			
	Add icon to Start Menu			
Installation Status Installation Complete				
	< Back Finish Cancel			

Pic11

Step 2 Install extra packages

Go back and double-click the very same setup.exe – we will need to install the extra packages necessary for Suricata to run.

Click next and ok until you are presented with the following screen:





Cygwin S	Setup - Select P	ackages	**	CO MICO	MBC N	SURI
Select P	ackages	stall.				٦
Selec	a packages to in:					
Catalon	Carret	New	D C C	C Keep O Pre	volum ob	cp View Category
	y Current	New	B S Size	Раскаде		
	operault Andreadibility 🗘 Def	ing de				
	min O Default	duit				
	nin Ə Default					
	dio 🖸 Default					
E Ra	se \Lambda Default					
E Da	itabase 📭 Defau	de .				
E De	vel 🕀 Default					
E Do	c 🕂 Default					L
E Ed	itors 👀 Default					
🕀 Ga	mes 🛈 Default					
🕀 Gn	ome 😯 Default					
⊞ Gra	aphics 📀 Defaul	t				
⊞ Int	erpreters 😯 Defa	ault				
E KD	E 🕄 Default					
🕀 Lib	s 😯 Default					
F M=	ail \Lambda Dafault					
•		III				4

Pic12

Here (Pic below) is where we search select and queue for installation the additional packages needed.

In the picture below , in the search box type in the name of the package- the search will return automatically , results , select the necessary package. Erase the contentment of the search box and type in the name of the next package, select ...

Do the same for all the needed packages, DO NOT hit next until you have selected all the packages.









elect Packages Select packages to insta	all		
earch autoconf	Clear		○ Keep ○ Prev
Category Current	New	B S Size	Package
All Detault			
	(10.1)		4k autoconf Wrapper scripts for autoconf commands
	Skip	ηία ηία	200k autoconf2.1: Stable version of the automatic configure script buik
	2.68-1		955k autoconf2.5 Development version of the automatic configure scr
	Skip	nja nja	426k gcc-tools-epoch1-autoconf: (gcc-special) automatic configure scr
	Skip	nja nja	713k gcc-tools-epoch2-autoconf: (gcc-special) automatic configure scr
1	III		•
Hide obsolete packages			

Pic13

The necessary packages are:





gwin Setup	- Select Packages	-				
elect Packa Select pack	ges ages to install					
earoh	Clear					© Keep ⊙ Prev
Current	New	Bin?	Src?	Categories	Size	Package
	10-1	\times		Devel	4k	autoconf: Wrapper scripts for autoconf commands
	2.68-1	\times		Devel	955k	autoconf2.5: Development version of the automatic configure script builder
	1.9.6-10	\times		Devel	556k	automake 1.9: (1.9) a tool for generating GNU-compliant Makefiles
	3.4.4-999	\times		Devel	3,630k	gcc-core: C compiler
	4.5.3-3	\times		Devel	10,508k	gcc4-core: Core C compiler subpackage
	4.5.3-3	\times		Devel	6,896k	gcc4-objc: Objective C and Objective C++ subpackage
	O 1.7.5.1-1	\times		Devel	2,725k	git: Fast Version Control System - core files
	0 2.22.4-2	\times		Gnome, Libs	1,734k	libglib2.0-devel: GNOME C function library (development)
	0 2.24.1-1	\times		Gnome, Libs	1,306k	libglib2.0_0: GNOME C function library (runtime)
	3.0.1-1	\times		Libs, Math	20k	libmpfr-devel: A library for multiple-precision floating-point arithmetic with exact rounding
	3.0.1-1	\times		Libs, Math	354k	libmpfr4: A library for multiple-precision floating-point arithmetic with exact rounding
	0.2-1	\times		Libs	3k	libpthread-stubs: Pthread stubs pkg-config metadata
	2.4-1	\times		Devel	768k	libtool: A shared library generation tool
	3.81-2	\times		Devel	350k	make: The GNU version of the 'make' utility
	3.0.1-1	\times		Libs, Math	70k	mpfr: A library for multiple-precision floating-point arithmetic with exact rounding
	0.23b-10	\times		Devel	68k	pkg-config: A utility used to retrieve information about installed libraries
	3.17-2	\times		Libs	1,274k	w32api: Win32 API header and library import files
	♦ 1.2.5-1	\boxtimes		Devel, Libs	69k	zlib-devel: The zlib compression/decompression library (development)

Pic14

After you are done selecting the packages - make sure the "search" box is cleared, click the "view" button until the text on the right of the button displays "pending".

Check and make sure all the needed packages are selected! If something is missing, go back and select it!

Click Next.

ĮL

After that click next (make sure the option "select required packages (RECOMMENDED)" is selected!):









Resolving Dependencies The following packages are required to sa	atisfy dependencies.
autoconf2.1 (2.13-10) Stable version of the auto Required by: autoconf, au	matic configure script builder
automake (4-10) Wrapper scripts for autom Required by: libtool	nake and aclocal
automake1.10 (1.10.3-1) (1.10) a tool for generating Required by: automake	g GNU-compliant Makefiles
III Select required packages (RECOMMEND)	ED)

Pic15

The extra packages that you have selected will start to download and install:









D					-
This p	age displays th	ne progress of the downloa	ad or installation.		E
	Downloading]			
	autoconf2.5-	2.68-1.tar.bz2 from http://	cygwin.mirrors.hoobly.c	:0	
	29 % (286k/	′977k) 113.5 kB/s			
	Package:				
	Total:				
	Disk:				

Pic16

This could also take 5 min or so.

Then click finish:









Create Icons Tell setup if you want it	to create a few icons for co	nvenient access to the	E
Cygwin environment.			
	Create icon on <u>D</u> esk	ctop	
	Add icon to Start Me	enu	
Installation Status			
installation complete			
		< <u>B</u> ack Finish	Cancel



Step 3 Add paths to system variables

Add path to system variables (Win 7, 2008 - Control Panel\System and Security\System\Advanced system settings\Environment Variables) :

C:\cygwin\bin;C:\cygwin\lib\pkgconfig;

Add the above to environment system variables in your windows system!!

See the picture below









System Properties					x
Computer Name	Hardware	Advanced	System Protection	Remote	
You must be log	gged on as a	an Administrat	tor to make most of th	hese chan	ges.
Visual effects,	, processor s	cheduling, m	emory usage, and vir	tual memo	ry
				Settings	
User Profiles					— II
Desktop settir	ngs related to	o your logon			
				Settings	
Startup and R	ecovery				— II
System startup	o, system fai	lure, and deb	ugging information		
				Settings	
			Environme	ent Variable	es
		ОК	Cancel		pply

Pic18

Edit the system path variable:









		x
User variables for Dor	nPedro	
Variable	Value	
PATH	C:\Program Files (x86)\Nmap;C:\win32\	=
PKG_CONFIG_P	/win32/lib/pkgconfig	
TEMP	%USERPROFILE%\AppData\Local\Temp	
TMP	%USERPROFILE%\AppData\Local\Temp	Ψ.
[New Edit Delete	
Variable	Value	•
Variable OS	Value Windows_NT	•
Variable OS Path	Value Windows_NT C:\Program Files (x86)\NVIDIA Corpora	•
Variable OS Path PATHEXT	Value Windows_NT C:\Program Files (x86)\NVIDIA Corpora .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;	•
Variable OS Path PATHEXT PROCESSOR_A	Value Windows_NT C:\Program Files (x86)\NVIDIA Corpora .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS; AMD64	•
Variable OS Path PATHEXT PROCESSOR_A	Value Windows_NT C:\Program Files (x86)\NVIDIA Corpora .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS; AMD64 New Edit Delete	•

Add "C:\cygwin\bin;C:\cygwin\lib\pkgconfig; " without the quotes to the end of the "Variable value path ":

Edit System Variab	le X
Variable name:	Path
Variable value:	<pre>bin;C:\cygwin\bin;C:\cygwin\ib\pkgconfig;</pre>
	OK Cancel









Step 4 Get libyaml Go to <u>http://pyyaml.org/wiki/LibYAML</u>

Download the yaml package (at the time of this writing the current version is yaml-0.1.4.tar.gz)

http://pyyaml.org/download/libyaml/yaml-0.1.4.tar.gz.

Unpack it in :

C:\cygwin\tmp

After you have unpacked it you should have the following directory:

C:\cygwin\tmp\yaml-0.1.4

Step 5 Get libpcap – for windows Go to

http://www.winpcap.org/install/default.htm

and download the WinPcap installer for windows (at the time of this writing the current version was 4.1.2)

Install the WinPcap (double click, and just use the default options, basically click next and ok until finished.)

After that is done go to

http://www.winpcap.org/devel.htm









This is IMPORTANT, this is the development pack, we need that for Suricata to be able to run on Windows.

Download the package

Unpack it anywhere you like.

Copy libraries (from the unpacked directory) like this:

- ✓ Copy ALL the content of WpdPack\Lib\ to cygwin\lib\
- ✓ Copy all headers (all the content)from WpdPack\Include\ to C:\cygwin\usr\include\
- ✓ Rename "libwpcap" to "libpcap" (in your cygwin\lib\ directory)

Step 6 Start Cygwin and compile yaml Open CYGWIN

Double click your CYGWIN icon on your desktop.

A Linux/bash like command prompt will open:











Type the following commands as shown in the picture below (hit enter after each command):











Basically the commands are :

cd /tmp

cd yaml-0.1.4

./configure --prefix=/usr && make && make install

The last command above is on one line.









This will configure and install the yaml package that we need for Suricata, let it finish:



Pic23

After it is done, it should be something like this:







E /tmp/yaml-0.1.4	
<pre>If you ever happen to want to link against installed libraries in a given directory, LIBDIR, you must either use libtool, and specify the full pathname of the library, or use the `-LLIBDIR' flag during linking and do at least one of the following: - add LIBDIR to the `PATH' environment variable during execution - add LIBDIR to the `LD_RUN_PATH' environment variable during linking - use the `-LLIBDIR' linker flag</pre>	
see any operating system documentation about shared libraries for more information, such as the ld(1) and ld.so(8) manual pages.	
<pre>make[2]: Nothing to be done for `install-data-am'. make[1]: Leaving directory '/tmp/yaml-0.1.4/src' Making install in . make[1]: Entering directory '/tmp/yaml-0.1.4' make[2]: Entering directory '/tmp/yaml-0.1.4' make[2]: Nothing to be done for `install-exec-am'. test -z "/usr/lib/pkgconfig" !! /usr/bin/mkdir -p "/usr/lib/pkgconfig' /usr/bin/install -c -m 644 yaml-0.1.pc '/usr/lib/pkgconfig' make[2]: Leaving directory '/tmp/yaml-0.1.4' make[1]: Leaving directory '/tmp/yaml-0.1.4' Making install in tests make[1]: Entering directory '/tmp/yaml-0.1.4/ Making install in tests make[2]: Nothing to be done for `install-exec-am'. make[2]: Nothing to be done for `install-exec-am'. make[2]: Nothing to be done for `install-exec-am'. make[2]: Leaving directory '/tmp/yaml-0.1.4/tests' make[2]: Entering directory '/tmp/yaml-0.1.4/tests' make[2]: Nothing to be done for `install-exec-am'. make[2]: Leaving directory '/tmp/yaml-0.1.4/tests' Making install in win32 make[1]: Entering directory '/tmp/yaml-0.1.4/tests' Making install in win32 make[2]: Nothing to be done for `install-exec-am'. make[2]: Leaving directory `/tmp/yaml-0.1.4/win32' make[2]: Leaving directory `/tmp/yaml-0.1.4/win32' make[2]: Leaving directory `/tmp/yaml-0.1.4/win32' make[1]: Leaving directory `/tmp/yam</pre>	,
\$	Ť

Pic24

Then go up one directory.

" cd .. "









Step 7 Compile Suricata

Step 7.1 – Suricata from git – <u>latest version</u>

(Step 7.2 – follows just after (7.1) – Suricata stable – <u>latest stable release</u>, if you would like to use the stable version, please go to 7.2)

Get and compile Suricata.

As you are still in the CYGWIN environment -

Type in

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

Then after it is done

cd oisf

like so:









🔄 /tmp/oisf	×	J
make[1]: Leaving directory `/tmp/yaml-0.1.4/src' Making install in . make[1]: Entering directory `/tmp/yaml-0.1.4' make[2]: Entering directory `/tmp/yaml-0.1.4' make[2]: Nothing to be done for `install-exec-am'. test -z ''/usr/lib/pkgconfig'' !! /usr/bin/mkdir -p ''/usr/lib/pkgconfig'' /usr/bin/install -c -m 644 yaml-0.1.pc '/usr/lib/pkgconfig' make[2]: Leaving directory `/tmp/yaml-0.1.4' make[1]: Leaving directory `/tmp/yaml-0.1.4' Making install in tests		
<pre>make[1]: Entering directory `/tmp/yaml-0.1.4/tests' make[2]: Entering directory `/tmp/yaml-0.1.4/tests' make[2]: Nothing to be done for `install-exec-am'. make[2]: Nothing to be done for `install-data-am'. make[2]: Leaving directory `/tmp/yaml-0.1.4/tests' make[1]: Leaving directory `/tmp/yaml-0.1.4/tests' Making install in win32 make[1]: Entering directory `/tmp/yaml-0.1.4/win32' make[2]: Entering directory `/tmp/yaml-0.1.4/win32' make[2]: Nothing to be done for `install-exec-am'. make[2]: Nothing to be done for `install-data-am'. make[2]: Nothing to be done for `install-data-am'. make[2]: Nothing to be done for `install-data-am'. make[2]: Leaving directory `/tmp/yaml-0.1.4/win32' make[1]: Leaving directory `/tmp/yaml-0.1.4/win32' </pre>		
DonPedro@DonPedro-PC /tmp \$ git clone git://phalanx.openinfosecfoundation.org/oisf.git Cloning into oisf remote: Counting objects: 18733, done. remote: Compressing objects: 100% (11632/11632), done. remote: Total 18733 (delta 15428), reused 8746 (delta 7071) Receiving objects: 100% (18733/18733), 5.59 MiB ¦ 362 KiB/s, done. Resolving deltas: 100% (15428/15428), done.		
DonPedro@DonPedro-PC /tmp \$ cd oisf/ DonPedro@DonPedro-PC /tmp/oisf \$	Ŧ	

Pic25

After that we execute the following (one line):

dos2unix.exe libhtp/configure.ac && dos2unix.exe libhtp/htp.pc.in && dos2unix.exe libhtp/Makefile.am

Like so:









🔄 /tmp/oisf	<u>}</u>	
<pre>make[2]: Leaving directory '/tmp/yaml-0.1.4' make[1]: Leaving directory '/tmp/yaml-0.1.4' Making install in tests make[1]: Entering directory '/tmp/yaml-0.1.4/tests' make[2]: Entering directory '/tmp/yaml-0.1.4/tests' make[2]: Nothing to be done for 'install-exec-am'. make[2]: Nothing to be done for 'install-data-am'. make[1]: Leaving directory '/tmp/yaml-0.1.4/tests' Making install in win32 make[1]: Entering directory '/tmp/yaml-0.1.4/win32' make[2]: Nothing to be done for 'install-exec-am'. make[2]: Nothing to be done for 'install-o.1.4/win32' make[2]: Nothing to be done for 'install-exec-am'. make[2]: Nothing to be done for 'install-exec-am'. make[2]: Entering directory '/tmp/yaml-0.1.4/win32' make[2]: Nothing to be done for 'install-data-am'. make[2]: Nothing to be done for 'install-data-am'. make[2]: Leaving directory '/tmp/yaml-0.1.4/win32' make[1]: Leaving directory '/tmp/yaml-0.1.4/win32'</pre>		
DonPedro@DonPedro-PC /tmp/yaml-0.1.4 \$ cd		
DonPedro@DonPedro-PC /tmp \$ git clone git://phalanx.openinfosecfoundation.org/oisf.git Cloning into oisf remote: Counting objects: 18733, done. remote: Compressing objects: 100% (11632/11632), done. remote: Total 18733 (delta 15428), reused 8746 (delta 7071) Receiving objects: 100% (18733/18733), 5.59 MiB ¦ 362 KiB/s, done. Resolving deltas: 100% (15428/15428), done.		
DonPedro@DonPedro-PC /tmp \$ cd oisf/		
DonPedro@DonPedro-PC /tmp/oisf \$ dos2unix.exe libhtp/configure.ac && dos2unix.exe libhtp/htp.pc.in && dos2unix .exe libhtp/Makefile.am dos2unix: converting file libhtp/configure.ac to Unix format dos2unix: converting file libhtp/htp.pc.in to Unix format dos2unix: converting file libhtp/Makefile.am to Unix format		
DonPedro@DonPedro-PC /tmp/oisf	-	

Then we execute the following command:

./autogen.sh && ./configure && make

The above command is on one line

That will start configuration and compilation of Suricata.

Like so:









E /tmp/oisf	٢.
remote: Total 18733 (delta 15428), reused 8746 (delta 7071) Receiving objects: 100% (18733/18733), 5.59 MiB ¦ 362 KiB/s, done. Resolving deltas: 100% (15428/15428), done.	^
DonPedro@DonPedro-PC /tmp \$ cd oisf/	
<pre>DonPedro@DonPedro-PC /tmp/oisf \$ dos2unix.exe libhtp/configure.ac && dos2unix.exe libhtp/htp.pc.in && dos2unix .exe libhtp/Makefile.am dos2unix: converting file libhtp/configure.ac to Unix format dos2unix: converting file libhtp/htp.pc.in to Unix format dos2unix: converting file libhtp/Makefile.am to Unix format</pre>	
<pre>DonPedro@DonPedro-PC /tmp/oisf \$./autogen.sh && ./configure && make Found libtoolize libtoolize: putting auxiliary files in `.'. libtoolize: copying file `./ltmain.sh' libtoolize: copying file `m4/libtool.m4' libtoolize: copying file `m4/ltoytions.m4' libtoolize: consider adding `AC_CONFIG_MACRO_DIR([m4])' to configure.in and libtoolize: rerunning libtoolize, to keep the correct libtool macros in-tree. autoreconf-2.68: Entering directory `.' autoreconf-2.68: configure.in: not using Gettext autoreconf-2.68: configure.in: tracing autoreconf-2.68: configure.in: tracing autoreconf-2.68: configure.ac: not using Gettext autoreconf-2.68: running: aclocalforce autoreconf-2.68: running: libtoolizecopyforce libtoolize: putting macros in AC_CONFIG_MACRO_DIR, `m4'. libtoolize: copying file `m4/libtool.m4'</pre>	
	-

Let it run.....this could take 10 min. or so

After it is done:









🔄 /tmp/oisf	
detect-byte-extract.o detect-replace.o util-print.o util-fmemopen.o util-cpu.o util-pidfile.o util-mpm.o util-spm.o util-spm-bs.o util-spm-bs2bm.o util-spm-bm. o util-mpm-wumanber.o util-mpm-b2g.o util-mpm-b2g-cuda.o util-mpm-b3g.o util-mpm- b2gc.o util-unittest-helper.o util-mpm-ac.ogfbs.o util-cidr.o util-unitte st.o util-unittest-helper.o util-hash.o util-hashlist.o util-bloomfilter.o util- bloomfilter-counting.o util-pool.o util-time.o util-var.o util-var-name.o util- bloomfilter-counting.o util-rule-vars.o util-fix_checksum.o util-daemon.o util-r andom.o util-classification-config.o util-checksum.o util-daemon.o util- random.o util-strlcatu.o util-strlcyu.o util-cuda.o util-cuda-handlers.o util-proto -name.o util-devoce.o util-device.o util-fixity.o util-memcmp.o util-proto -name.o util-syslog.o util-device.o util-checksum.o util-memcmp.o util-proto -name.o tm-gueuehandlers.o tm-threads.o tmgh-simple.o tmgh-ngk.o tmedh-packe tpool.o tmgh-flow.o tmgh-ringbuffer.o alert-fastlog.o alert-debuglog.o alert-pre lude.o alert-unified2-alert.o alert-syslog.o alert-pcapinfo.o log-droplog.o log- thtplog.o log-pcap.o stream.o stream-tcp.o stream-tcp-reassemble.o stream-tcp- app-layer-garker.o app-layer-detect-proto.o app-layer-ssh.o app-layer-protos.o app-layer.o app-layer-detect-proto.o app-layer-ssh.o app-layer-smtp.o defrag.o outpl.o win32-misc.o win32-service.o util-action.o util-profiling.o cuda-packet-batcher.o util-ioctl.o/libhtp/ht p/.libs/libhtp.a -lz -lpcap -lpthread /usr/lib/libyaml.a /usr/lib/libpcre.dll.a make[2]: Leaving directory `/tmp/oisf/src' Making all in qa make[2]: Entering directory `/tmp/oisf/ga'	
Making all in coccinelle make[3]: Entering directory '/tmp/oisf/qa/coccinelle' make[3]: Nothing to be done for `all'. make[3]: Leaving directory '/tmp/oisf/qa' make[3]: Entering directory '/tmp/oisf/qa' make[3]: Nothing to be done for `all-am'. make[3]: Leaving directory `/tmp/oisf/qa' make[2]: Leaving directory `/tmp/oisf/qa' make[2]: Entering directory `/tmp/oisf' make[2]: Leaving directory `/tmp/oisf' make[1]: Leaving directory `/tmp/oisf'	-

Pic28

Step 7.2 Suricata stable

Get and compile Suricata.

As you are still in the CYGWIN environment -

Suricata Stable (at the moment of this writing the stable version is 1.2.1):

If you want to install Suricata stable – <u>latest stable version (production)</u>









(You can find it here - http://www.openinfosecfoundation.org/index.php/download-suricata)

go to a tmp dir. Type in (if you do not have "wget" installed - go ahead and install it the very same way you searched and added/installed the other pkgs to Cygwin) :

- f) wget http://www.openinfosecfoundation.org/download/suricata-1.2.1.tar.gz
- g) tar –zxf suricata-1.2.1.tar.gz
- h) cd suricata-1.2.1
- i) dos2unix.exe libhtp/configure.ac && dos2unix.exe libhtp/htp.pc.in && dos2unix.exe libhtp/Makefile.am
- j) libtoolize -c && autoreconf -fv --install && ./configure && make











(--enable-debug and --enable-profiling are optional, you do not have to add them, I just add them because I like them, (pic above))

Then continue with the instructions below, just substitute the **oisf** directory with **suricata-1.2.1** directory!

Step 8 Set up Suricata for Windows

Create a directory C:\Suricata – you can use Win Explorer, you don't need to make it from Cygwin.









Then copy the Suricata.exe file from C:\cygwin\tmp\oisf\src\.libs

То

C:\Suricata

Create a directory C:\Suricata\log

Then create a directory C:\Suricata\rules

This will hold the rule files for Suricata.

Go to http://rules.emergingthreats.net/open/suricata/

Download a rule set.

http://rules.emergingthreats.net/open/suricata/emerging.rules.tar.gz

Or download rules from <u>http://www.snort.org/</u>, whichever you like, however Emerging Threats are developing rules especially for Suricata, in order to use its capabilities to a full extend.

Unzip/untar the rule set in the C:\Suricata\rules directory.

Then go to C:\cygwin\tmp\oisf

Сору

classification.config, reference.config and suricata.yaml to

C:\Suricata

Open suricata.yaml with an editor – Notepad, Notepad++, whichever you like.

Change the following lines:

default-log-dir: C:\Suricata\log









•••••

- file:

enabled: yes

filename: C:\Suricata\suricata.log

.....

default-rule-path: C:\Suricata\rules\

classification-file: C:\Suricata\classification.config

HOME_NET: "[192.168.0.0/16]" - (here actually you put any network you want Suricata to inspect)

Like shown on the pictures below:









📄 surica	sta yami						
22	# Preallocated size for packet. Default is 1514 which is the classical						
23	# size for pcap on ethernet. You should adjust this value to the highest						
24	<pre># packet size (MTU + hardware header) on your system.</pre>						
25	#default-packet-size: 1514						
26							
27	# Set the order of alerts bassed on actions						
28	# The default order is pass, drop, reject, alert						
29	action-order:						
30	- pass						
31	- drop						
32	- reject						
33	- alert						
34							
35							
36	# The default logging directory. Any log or output file will be						
37	# placed here if its not specified with a full path name. This can be						
38	# overridden with the -1 command line parameter.						
39 <	default-log-dir: C:\Suricata\log						
40							
41	# Configure the type of alert (and other) logging you would like.						
42	outputs:						
43							
44	# a line based alerts log similar to Snort's fast.log						
45	- fast:						
46	enabled: yes						
47	filename: fast.log						
48	append: yes						
49							
50	# log output for use with Barnyard						
51	- unified-log:						
52	enabled: no						
53	filename: unified.log						
54							
55	# Limit in MB.						
56	flimit: 32						
vormal te	ext file length : 23820 lines : 001 Lh : 39 Col : 33 Sel : 0 Dos/Windows ANSI as UTF-8 INS						









📄 suricata.yaml

461	÷							
462	# This value is overriden by the SC_LOG_FORMAT env var.							
463	#default-log-format: "[%i] %t - (%f.%l) <%d> (%n) "							
464								
465	# A regex to filter output. Can be overridden in an output section.							
466	# Defaults to empty (no filter).							
467	÷ · · · · · · · · · · · · · · · · · · ·							
468	# This value is overriden by the SC LOG OP FILTER env var.							
469	default-output-filter:							
470								
471	# Define your logging outputs. If none are defined, or they are all							
472	# disabled you will get the default - console output.							
473	outputs:							
474	- console:							
475	enabled: yes							
476	- file:							
477	enabled: yes							
478	filename: C:\Suricata\suricata.log							
479	- syslog:							
480	enabled: no							
481	facility: local5							
482	format: "[%i] <%d> "							
483								
484	84 # PF_RING configuration. for use with native PF_RING support							
485	<pre>85 # for more info see http://www.ntop.org/PF_RING.html</pre>							
486	pfring:							
487	# Number of receive threads (>1 will enable experimental flow pinned							
488	<pre># runmode)</pre>							
489	threads: 1							
490								
491	# Default interface we will listen on.							
492	interface: eth0							
493								
494	# Default clusterid. PF_RING will load balance packets based on flow.							
495	# All threads/processes that will participate need to have the same							
1 4 6 6								
Normal t	ext file length : 23826 lines : 661 Ln : 478 Col : 28 Sel : 11 Dos/Windows ANSI as UTF-8 INS							









😑 suricata.vaml

😑 sur	icata.yami						
514	ipfw:						
515							
516	# Reinject packets at	the specified ipfw rule number.	This config				
517	# option is the ipfw rule number AT WHICH rule processing continues						
518	# in the ipfw processing system after the engine has finished						
519	# inspecting the packet for acceptance. If no rule number is specified,						
520	# accepted packets are	reinjected at the divert rule w	which they entered				
521	# and IPFW rule proces	sing continues. No check is dor	ne to verify				
522	# this will rule makes	sense so care must be taken to	avoid loops in ipfw.				
523	+						
524	## The following examp	le tells the engine to reinject	packets				
525	# back into the ipfw f	irewall AT rule number 5500:					
526	+						
527	<pre># ipfw-reinjection-rul</pre>	e-number: 5500					
528							
529	# Set the default rule p	ath here to search for the files	з.				
530	# if not set, it will lo	ok at the current working dir					
531	default-rule-path: C:\Su	ricata\rules\					
532	rule-files:						
533	- emerging-current_even	ts.rules					
534							
535	Classification-file: C:\	Suricata\classification.config					
536	<pre>#reference-config-file:</pre>	/etc/suricata/reference.config					
537							
538	# Holds variables that w	ould be used by the engine.					
539	vars:						
540							
541	# Holds the address group vars that would be passed in a Signature.						
542	# These would be retrieved during the Signature address parsing stage.						
543	address-groups:						
544		0.0/1018					
545	HOME_NET: "[192.168.	J. U/ 16])					
546							
547	EXTERNAL_NET: any						
548		NETH					
	Seen conversi incorne			-			
Normal	text file	length : 23826 lines : 661	Ln:531 Col:31 Sel:11	Dos\Windows	ANSI as UTE-8	INS	

Pic31

Adjust your home network to whatever network you intend the Suricata to protect/inspect (as shown in the picture above).

Step 9 Runing Suricata

....run intruders run...

Open a cmd as ADMINISTRATOR!!!.









Got to C:\Suricata and execute

suricata.exe -c suricata.yaml -i 192.168.1.71

like shown on the picture below (in this case - 192.168.1.71 is the IP/interface I want Suricata to listen to, i.e. the IP that my network card has been configured with):

Administrator: C:\Windows\System32\cmd.exe				
C:\Suricata>				
C:\Suricata/				
C:\Sumicata>				
C:\Sumicata>				
C:\Suricata>				
G:\Suricata>				
G:\Suricata/				
C.\Sumicata)				
C:\Supicata>				
C:\Suricata>				
G:\Suricata>	- 400 460	4 114		
C:\Suricata/suricata.exe -c suricata.yami	-1 172.168.	1.71		



And you have yourself Suricata running (the start time could depend the PC/Server CPU/MEM availability and of course how many rules do you load, but it is max about 1.5 min):









📷 Administrator: C:\Windows\System32\cmd.exe - suricata.exe -c suricata.yaml -i 192.168.1.71	
filename: http.log	
[4452] 6/11/2011 - 19:06:14 - (alert-debuglog.c:542) <info> (AlertDebugLogInitCtx) Alert debug</info>	log output
initialized, filename: alert-debug.log	
14452] b/11/2011 19:06:14 - (alert-sysiog.c:170) (info) (Hiertsysioginittex) Sysiog output i 14452] b/11/2011 19:06:14 - (log-dwollog.c:170) (info) (logDwollogIpit(tx) Dwon log output i	itialized
filename: drop.log	icializea,
[4452] 6/11/2011 19:06:14 - (runmode-pcap.c:126) (Info) (ParsePcapConfig) Unable to find pcap	p config for
interface \Device\NFF_{DD'E8C68-52C7-439D-83P-19YABB22A849}, using default value	
100010/11/2011 - 17.00.14 - (Source-pcap.c.sio) (100) (Necesser) and (101) - using interaction of the second sec	: NDEVICENNF
[668] 6/11/2011 19:06:14 - (source-pcap.c:359) <info> (ReceivePcapThreadInit) Going to use po</info>	cap buffer s
14452] b/11/2011 19:05:14 - (Punmode-pcap.c:229) (Info) (KunmodelasPcapHuto) KunmodelasPcapHu ced	ito initiali
] [4452] 6/11/2011 19:06:14 - (stream-tcp.c:346) {Info} (StreamTcpInitConfig) stream "max_sess	ions": 26214
4	
[4452] 6/11/2011 19:06:14 - (stream-tcp.c:358) {Info} (StreamIcpInitConfig) stream "prealloc 09950	_sessions":
14452] 6/11/2011 19:06:14 - (stream-tcn.c:368) {[nfo] (StreamTcn]nitConfig) stream "memcan":	33554432
[4452] 6/11/2011 19:06:14 - (stream-tcp.c:374) <info> (StreamTcpInitConfig) stream "midstream"</info>	n" session p
14323 b/11/2011 - 17:06:14 - (stream-tcp.c:380) (info) (streamicpinitConfig) stream "async_one bled	eside": disa
[[4452]] 6/11/2011 19:06:14 - (stream-tcp.c:397) <info> (StreamTcpInitConfig) stream "checksum</info>	validation"
enabled	
[4452] b/11/2011 19:06:14 - (stream-tcp.c:407) (into) (stream[cpinitConfig) stream."inline": [4452] b/11/2011 19:06:14 - (stream-tcp.c:416) (lofo) (stream[cpinitConfig) stream."inline":	disabled
67108864	у мемсар -
[4452] 6/11/2011 19:06:14 - (stream-tcp.c:426) <info> (StreamTcpInitConfig) stream.reassembl</info>	y "depth": 1
(4452) 0/11/2011 17-06-14 - (stream-tcp.c-447) (into) (streamicpinitionfig) stream.reassembly) "toserver_
[4452] 6/11/2011 19:06:14 - (stream-tcp.c:451) (Info) (StreamTcpInitConfig) stream.reassembly	y "toclient_
chunk_size": 2560	
14452] b/11/2011 19:00:14 - (En-threads.c:1806) (Info) (ImihreadWaitOnihreadInit) all 16 paci- og thweads 3 management thweads initialized engine started	ket processi
ng chicaas, 5 Management chicaas inicializea, engine startea.	

That's it.

From here on it is up to you to configure Suricata the way it suits you best!

Thanks

More info and documentation

You can find much more info about setting up and tuning Suricata here:

https://redmine.openinfosecfoundation.org/projects/suricata/wiki











