SnortALog 2.3.0

SnortALog v2.3.0

Technical Documentation
http://jeremy.chartier.free.fr/snortalog/

Table Contents

1 Introduction

The purpose of this paper is to provide complete documentation for the installation, configuration and use of Snortalog.

This guide is most definitely not the end-all or the be-all, but it will tell you how to setup the program and to get it running in a relatively quick fashion.

2 Additional Information

If you have questions, comments, corrections, additions or whatever else please let me know. I can be reached via email at jeremy.chartier@free.fr and I like hearing people.

3 Licence

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

4 What is Snortalog

4.1 Overview

Snortalog is a powerfull perl program that summarizes Snort and Firewalls logs making it easy to view any attacks against your network.

Snortalog works with all versions of SNORT and is the only perl program which can analyse snort's logs in all formats (Syslog, Fast and Full alerts).

Also, it is able to summarize FW-1 (4.1 and NG), PIX, Netfilter and IPFilter logs in a similar way.

4.2 Possibilities

4.2.1 Main Possibilities

Available :

- Create HTML and text reports
- Can specify order (ascending or decscending)
- Can specify the number of occurences to view
- Can resolve IP addresses and domains
- Add colors for best visibility
- Graphic User Interface
- Mulit-language output
- Possibility to do filtering (e.g if you only want src logs) reference's rules
- Generate GIF, PNG or JPG graph in HTML output
- Possibility to generate PDF output on the fly
- Possibilitiy to use DBM

4.2.2 Snort Possibilities

- Works with Syslog, Fast and Full alerts
- Works with all preprocessor (spp_stream4, spp_portscan, spp_decoder, flow and flow-portscan ...)
- Has the possibility to link the signature to the web reference attack description
- Works with "-I" snort's option to specify an interface and add report
- Work now with "-e" Snort option (Display the second layer header info)
- Use a specific plugin to generate your owns reference's rules
- The ability to get Whois Database information

4.2.3 Logs compatibility

	Syslog	Other
Snort >= 2.1	OK	Snort Fast and Full alert
PIX	OK	
Fw-1 4.1	OK	Fw logexport
Fw-1 NG	OK	Fwm logexport
SmartDefense	OK	Fwm logexport
IPFilter	OK	
Netfilter	OK	

5 Why a Perl Program ?

There are several reasons why I choose to develop my program in perl.

I have been working with SNORT for 4 years and I couldn't find any existing scripts that were able to report on potential attacks quickly.

My first goal was to generate a text output (ASCII) to provide many sorting and filtering statistics. Eventually, I improved my program to generate charts (HTML) with graphics and a GUI.

You may ask why not use MySQL database or similar like ACID. As a member of SNORT's mailing list for a long time now, I often read questions about this error "Fatal error: Maximum execution time of 180 seconds exceeded".

You can regularly purge your database but this task could prove tough for the administrator. Moreover, in a network with a lot of NIDS and several thousand log alerts, a request to the database will have a long response time.

The use of a program like **Snortalog** is more easier, efficient and appropriate. Do your own tests and send me your feedback :))

6 Installation

6.1 Main installation

It's very easy to use Snortalog in standard mode (simple command line without graphics generation). The only things you should have is Perl 5.8 installed on your box.

Snortalog runs on many Operating Systems :

- Linux
- FreeBSD
- OpenBSD
- Solaris
- Windows
- MacOS

6.2 Graphic plugin installation

If you have decided to use Snortalog with any of its extended options, you will need to install some specific plugins not included as standard with Perl 5.8

Option	Plugin
• **-g :** Graphics generation	You will need to install : • gd-2.0.11.tar.gz (PNG and JPG format) or GD-1.19.tar.gz (GIF format) • GDGraph-1.39.tar.gz • GDTextUtil-0.85.tar.gz
• **-x :** Graphic User Interface	Modules and TK code for Perl/Tk :

	- Tk-800.024.tar.gz - perl-Tk-800.024-2.i386.rpm
- **-p :** PDF generation	You will need to install : - htmldoc-1.8.23-source.tar.gz - HTML-HTMLDoc-0.07.tar.gz
- **-w :** Whois Database information	You will need to install : - Net-Whois-IP-0.50.tar.gz

6.2.1 GD-1.19.tar.gz

```
# tar xzvf GD-1.19.tar.gz
# cd GD-1.19
#
# perl Makefile.PL
Checking if your kit is complete...
Looks good
MakeMaker (v6.03)
Writing Makefile for libgd
Writing Makefile for GD
#
# make
# make install
```

6.2.2 GDTextUtil-0.85

```
# tar xzvf GDTextUtil-0.85.tar.gz
# cd GDTextUtil-0.85
#
# perl Makefile.PL
Checking if your kit is complete...
Looks good
Writing Makefile for GD::Text
#
# make
# make install
```

6.2.3 GDGraph-1.39

```
# tar xzvf GDGraph-1.39.tar.gz
# cd GDGraph-1.39
#
# perl Makefile.PL
Checking if your kit is complete...
Looks good
Writing Makefile for GD::Graph

The automatic tests for GDGraph are not really a solid workout of the
library. The best way to test the package is to run the examples
before installing it.  You can run the examples in the samples
directory with `make samples` or by going into that directory, and
just running `make`.
If that fails, please read samples/Makefile.
```

```
#
# make
# make install
```

6.2.4 Gd-2.0.11

```
# tar xzvf gd-2.0.11.tar.gz
# cd gd-2.0.11
#
# ./configure
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking for gcc... gcc
checking for C compiler default output... a.out
checking whether the C compiler works... yes
...
checking for jpeg_set_defaults in -ljpeg... yes
checking for XpmReadFileToXpmImage in -lXpm... no

** Configuration summary for gd 2.0.11:

  Support for PNG library:          yes
  Support for JPEG library:         yes
  Support for Freetype 2.x library: no
  Support for Xpm library:          no

configure: creating ./config.status
config.status: creating Makefile
config.status: creating config/Makefile
config.status: creating config/gdlib-config
config.status: creating test/Makefile
config.status: creating config.h
config.status: executing depfiles commands
#
#make
#make install
```

6.2.5 HTMLDoc-1.8.23

```
# tar xzvf htmldoc-1.8.23-source.tar.gz
# cd htmldoc-1.8.23
#
# ./configure
# make
```

Please consult the HTMLDOC Users Manual http://www.easysw.com/htmldoc/documentation.html or the COMPILE.txt file for more information.

Current Limitations :

- No support for style sheets.
- No support for HTML forms.
- CAPTIONs are always shown at the top of the table.
- HTML 4.0 table elements and attributes are not supported (rules, THEAD, TFOOT, etc.).

6.2.6 HTML-HTMLDoc-0.07

```
# tar xzvf HTML-HTMLDoc-0.07.tar.gz
# cd HTML-HTMLDoc-0.07
#
# perl Makefile.PL
Checking if your kit is complete...
Looks good
Writing Makefile for HTML::HTMLDoc
#
# make
# make install
```

6.2.7 Net-Whois-IP-0.50

```
# tar xzvf Net-Whois-IP-0.50.tar.gz
# cd Net-Whois-IP-0.50
#
# perl Makefile.PL
Checking if your kit is complete...
Looks good
Writing Makefile for Net::Whois::IP
#
# make
# make install
```

7 Configuration

7.1 Variables

You need to specify the PATH of the PERL binary in the first line of the script as shown below. The path for the perl interpreter in your system can be found using "which perl " command at your shell.

```
# vi snortalog.pl

#!/usr/bin/perl
.
```

Here, the Snortalog initialization part :

```
#!/usr/bin/perl
#
# Jeremy Chartier, <jeremy.chartier@free.fr>
# Date: 2004/03/09
# Revision: 2.2.0
#

# User variables
# General Libraries - Never comment
use Getopt::Long;                # use Getopt for options
use Socket;                      # use socket for resolving domain name from IP
use Time::localtime;             # use for Time
use DB_File;                    # use DBM usage
# Graphical Tool Kit Libraries
use Tk; $TK = 1;                    # use Tk for using GUI
use Tk::NoteBook; $TK = 2;          # use Tk::NoteBook for using GUI
# GD Librairies for charts
use GD::Graph::pie; $GD = 1;
use GD::Graph::bars; $GD = 2;
use GD::Graph::lines; $GD = 3;
use GD::Graph::area; $GD = 4;
# HTML and PDF manipulation libraries
use HTML::HTMLDoc; $HTML = 1;

# Main variables
$domains_file = "/tmp/domains"; $DOMAINS = 1;   # Path to find Domain file
$rules_file = "/tmp/rules"; $RULES = 1;         # Path to find Rules file
$hw_file = "/tmp/hw"; $HW = 1;                  # Path to find Hardware file
$html_directorie = "/tmp/";                     # Default output directories (HTML
output exclusively)
$dbm_directory = "/tmp/";                       # Default output directories (HTML
output exclusively)
$tmpout_file = "/tmp/.snortalog.tmp";           # Default tempory file (GUI
exclusively)

# Comment variables
$legende_red = "Dangerous connections (potentially bad, further investigation
needed!)";
$legende_green = "Warning connections (strange, may need further intevestigation!)";
$legende_black = "Not dangerous alert";
```

Any variable you don't need may be commented out with a hash "#" (except General Librairies). For example, it's possible to disable specific features like GUI for folks who don't need it, or not to generate charts.

Also, if you have problem with your perl's librairies, it's easy to comment out the following line :

```
# Graphical Tool Kit Libraries
use Tk; $TK = 1;                    # use Tk for using GUI
use Tk::NoteBook; $TK = 2;          # use Tk::NoteBook for using GUI
# GD Librairies for charts
use GD::Graph::pie; $GD = 1;
use GD::Graph::bars; $GD = 2;
use GD::Graph::lines; $GD = 3;
use GD::Graph::area; $GD = 4;
# HTML and PDF manipulation librairies
use HTML::HTMLDoc; $HTML = 1;
```

Or modify my own comments with yours :

# Comment variables
$legende_red = "Dangerous connections (potentially bad, further investigations needed!)";
$legende_green = "Warning connections (strange, may need further investigation!)";
$legende_black = "Not dangerous alert";

7.2 Domain File

The aim of this file is to provide a database of international domain extension (.com .fr .uk etc ...) and its full name (United States, France, United Kingdom etc ...).

As an initial step in the full process of deciphering source domains and including these in the report, SnortALog reads this file and put initialize a hash table in memory. So, it's important to specify the directory where Snortalog can find it. Simply edit Snortalog and set "$domain_file" variable.

It's possible you don't have this file or you don't want to use it, in this case, comment out with "#" the "$domain_file" variable. You must remember that if you comment it out, you will not have the possibility to have certain reports like the **domain report**.

It's also possible to modify this file. You can add new extension (if it doesn't exist) or modify it (if you don't like the full name). Be careful, it's very important to always respect the format :

<EXTENSION>   <Full name>

Here is, an example :

```
DK   Denmark
DO   Dominican Republic
DZ   Algeria
EC   Ecuador
EE   Estonia
EG   Egypt
EH   Western Sahara
ES   Spain
FI   Finland
FR   France
GB   Great Britain (UK)
GD   Grenada
GE   Georgia
GH   Ghana
GL   Greenland
```

7.3 Rules File

The aim of this file is to provide a file which contains all snort's reference attack signatures.

What is a Snort reference attack signature : It's a official internet link which give information about the detected attack. It's looks something like this :

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 27374
(msg:"MISC ramen worm incoming"; flow:established;
content: "GET "; depth: 8;
nocase;reference:arachnids,460; classtype:bad-
unknown; sid:506; rev:3;)
```

In a Snort signature, it's possible to have several references. However, you must realise that Snortalog works with only one reference. If your Snort signature contains several references, then it's important to put your prefered reference first.

You can find a sample reference rule file at http://jeremy.chartier.free.fr/snortalog/rules but Snortalog is able to generate its own rule file by performing a function on your existing rule files. For that you must do :

```
cat *.rules | ./snortalog.pl –genref <your file>
```

In its full process, Snortalog reads this file at the first step for initializing a hash table in memory. So, it's important to specify the directory where Snortalog can find it. Simply edit Snortalog and set the "$rules_file" variable.

It's possible that you don't have this file or you don't want to use it, in this case, comment out with "#" the "$rules_file" variable.

Here an example :

```
ATTACK-RESPONSES Microsoft cmd.exe banner {TCP}         nessus,11633
BACKDOOR subseven 22 {TCP}              url,www.hackfix.org/subseven/
BACKDOOR netbus active {TCP}            arachnids,401
BACKDOOR netbus getinfo {TCP}           arachnids,403
BACKDOOR netbus active {TCP}            arachnids,401
BACKDOOR DeepThroat 3.1 Server Response {UDP}           arachnids,106
BACKDOOR DeepThroat 3.1 Server Response [3150] {UDP}            arachnids,106
BACKDOOR DeepThroat 3.1 Server Response [2140] {UDP}            arachnids,106
BACKDOOR Doly 2.0 access {TCP}          arachnids,312
BAD-TRAFFIC IP Proto 53 (SWIPE) {IP}            cve,CAN-2003-0567
BAD-TRAFFIC IP Proto 55 (IP Mobility) {IP}              cve,CAN-2003-0567
BAD-TRAFFIC IP Proto 77 (Sun ND) {IP}           cve,CAN-2003-0567
BAD-TRAFFIC IP Proto 103 (PIM) {IP}             cve,CAN-2003-0567
CHAT ICQ forced user addition {TCP}             cve,CAN-2001-1305
DDOS TFN Probe {ICMP}           arachnids,443
DDOS tfn2k icmp possible communication {ICMP}           arachnids,425
DDOS Trin00\:DaemontoMaster(PONGdetected) {UDP}         arachnids,187
DDOS TFN client command BE {ICMP}               arachnids,184
DDOS shaft client to handler {TCP}              arachnids,254
DDOS Trin00\:DaemontoMaster(messagedetected) {UDP}              arachnids,186
```

It's also possible to modify this file. You can have several reasons :

- If the snort refrence signature doesn't satisfy you : you can modify the link refrence or simply delete it
- If the official snort reference doesn't exist : you can add it
- If the snort reference doesn't exist : you can add your own rule and choose to reference that

Warning : works only with HTML.

Be careful, it's very important to always respect the format :

<Attack designation> {<Protocol>} <referer>,<ID reference>

7.4 Hardware File

The aim of this file is to provide a file which contains all hardware related message. At this moment, it only contains hardware PIX message but can easily modify to add other.

You can find a sample reference hardware file at http://jeremy.chartier.free.fr/snortalog/hw. In its full process, SnortALog reads this file at the first step for initializing a hash table in memory. So, it's important to specify the directory where SnortALog can find it. Simply edit SnortALog and set the "$hw_file" variable.

It's possible that you don't have this file or you don't want to use it, in this case, comment out with "#" the "$hw_file" variable.

Here an example :

```
%PIX-1-101001   Failover cable OK
%PIX-1-101002   Bad failover cable
%PIX-1-101003   Failover cable not connected (this unit)
%PIX-1-101004   Failover cable not connected (other unit)
%PIX-1-101005   Error reading failover cable status
%PIX-1-102001   Power failure/system reload other side
%PIX-1-103001   No response from other firewall
%PIX-1-103002   Other network interface number OK
%PIX-1-103003   Other network interface number failed
%PIX-1-103004   Other firewall reports this firewall failed
%PIX-1-103005   Other firewall reporting failure
%PIX-1-104001   (P) Switching to ACTIVE
%PIX-1-104002   (P) Switching to STANDBY
%PIX-1-104003   (P) Switching to FAILED
%PIX-1-104004   (P) Switching to OK
%PIX-1-105001   (P) Disabling failover
%PIX-1-105002   (P) Enabling failover
%PIX-1-105005   (P) Lost failover communication
%PIX-1-105006   (P) Link status UP
%PIX-1-105007   (P) Link status DOWN
%PIX-1-105008   (P) Testing interface
%PIX-1-105009   (P) Testing interface passed|failed
%PIX-1-105011   (P) Failover cable communication error
%PIX-1-105020   (P) Incomplete/slow config replication
%PIX-1-105031   Failover LAN interface is UP
%PIX-1-105032   Failover LAN interface is DOWN
%PIX-1-105034   Receive a LAN failover UP msg from peer
%PIX-1-105035   Receive a LAN failover DOWN msg from peer
%PIX-1-105036   PIX dropped a LAN failover cmd msg
%PIX-1-105037   Primary/secondary are switching back and forth
%PIX-1-106101   Number of DENY acl-flows reached limit
%PIX-2-[illegible]600[illegible]   Denied new tunnel. VPN peer limit exceeded
%PIX-3-105010   (P) Failover msg block aloc failed
%PIX-3-201002   Too many connections on static|xlate global address
%PIX-3-201009   PIX is disallowing new connections
%PIX-3-202001   Out of address translation slots
%PIX-3-211001   Memory allocation error
%PIX-3-211003   CPU utilization
```

It's also possible to modify this file. Simply, be careful to respect the format :

<Text to search> <Description>

8 How to use Snortalog ?

You have two solutions for using Snortalog, with the Command Line Interface or with the Graphic User Interface.

8.1 Command Line Interface

8.1.1 Example

By this way, you must redirect the logs to Snortalog as shown by the following shell command :

```
#
# cat logs.file | ./snortalog.pl -n 50 -report
#
```

Why I did not ask for a specific file name ?

Just for one reason (but a smart one :-). For daily logs rotation, I'm using the file name format file_yyyymmdd.log (Year, Month and Day). So it's easy for me to generate daily, weekly, monthly and yearly report without any file renaming operations but we will see that in examples.

So, this is the command line argument :

```
#
# cat <alerts file> or <snort.rules> | ./snortalog.pl <options>
<reports> <filters>
#
```

Also, you can do like this :

```
#
# ./snortalog.pl –file logs.file -n 50 -report
#
```

8.1.2 Available options

The following options are available :

-x	Mode GUI
-r	Resolve IP adresses
-c	Resolve domains
-h <file.html>	Specify a HTML file
-p <file.pdf>	Specify a PDF file
-o <directory>	Specify an output directory
-dbm_dir <directory>	Specify an output directory for DBM usage
-g <gif|png|jpg>	Graph output format
-i	Inverse the result
-d	Mode debug
-n <integer>	Specify a number of line in the result
-file <log file>	Specify an input alert log file
-rulesfile <file>	Specify name and directorie to search rules file
-hwfile <file>	Specify name and directorie to search hardware file
-domainsfile <file>	Specify name and directorie to search domains file
-genref <rules file>	Generate the reference rules file
-help	View this help

The following reports are available :

-src	Top IPs sources
-dst	Top IPs destination
-src_attack	Top IPs sources grouped by attack
-dst_attack	Top IPs destination grouped by attack
-src_dst_attack	Top alert grouped by IPs sources, Ips destination and attack
-attack	Top attack
-class	Top classification
-severity	Top severity
-daily_event	Top number of attack grouped by day
-hour	Top number of attack grouped by hour
-hour_attack	Top specific attack grouped by hour
-dport	Top destination port
-proto	Top usage of protocole
-dport_attack	Top destination port grouped by attack
-nids	Top NIDS host
-stateful	Top stateful problems
-interfaces	Top interfaces events
-domain_src	Top of domain source
-portscan	Top of portscan alert
-actions	Top of firewall action (DROP, REJECT, ACCEPT, etc ...)
-rules	Top number of DROP by rule (only Fw-1)
-reasons	Top number of DROP reason (only Fw-1)
-src_dport	Top IPs sources grouped by destination port
-dst_dport	Top IPs destination grouped by destination port
-typelog	Number of occurrences by type of logs
-hwlog	Number of occurrences by hardware related message logs
-report	All reports

<u>The following filters are available :</u>

-fsrc	Sources filter
-fdst	Destination filter
-fproto	Protocole filter
-fdport	Destination port filter
-fmonth	Month filter
-fday	Day filter
-fhour	Hour filter
-fether	Interface filter
-fseverity	Severity filter
-faction	Firewall action filter
-frule	Firewall rule filter
-ftype	Type of logs

8.1.3 Examples

```
# cat snort*.rules | ./snortalog.pl -genref refsigtxt
```

Snortalog will generate a referenced rules file from your Snort rule or your own signatures.

```
# cat file.logs | ./snortalog.pl -r -n 30 -report
```

Snortalog will generate a report in ASCII format with address resolution and a maximum of 30 occurences for all reports.

```
# ./snortalog.pl –file file.logs -r -n 30 -dst_attack -report
```

Snortalog will generate a report in ASCII format with address resolution and a maximum of 30 occurences for the report dst_attack.

```
# cat file.logs | ./snortalog.pl -r -i -h file.html -report
```

Snortalog will generate a report in HTML format stored in file.html with address resolution and display the results from least frequent to most frequent occurences (reverse mode).

```
# cat file.logs | ./snortalog.pl -r -g gif -h file.html -u /tmp/ -report
```

Same as the previous example but with Gif graphs and in a specific directorie.

```
# cat file.logs | ./snortalog.pl –n 50 -report –fether eth0
```

Snortalog will generate a report with filter interface “eth0”.

```
# cat file.logs | ./snortalog.pl -i -n 30 -report | /usr/sbin/sendmail -f user@domain
user@domain
```

Snortalog will generate a report in ASCII format with reverse request, and a maximum of 30 occurences for all reports and send the result by mail.

```
# cat file_200212[1-7] | ./snortalog.pl -report
```

Snortalog will generate a report in ASCII format with all events of the first week of December (between the 1st and 7th).

```
# cat file_20021* | ./snortalog.pl -report
```

Snortalog will generate a report in ASCII format with all events of the three last months of the year 2002 (month 10, 11 and 12).

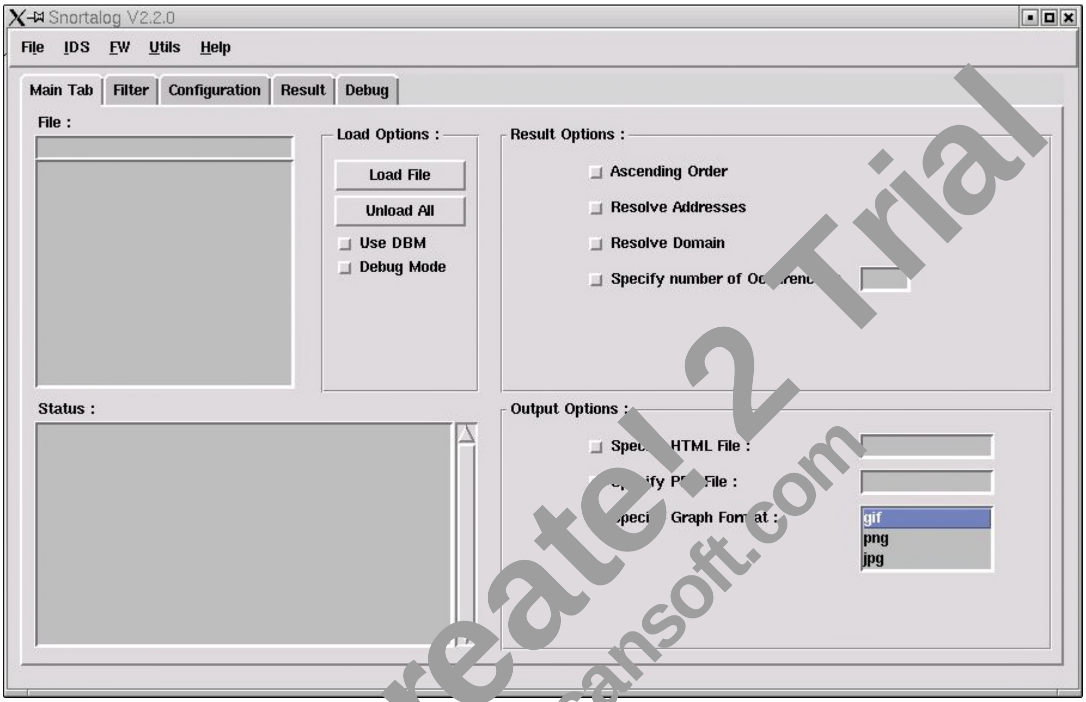
<u>Warning :</u> The usage of “-r” and “-c” option will slow down the process.

8.2 Graphic User Interface

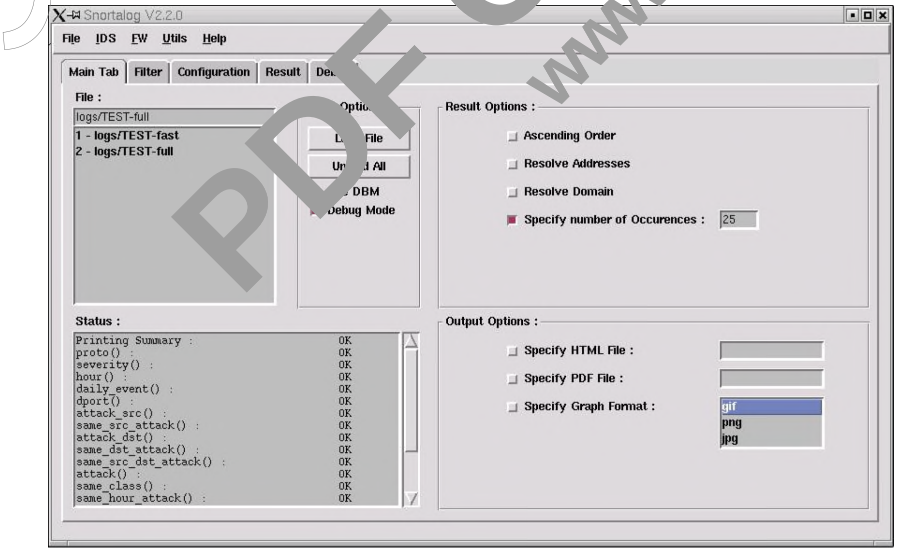
When launching the GUI, be careful to install all dependencies and perform Snortalog with this option :

```
# ./snortalog.pl –x
```

If everything is okay, you will see this :



Below, an easy example step by step for using GUI :



File : logs/TEST-full

1 - logs/TEST-fast
2 - logs/TEST-full

☒ Specify number of Occurences : 25

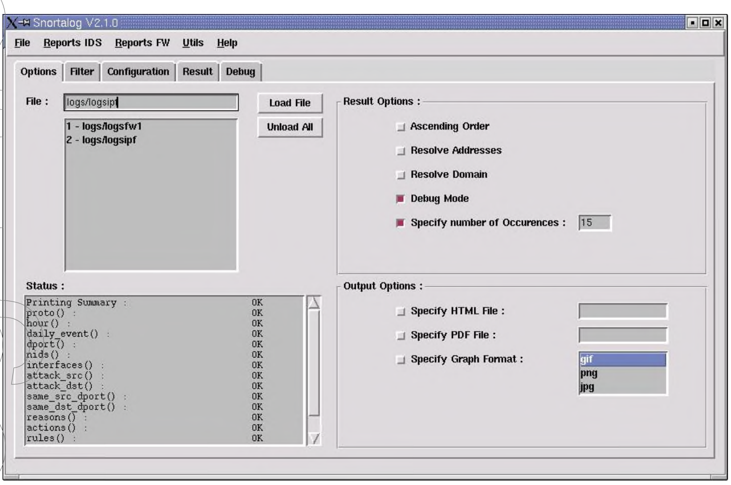
```
Printing Summary :                      OK
proto() :                               OK
severity() :                            OK
hour() :                                OK
daily_event() :                         OK
dport() :                               OK
attack_src() :                          OK
same_src_attack() :                     OK
attack_dst() :                          OK
same_dst_attack() :                     OK
same_src_dst_attack() :                 OK
attack() :                              OK
same_class() :                          OK
same_hour_attack() :                    OK
```

First, we need to load all the log files we want. To do this, enter the path and the file name in the "File box" and click "Load File".

We can select or unselect "Result Options" or "Output Options".

Warning : "Resolve Addresses" and "Resolve Domain" can take few minutes for result.

Second, we need to select a report from "Reports IDS or FW" tasks menu bar.



File : logs/logsipf

1 - logs/logsfw1
2 - logs/logsipf

Result Options :

- ☐ Ascending Order
- ☐ Resolve Addresses
- ☐ Resolve Domain
- ☒ Debug Mode
- ☒ Specify number of Occurences : 15

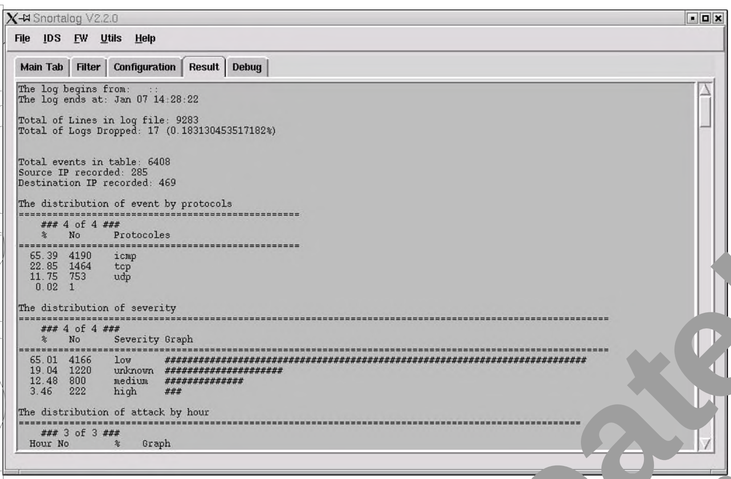
Status :

```
Printing Summary :                              OK
proto() :                                       OK
hour() :                                        OK
daily_event() :                                 OK
dport() :                                       OK
nids() :                                        OK
interfaces() :                                  OK
attack_src() :                                  OK
attack_dst() :                                  OK
same_src_dport() :                              OK
same_dst_dport() :                              OK
reasons() :                                     OK
actions() :                                     OK
rules() :                                       OK
```

Output Options :

- ☐ Specify HTML File :
- ☐ Specify PDF File :
- ☐ Specify Graph Format : gif / png / jpg

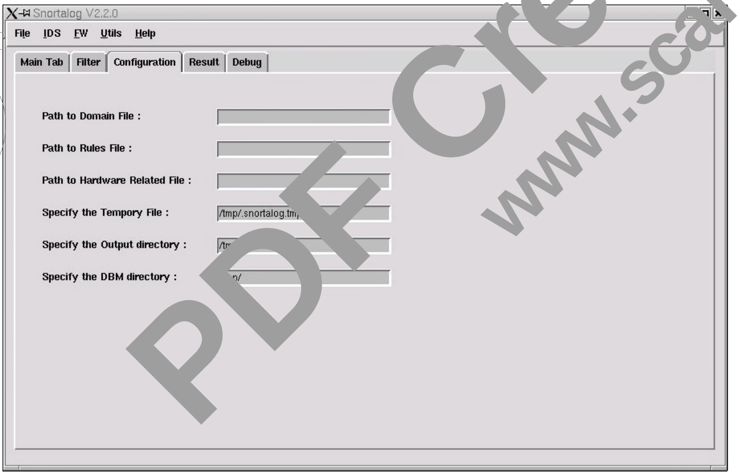
We can see if all reports are OK in "States" list.



```
The log begins from:   ::
The log ends at: Jan 07 14:28:22

Total of Lines in log file: 9283
Total of Logs Dropped: 17 (0.183130453517182%)


Total events in table: 6408
Source IP recorded: 285
Destination IP recorded: 469

The distribution of event by protocols
==================================================
    ### 4 of 4 ###
    %    No      Protocoles
==================================================
  65.39  4190    icmp
  22.85  1464    tcp
  11.75  753     udp
   0.02  1

The distribution of severity
=========================================================================================================
    ### 4 of 4 ###
    %    No      Severity Graph
=========================================================================================================
  65.01  4166    low      ################################################################################
  19.04  1220    unknown  ######################
  12.48  800     medium   ###############
  3.46   222     high     ###

The distribution of attack by hour
====================================================================================================
    ### 3 of 3 ###
  Hour No        %    Graph
```

We can view the result in "Result" tab and navigate with the right scrollbar.

Also, as we have selected "Debug Mode" on main screen, we can view the logs that Snortalog can't load in the "Debug" tab.



Path to Domain File :

Path to Rules File :

Path to Hardware Related File :

Specify the Tempory File : /tmp/.snortalog.tm

Specify the Output directory : /tr

Specify the DBM directory : n/

It's possible with the "Configuration" tab to configure several variables.

We can specify the path to "Domain File" and "Rules File". It's important to have something in "Tempory File" and "Output Directory" if we want everything to work correctly.

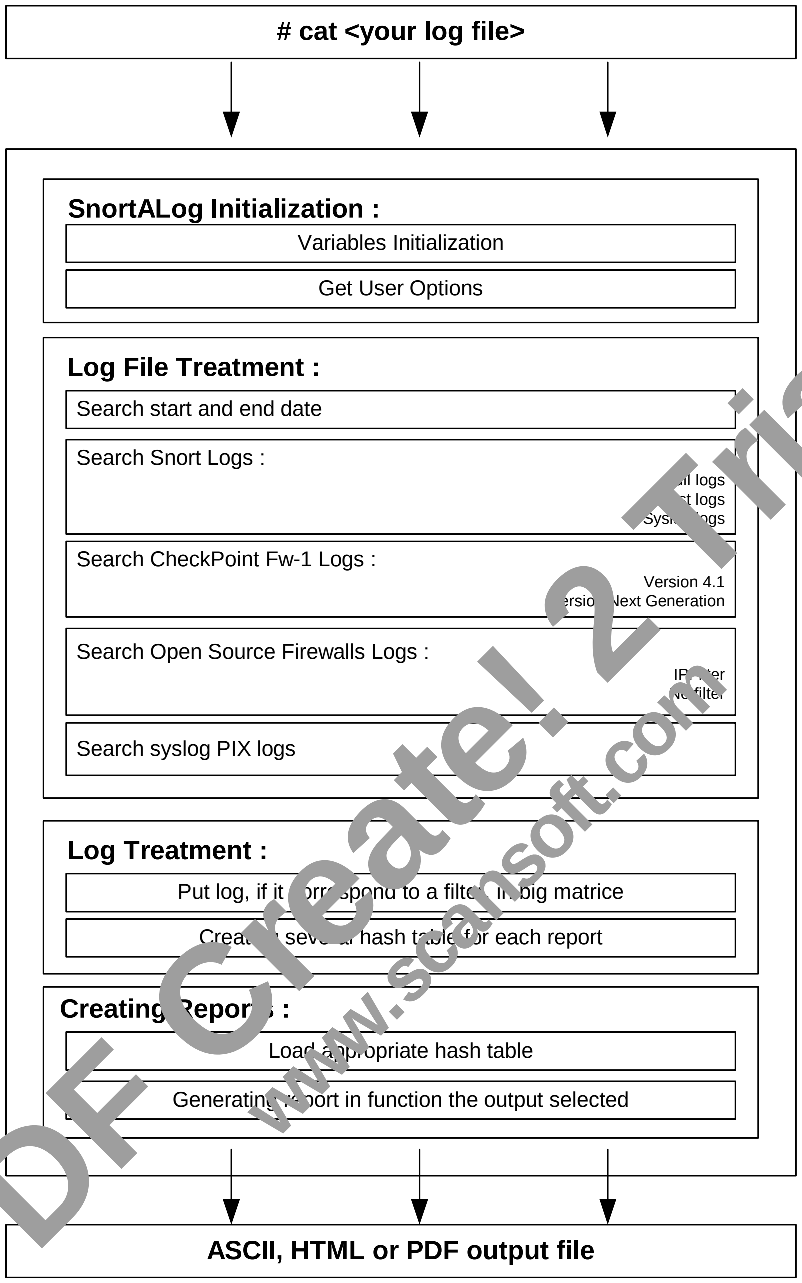
These default variables can also be modified directly in PERL program.

What advantages does using the GUI bring me ?

It's interesting to use the GUI because you can load several log files at the same time and generate as many reports as you want. In CLI mode, you can't do that because you need to redirect your logs each time you want to use Snortalog.

Using this method (GUI), you can generate severals reports (ASCII, HTML or PDF output) in one step.

9 How Snortalog works ?



```
# cat <your log file>
```

SnortALog Initialization :

Variables Initialization

Get User Options

Log File Treatment :

Search start and end date

Search Snort Logs :

Full logs
Fast logs
Syslog logs

Search CheckPoint Fw-1 Logs :

Version 4.1
Version Next Generation

Search Open Source Firewalls Logs :

IPFilter
Netfilter

Search syslog PIX logs

Log Treatment :

Put log, if it correspond to a filter, in big matrice

Creating several hash table for each report

Creating Reports :

Load appropriate hash table

Generating report in function the output selected

ASCII, HTML or PDF output file

SnortALog is a wonderfull and efficient mechanism to work with several kind of logs. It can easy work with one million of them but first, you need to take care.

You must to know that biggest the CPU and Physical memory is, faster and better the result will be. So, you can have a problem with log file which contains several million of alerts (more 100Mo) because SnortALog extract each logs from file to put it in several table and release the memory after result generation. With huge log file, you need more RAM and Swap space else the process SnortALog kill.

By example, if your log file size is 100Mo, you need 600Mo of system memory swap.

If SnortALog use all swap memory, I council you to modify your rotation log or to do something for decrease the amount of Megabits (best file handliing).

10 What kind of logs does Snortalog expect ?

Here, you can find all kind of logs SnortALog are expecting. If you have logs that SnortALog don't recognize ("-d" option can help you), send me them by email, I will be happy to upgrade SnortALog.

10.1 Snort logs

10.1.1 Snort fast alert

```
01/31-17:37:39.987506  [**] [1:671:4] SMTP sendmail 8.6.9c exploit [**]
[Classification: Attempted_User_Privilege_Gain] [Priority: 1] {TCP} 1.2.3.4:27191 ->
192.168.1.97:25

01/31-17:37:39.989398  [**] [1:1160:6] WEB-MISC netscape dir index wp [**]
[Classification: Attempted_Information_Leak] [Priority: 2] {TCP} 1.2.3.4:52502 ->
192.168.1.97:80

01/31-17:37:39.991339  [**] [1:677:5] MS-SQL/SMB sp_password password change [**]
[Classification: Attempted_User_Privilege_Gain] [Priority: 1] {TCP} 1.2.3.4:38263 ->
192.168.1.97:139

01/31-17:37:39.999730  [**] [1:345:5] FTP EXPLOIT wu-ftpd 2.6.0 site exec format
string overflow generic [**] [Classification: Attempted_Administrator_Privilege
Gain] [Priority: 1] {TCP} 1.2.3.4:1[illegible] -> 192.168.1.97:21

01/31-17:37:40.008521  [**] [1:221:1] DDOS TFN Probe [**] [Classification:
Attempted_Information_Leak] [Priority: 2] {ICMP} 1.2.3.4 -> 192.168.1.97
```

Snort Command Line Example : # snort **–A fast** –Cdi <interface> -c <snort configfile> -l <snort directorie>

10.1.2 Snort full alert

```
[**] [1:540:8] CHAT MSN message [**]
[Classification: Misc activity] [Priority: 3]
09/23-09:09:[illegible]0.6[illegible]544 10.21.145.60:1714 -> 207.46.108.21:1863
TCP TTL:1[illegible]4 TOS:0x0 ID:16516 IpLen:20 DgmLen:201 DF
***AP*** Seq: 0xAFF293D6  Ack: 0x9549F9D7  Win: 0xF9F8  TcpLen: 20

 [**] [1:528:3] BAD TRAFFIC loopback traffic [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
09/23-09:09:22.581891 10.29.12.177:161 -> 127.0.0.1:162
UDP TTL:57 TOS:0x0 ID:40247 IpLen:20 DgmLen:211
Len: 191
[Xref => http://rr.sans.org/firewall/egress.php
```

Snort Command Line Example : # snort **–A full** –Cdi <interface> -c <snort configfile> -l <snort directorie>

10.1.3 Snort syslog alert

```
Mar 12 14:10:31 10.0.0.2/10.0.0.2 snort[524]: [1:1287:5] WEB-IIS scripts access
```

```
[Classification: access to a potentially vulnerable web application] [Priority: 2]:
{TCP} 193.219.28.101:63582 -> 149.46.224.194:80

Mar 12 13:44:28 10.0.0.1/10.0.0.1 snort[22976]: [1:895:5] WEB-CGI redirect access
[Classification: Attempted Information Leak] [Priority: 2]: {TCP}
208.214.188.62:61119 -> 149.46.214.192:80

Mar 12 14:24:50 10.0.0.2/10.0.0.2 snort[524]: [1:1244:6] WEB-IIS ISAPI .idq attempt
[Classification: Web Application Attack] [Priority: 1]: {TCP} 193.249.155.3:1424 ->
149.46.224.192:80

Mar 12 14:25:20 10.0.0.2/10.0.0.2 snort[524]: [1:466:1] ICMP L3retriever Ping
[Classification: Attempted Information Leak] [Priority: 2]: {ICMP} 80.13.197.190 ->
149.46.224.13

Mar 12 14:25:55 10.0.0.2/10.0.0.2 snort[524]: [1:1042:6] WEB-IIS view source via
translate header [Classification: access to a potentially vulnerable web
application] [Priority: 2]: {TCP} 80.13.197.190:4787 -> 149.46.244.192:80
```

Snort Command Line Example : # snort **–A none** –Cdi <interface> -c <snort configfile> -l <snort directorie> **-s** “”

10.2 CheckPoint FW-1 syslog Format

It’s very easy to redirect Fw-1 logs to a server syslog on all Unix Plateform (Management or Enforcement Module) :

```
# fw log –fnt | logger &
```

Explanations :

-f : Upon reaching end of file stay and wait for new records forever. Default is to stop at end of file
-t : Goto file end. -t must come with -f flag
-n : Do not resolve IP addresses. Default is to resolve Ips

LOGGER in the end of the syntax permit to transmit new logs, in the local syslog server, to the remote syslog server. So, it’s possible to see logs in **/var/log/messages** or **/var/adm/messages** files.

Moreover, you can with your local syslog server, redirect the flaw to an other syslog server :

```
# vi /etc/syslog.conf
*.*             /var/log/messages
*.*             @10.0.0.1
```

10.2.1 FW-1 4.1

```
May  6 04:12:51 10.0.0.1/10.0.0.1 root:  4:12:50 drop   flamm.192.225 >qfe12 proto
tcp src 17.216.0.58 dst 206.117.161.100 service mail s_port 37163 len 48 rule 191
xlatesrc 195.46.206.51 xlatedst 206.117.161.100 xlatesport 37163 xlatedport mail

May  6 04:12:55 10.0.0.1/10.0.0.1 root:  4:12:55 drop   flamm.192.225 >qfe12 proto
tcp src 17.216.0.58 dst 216.219.253.216 service mail s_port 37025 rule 0 reason:
unknown established TCP packet

May  6 04:03:55 10.0.0.2/10.0.0.2 root:  4:03:54 reject fwvtx      >hme4 proto tcp
src 172.171.144.17 dst 162.168.1.9 service pop-3 s_port sqlnet1 rule 34 reason: port
belong to service in TCP Fast Mode, port: sqlnet1
```

```
May  6 04:12:40 10.0.0.2/10.0.0.2 root:  4:12:39 drop   picasso    >qfe0 proto icmp
src 203.148.174.121 dst 145.246.217.61 rule 142 icmp-type 3 icmp-code 1

May  6 04:08:21 10.0.0.1/10.0.0.1 root:  4:08:21 drop   cosme      >hme0 proto udp
src 200.41.92.96 dst 144.127.222.45 service nbname s_port 1045 len 78 rule 41
```

10.2.2 FW-1 Next Generation

```
Aug 26 23:53:10 10.0.0.1 root: [ID 702911 user.notice] 23:53:10 drop   10.0.0.1
>qfe0 product: VPN-1 & FireWall-1; src: 195.46.223.247; s_port: nbdatagram; dst:
195.146.223.2; service: nbdatagram; proto: udp; message_info: Address spoofing;

Aug 26 23:53:54 10.0.0.1 root: [ID 702911 user.notice] 23:53:54 drop   10.0.0.1
>hme0 product: VPN-1 & FireWall-1; src: 68.155.36.158; dst: 14.17.218.26; proto:
icmp; icmp-type: 8; icmp-code: 0; rule: 28;

Aug 27 05:56:53 10.0.0.1 root: [ID 702911 user.notice]  5:56:52 drop   10.0.0.1
>hme0 product: VPN-1 & FireWall-1; src: 62.149.140.15; s_port: http; dst:
191.17.218.225; service: 1223; proto: tcp; th_flags: 12; message_info: TCP packet
out of state;

Aug 27 09:18:15 10.0.0.1 root: [ID 702911 user.notice]  9:18:14 drop   10.0.0.1
>qfe3 product: VPN-1 & FireWall-1; src: 12.18.14.20; s_port: 35896; dst:
171.171.0.12; service: syslog; proto: udp; rule: 28;

Aug 27 10:06:14 10.0.0.1 root: [ID 702911 user.notice] 10:06:13 drop   10.0.0.1
>hme0 product: VPN-1 & FireWall-1; src: 213.15.59.175; s_port: 1619; dst:
191.17.218.246; service: 135; proto: tcp; rule: 28;
```

10.3 CheckPoint FW-1 fw logexport Format

An other way to work with Fw-1 logs is to export them with Fw-1 command :

```
# fw logexport –np –o <file>                                  (with Fw-1 4.1)
# fwm logexport –np –o <file>                                 (with Fw-1 NG AI)

or

# fw logexport –np | ./snortalog.pl –n 10 -report             (with Fw-1 4.1)
# fwm logexport –np | ./snortalog.pl –n 10 -report            (with Fw-1 NG AI)
```

<u>Explanation :</u>

-o : Output file name. Default is printing to the screen
-n : No IP resolving. Default is to resolve all IPs
-p : No port resolving. Default is to resolve all ports

The output of the fwm logexport got a little "unpredictable" after NG, and the logformat is now documented in the first line of every logfile.

```
num;date;time;orig;type;action;alert;i/f_name;i/f_dir;proto;
```

```
src;dst;service;s_port;len;rule;xlatesrc;xlatedst;xlatesport;xlatedport;icmp-
type;icmp-code;reason:;IKE Log:;rpc_prog;sys_msgs
```

10.3.1 FW-1 4.1

```
960982;18Dec2003; 6:21:36;2.2.64.48;log;accept;;eth-s1p1c0;inbound;udp;
204.74.161.2;2.2.64.53;domain;36282;72;37;204.74.161.2;64.32.4.53;36282;domain;;;;;;

960993;18Dec2003; 6:21:36;2.2.64.48;log;accept;;eth-s1p1c0;inbound;50;
67.98.22.44;2.2.64.17;61855;smtp;608;34;67.98.22.44;192.168.3.15;;;;;;;;

960996;18Dec2003; 6:21:36;2.2.64.48;log;accept;;eth-s4p2c0;inbound;50;
2.2.64.17;67.98.22.44;2597;60745;96;35;192.168.3.15;67.98.22.44;;;;;;;;

960998;18Dec2003; 6:21:36;2.2.64.48;log;accept;;eth-
s1p1c0;inbound;udp;63.251.230.246;2.2.64.53;
domain;31334;45;37;63.251.230.246;64.32.4.53;31334;domain;;;;;;

961017;18Dec2003; 6:21:36;2.2.64.48;log;accept;;eth-s4p1c0;inbound;tcp;
10.170.96.53;12.159.228.50;LifeWatch.Host.20024;1883;48;101;2.2.64.[illegible]2;
12.159.228.50;28564;LifeWatch.Host.20024;;;;;;
```

10.3.2 FW-1 Next Generation

```
86;17Dec2003;23:55:55;10.0.0.1;log;accept;;eth-s1p3c0;inbound;VPN-1 & FireWall-
1;;204.11.33.38;145.77.4.18;tcp;53;23;53683;;;;;;;;;;;;;;;;;;

87;17Dec2003;23:55:56;10.0.0.1;log;drop;;eth-s1p1c0;inbound;VPN-1 & FireWall-
1;;195.123.22.46;198.186.126.12;tcp;[illegible];in[illegible]pc.tcp;1747;;;;;;;;;;;;;;;;;;

88;17Dec2003;23:55:57;10.0.0.1;log;accept;;eth-s1p3c0;inbound;VPN-1 & FireWall-
1;;172.75.75.60;194.96.111.[illegible];tcp;[illegible]3;nbsession;2523;;;;;;;;;;;;;;;;;;

89;17Dec2003;23:55:59;10.0.0.1;log;accept;;eth-s1p3c0;inbound;VPN-1 & FireWall-
1;;204.18.3.39;145.11[illegible]74.19;[illegible];53;26[illegible];36748;;;;;;;;;;;;;;;;;;

90;17Dec2003;23:56:[illegible]1;10.0.0.1;log;accept;;eth-s1p1c0;inbound;VPN-1 & FireWall-
1;;194.116.52.31;19[illegible]46.166.78;udp;14;snmp-read;1233;;;;;;;;;;;;;;;;;;

91;17Dec2003;[illegible]3:56:[illegible]4;10.0.0.1;log;accept;;eth-s1p3c0;inbound;VPN-1 & FireWall-
1;;204.18.[illegible]138;145.[illegible]13.16.41;tcp;53;HP.JetDirect;53687;;;;;;;;;;;;;;;;;;
```

10.4 Cisco PIX syslog Format

The only things to do is to redirect PIX logs via syslog server :

```
Jan 28 12:30:25 [10.200.7.12.2.2] %PIX-4-106023: Deny tcp src
outside:62.4.95.39/26457 dst DMZ:62.4.85.170/1721 by access-group "outside"
Jan 28 12:30:25 [10.200.7.12.2.2] %PIX-4-106023: Deny tcp src
outside:62.4.95.39/26458 dst DMZ:62.4.85.170/1722 by access-group "outside"
Jan 28 12:30:25 [10.200.7.12.2.2] %PIX-4-106023: Deny tcp src
outside:62.4.95.39/26459 dst DMZ:62.4.85.170/1723 by access-group "outside"
Jan 28 12:30:25 [10.200.7.12.2.2] %PIX-4-106023: Deny tcp src
outside:62.4.95.39/26460 dst DMZ:62.4.85.170/1724 by access-group "outside"
```

```
Jan 28 12:30:25 [10.200.7.12.2.2] %PIX-4-106023: Deny tcp src
outside:62.4.95.39/26461 dst DMZ:62.4.85.170/1725 by access-group "outside"
```

Or

```
Jan 26 14:07:01 [10.200.7.12.2.2] Jan 26 2004 13:54:28: %PIX-4-106023: Deny icmp src
outside:62.2.91.125 dst DMZ:62.4.85.189 (type 8, code 0) by access-group "outside"
Jan 26 14:07:01 [10.200.7.12.2.2] Jan 26 2004 13:54:28: %PIX-4-106023: Deny icmp src
outside:62.2.91.125 dst DMZ:62.4.85.190 (type 8, code 0) by access-group "outside"
Jan 26 14:07:01 [10.200.7.12.2.2] Jan 26 2004 13:54:28: %PIX-4-106023: Deny icmp src
outside:62.2.91.125 dst DMZ:62.4.85.191 (type 8, code 0) by access-group "outside"
Jan 26 14:07:01 [10.200.7.12.2.2] Jan 26 2004 13:54:29: %PIX-4-106023: Deny tcp src
outside:62.2.91.125/4564 dst DMZ:62.4.85.178/135 by access-group "outside"
Jan 26 14:07:02 [10.200.7.12.2.2] Jan 26 2004 13:54:29: %PIX-4-106023: Deny tcp src
outside:62.4.95.108/25 dst DMZ:62.4.85.178/56518 by access-group "outside"
```

Two kind of logs arent identical except there are time tow times in one. We had a configuration fault in the PIX. The PIX was sending the time and also the Syslog daemon was adding the time to the log entry. So, it’s possible to adjust the PIX config not sending the time with the message to the Syslog daemon.

All message entries have a PIX alert number, for example: %PIX-1-101001 for Severity 1, %PIX-2-106001 for Severity 2 and %PIX-3-105010 for Severity 3. Snortalog is able to manage this severity level.

The PIX has 7 levels of messages :

- Alert Messages, Severity 1
- Critical Messages, Severity 2
- Error Messages, Severity 3
- Warning Messages, Severity 4
- Notification Messages, Severity 5
- Informational Messages, Severity 6
- Debugging Messages, Severity 7

Also, I implementing two additionnal features in SnortAlog 2.2. First, possibility to integrate Cisco PIX IDS. Second, there are an extra diagram in the PIX output with a summary of PIX hardware related messages from the Syslog file.

10.5 Free Firewalls

10.5.1 IPFilter

```
May  6 05:42:54 10.0.0.1/10.0.0.1 ipmon[91]: 05:42:54.104248 fxp0 @0:26 b
212.73.231.228 -> 190.17.117.36 PR icmp len 20 40 icmp echo/0 IN

May  5 22:44:40 10.0.0.2/10.0.0.2 ipmon[66]: 22:44:40.872086 fxp4 @0:285 b
192.178.8.17,32845 -> 190.65.60.33,80 PR tcp len 20 48 -S 1744106958 0 24820 IN

May  6 03:49:14 10.0.0.2/10.0.0.2 ipmon[9775]: 03:49:14.170319 sf2 @0:181 b
191.46.17.167,1444 -> 191.46.17.146,6050 PR tcp len 20 40 -A 3374366425 1952656703
8760 IN

May  6 04:05:56 10.0.0.1/10.0.0.1 ipmon[9942]: 04:05:56.214183 sf3 @0:1204 b
101.88.2.3,137 -> 101.88.2.254,137 PR udp len 20 78  IN
```

```
May  6 04:00:16 10.0.0.1/10.0.0.1 ipmon[9775]: 04:00:16.730522 sf2 @0:181 b
191.146.27.167,1444 -> 191.146.17.146,6050 PR tcp len 20 40 -A 3374366425 507567241
8760 IN
```

10.5.2 Netfilter

```
Nov 17 16:52:52 host kernel: IN=eth0 OUT=
MAC=00:10:5a:b1:25:1d:00:d0:b7:bd:aa:28:08:00 SRC=197.163.1.92 DST=10.18.1.49 LEN=48
TOS=0x00 PREC=0x00 TTL=128 ID=31660 DF PROTO=TCP SPT=4075 DPT=25 WINDOW=16384
RES=0x00 SYN URGP=0

Nov 17 16:52:54 host kernel: IN=eth0 OUT=
MAC=00:10:5a:b1:25:1d:00:d0:b7:bd:aa:28:08:00 SRC=197.163.1.92 DST=10.18.1.49 LEN=48
TOS=0x00 PREC=0x00 TTL=128 ID=31677 DF PROTO=TCP SPT=4075 DPT=21 WINDOW=16384
RES=0x00 SYN URGP=0

Nov 17 16:53:00 host kernel: IN=eth0 OUT=
MAC=00:10:5a:b1:25:1d:00:d0:b7:bd:aa:28:08:00 SRC=197.163.1.92 DST=10.18.1.49 LEN=48
TOS=0x00 PREC=0x00 TTL=128 ID=31711 DF PROTO=TCP SPT=4075 DPT=23 WINDOW=16384
RES=0x00 SYN URGP=0

Nov 17 16:53:48 host kernel: IN=lo OUT=
MAC=00:00:00:00:00:00:00:00:00:00:00:00:08:00 SRC=197.163.1.92 DST=10.18.1.49 LEN=36
TOS=0x10 PREC=0x00 TTL=64 ID=0 DF PROTO=UDP SPT=123 DPT=32768 LEN=16

Nov 17 16:54:48 host kernel: IN=lo OUT=
MAC=00:00:00:00:00:00:00:00:00:00:00:00:08:00 SRC=197.163.1.92 DST=10.18.1.49 LEN=36
TOS=0x10 PREC=0x00 TTL=64 ID=0 DF PROTO=UDP SPT=123 DPT=32768 LEN=16
```

11 FAQ

1) When I try to run Snortalog, this error message appears :

```
Can't locate GD/Graph/pie.pm in @INC (@INC contains: /usr/local/lib/perl5/5.8.0/ sun4-
solaris /usr/local/lib/perl5/5.8.0 /usr/local/lib/perl5/site_perl/5.8.0/sun 4-solaris
/usr/local/lib/perl5/site_perl/5.8.0 /usr/local/lib/perl5/site_perl .) at
./snortalog.pl line 96.
```

- You can be sure that Perl isn't finding the appropiate librairies. For help in correcting this, go to the dependencies page.

2) I correctly compiled dependency libraries but it's no better :

You can be sure you are not using Perl 5.8. Verify like this :

```
Summary of my perl5 (revision 5.0 version 8 subversion 0) configuration:
 Platform:
      ...
 Compiler:
      ...
 Linker and Libraries:
      ...
 Dynamic Linking:
       ...

Characteristics of this binary (from libperl):
 Compile-time options: MULTIPLICITY USE_ITHREADS USE_LARGE_FILES
PERL_IMPLICIT_CONTEXT
 Built under linux
 Compiled at Sep  6 2002 23:24:44
 @INC:
   /usr/lib/perl5/5.8.0/i386-linux-thread-multi
   /usr/lib/perl5/5.8.0
   /usr/lib/perl5/site_perl/5.8.0/i386-linux-thread-multi
   /usr/lib/perl5/site_perl/5.8.0
   /usr/lib/perl5/site_perl
   /usr/lib/perl5/vendor_perl/5.8.0/i386-linux-thread-multi
   /usr/lib/perl5/vendor_perl/5.8.0
   /usr/lib/perl5/vendor_perl
```

3) When I try to generate PNG charts, this error message appears :

```
Can't locate object method "png" via package "GD::Image" at ./snortalog.pl line XXX.
```

- Your Perl's libraries don't support PNG format.To correct this, try to use GIF or JPG format instead.

4) When I perform SnortALog, the process kill after a long moment :

It seems to be a system swap memory problem. Verify two things :

- The amount of swap that SnortALog use
- The size of your log file

To solve this problem, you can follow several ways :

- Change your log rotation
- Tune your Snort or Firewall configuration to log less
- Increase your RAM or swap partition
- Use SnortALog filter features to select what you want (only HIGH severity alert for Snort or only DROP alert for Firewall)

5) When I try to run SnortALog GUI, this error appears :

```
Tk::Error: Can't set width to `895' for MainWindow=HASH(0x8684054): unknown option
"width" at /usr/local/lib/perl5/site_perl/5.8.3/i686-linux/Tk/Configure.pm line 46.
 at /usr/local/lib/perl5/site_perl/5.8.3/i686-linux/Tk/Derived.pm line 294
 Tk callback for .
 Tk callback for .
 Tk::Derived::configure at /usr/local/lib/perl5/site_perl/5.8.3/i686-
linux/Tk/Derived.pm line 306
Can't set width to `895' for MainWindow=HASH(0x8684054): unknown option "width"
at /usr/local/lib/perl5/site_perl/5.8.3/i686-linux/Tk/Configure.pm line 46.
 at /usr/local/lib/perl5/site_perl/5.8.3/i686-linux/Tk/Derived.pm line 294
```

- You can be sure you are not using appropriate Tk librairies. For help in correcting this, go to the dependencies page, download and compile Tk-800.024.tar.gz

6) When I try to perform SnortALog, this error message appears :

```
syntax error at snortalog.pl line 1901, near "$opton;"
Execution of snortalog.pl aborted due to compilation errors.
Starting... There are 1034969 log records in the file
```

- You can be sure you are not using Perl 5.8. Verify like this : `perl -V`