HOSD -
Installation and User Guide

François Gagnon
fgagnon@sce.carleton.ca
www.sce.carleton.ca\~fgagnon

NMAI Research Group
www.nmai.ca
Carleton University

Version: 0.2
Last Modified: June 22, 2009

Abstract

This document describes the installation (and initial configuration) process of HOSD. It also provides a brief introduction regarding the usage of HOSD.

1 System Requirements

You will need a Windows computer (HOSD has been tested on Windows 2000 SP4 and Windows XP SP3). Java JRE 1.5 or 1.6 must be installed on that computer.

To install HOSD, follow these steps:

- Download and install SWI-Prolog from `www.swi-prolog.org`. After the installation add SWI-Prolog "bin" directory to the "PATH" environment variable of your computer (Usually, the "bin" directory would be "C:\Program Files\pl\bin\").
- Download HOSD-X.Y.zip from `hosd.sourceforge.net`. Extract the file where you want HOSD to reside. This location will then become the HOSD path, denoted "$HOSD_Path".
- Download the Windump executable from `www.winpcap.org/windump/` and put the executable in "$HOSD_Path\TOOLS\". The executable should be named "windump.exe". Note that for Windump to run properly, you might also need to install WinPcap.

2 Installation

After the extraction of HOSD-X.Y.zip, "$HOSD_Path" will contain the files POSD-X.Y.jar and AOSD-X.Y.jar as well as the following folders:

- CONFIG
- DOCUMENTS
- FILES
- LIB
- TOOLS

"CONFIG" contains the configuration file "Preference.txt". The file must be configured once, according to Section 3, before running HOSD.

"DOCUMENTS" contains the documents related to HOSD installation and usage.

"FILES" contains the rules used by HOSD and this is also where the temporary computation files will be stored.

"LIB" contains the libraries needed by HOSD.

"TOOLS" contains the third party tools required by HOSD (e.g., Windump).

"POSD-X.Y.jar" is the passive module of HOSD program. Section 4 explains how it can be launched.

"AOSD-X.Y.jar" is the active module of HOSD program. Section 5 explains how it can be launched.

But first, you need to configure HOSD.

3 Preferences Configuration

The file "Preferences.txt" located in "$HOSD_Path\CONFIG" must contains a value for the following entries:

- HOSD_Windump_Device_Id = XYZ
- HOSD_Windump_Device_Name = XYZ
- SOURCE_IP = A.B.C.D

Figure 1: Example of output from Windump -D

```
1.\Device\NPF_GenericDialupAdapter (Adapter for generic dialup and VPN capture)
2.\Device\NPF_61A50110-307B-4F0F-A408-70D72B7279A6 (Intel(R) PRO/100 VM Network Connection)
```

Table 1: Understanding the output from Windump -D

Device Id	Device name	Device description
1	\Device\NPF_GenericDialupAdapter	(Adapter for generic dialup and VPN capture)
2	\Device\NPF_61A50110-307B-4F0F-A408-70D72B7279A6	(Intel(R) PRO/100 VM Network Connection)

To figure out the windump values, run Windump with the "-D" option. This will give you a list of network devices available to Windump. Each item of the list

will contain an id, a name and a description, see Figure 1 and Table 1. Find the device you want to use and enter its id in the field HOSD_Windump_Device_Id and its name in the field HOSD_Windump_Device_Name.

The source ip entry provides the ip address of the computer running hosd (when packets need to be sent).

An example of "Preferences.txt" is presented in Figure 2.

Figure 2: Example of "Preferences.txt"

```
HOSD_Windump_Device_Id = 2
HOSD_Windump_Device_Name = \Device\NPF_61A50110-307B-4F0F-A408-70D72B7279A6
SOURCE_IP = 10.92.39.6
```

4 Running POSD

To run POSD, you must provide a pcap input file (-r), the name of an output file (-o) and the IP of the host to fingerprint (-h).

Assuming you are in "$HOSD_Path, you can start POSD using the following command:

java -Djava.library.path=./LIB/;. -jar POSD-X.Y.jar -r C:\test.pcap -o C:\outputPOSD.txt -h 10.92.39.4

This will provide the OS for host "10.92.39.4" based on the traffic found in trace "C:\test.pcap" and the result will be stored in "C:\test.out". Section 6 discusses how to interpret the output file.

Currently, it is not possible to analyze the traffic directly from the network. So the traffic must be given through a pcap file using the -r option. Moreover, a single IP must be specified at each run (i.e., it is not possible to do the analysis for all the IP found in the pcap file).... Well, this is still a proof-of-concept tool!

5 Running AOSD

To run AOSD, you must provide:

- the IP address (-h) of the host to fingerprint.
- the output file (-o) where to store the result (see Section 6).

- the query (-q) to be made, represented by an integer. We currently support only two query:

  0- What is the actual OS?

  1- Is θ the actual OS, for a given θ ?

  2- Does the actual OS belong to a given set Ω?

- when using query 1, an extra argument must be provided to represent θ. This is done by using the -os option following by an integer. Each OS has a unique numerical id as defined in the file "$HOSD_Path\FILE\RULES\allOSList.txt".

- when using query 2, an extra argument must be provided to represent Ω. This is done by using the -oses option followed by a set of integer in brackets (each integer being an OS id). For instance -oses [1,34,42,4] would represent a set of 4 OSes.

Two other arguments are facultative:

- (optional) an initial information file (-i) containing information gathered passively. The format of that file is exactly the same as the output format of POSD. Not providing this file implies no passive information and the active module starts from scratch (i.e., aosd). Specifying such a file implies the use of passive information inside the active module (i.e., hosd)

- (optional) a pcap output file (-w) to record the traffic generated by the specified host (mainly to check if the active tests are executed correctly).

Assuming you are in "$HOSD_Path, you can start AOSD using the following command:

java -Djava.library.path=./LIB/;. -jar AOSD-X.Y.jar -h 10.92.39.4 -o C:\outputAOSD.txt -q 2 -oses [24,324,32] -i C:\outputPOSD.txt

6 Understanding the Output

The output of both the passive and active module follow the same format. Figure 3 provides an example of such an output.

Lines 1 and 7 are pretty much useless. They try to give information about the computation time required, but it is not very accurate (consider them as markers separating the two important sections). The output consists of two important sections: the OS section (lines 2-6) and the port status section (lines 8-9).

Figure 3: Output Format for HOSD

```
OS Computation Times in ms (10.92.39.50):16
10.92.39.50 −−> FreeBSD 4.1
10.92.39.50 −−> FreeBSD 4.0
10.92.39.50 −−> FreeBSD 4.3
10.92.39.50 −−> FreeBSD 4.1.1
10.92.39.50 −−> FreeBSD 4.2
Port Computation Times in ms (10.92.39.50):16
10.92.39.50 −−> 21 tcp open
10.92.39.50 −−> 4444 tcp closed
```

The OS section lists the possible OSes for computer 10.92.39.50 based on the analyzed traffic trace (and maybe on the tests performed). The output in Figure 3 indicates that 10.92.39.50 is running either FreeBSD 4.1, 4.0, 4.3, 4.1.1, or 4.2. HOSD works by eliminating OSes that cannot generate the traffic observed, so the remaining OSes represent the only possibilities (unless HOSD made a mistake). Most of the time, each line in the OS section will contain a single OS. It is possible that a line contains no OS (e.g., just "10.92.39.50 −−> "); this means HOSD was unable to identify the OS (e.g., the traffic trace was empty for posd). It is also possible that a line contains more than 1 OS separated by commas (e.g., "10.92.39.50 −−> FreeBSD 4.0, Windows 2000 sp3"); this means that the IP 10.92.39.50 is behaving like the two OSes simultaneously, the following reasons might explain this:

- 10.92.39.50 is actually a NAT device hiding the two OSes.
- 10.92.39.50 is running an OS unknown by HOSD (and it turns out this OS sometimes behaves like FreeBSd 4.0 and sometimes like Windows 2000 sp3).
- The IP 10.92.39.50 has been reassigned to another computer during the execution of HOSD. It was previously assigned to a FreeBSD 4.0 machine and is now assigned to a Windows 2000 sp3 machine (or vice versa).

The port status section simply lists the status of some open/closed ports, either TCP or UDP.

7 Contact

Please report any bugs, problems, comments, suggestions to:

fgagnon@sce.carleton.ca