Laboratory Assignment 4:
Network Intrusion Detection using Snort

1 Usage of the Virtual Machines

Students are recommended to use their own equipment to do the lab assignments as that will make it easier for them to prepare for the lab. It is also possible to run the virtual machines in the computers at the lab. A more detailed description about the necessary tools, instructions and links to download the virtual machines is available on Canvas.

Follow the steps below before starting with the lab assignment:

1. Download the lab machines and extract them using 7zip
2. Start *Oracle VM VirtualBox Manager*
3. Click on the menu Machine → Add
4. Go to the location where you have downloaded and unzipped the virtual machines, and select the *.vbox* file.
5. Repeat for all VMs needed in the lab.
6. If a snapshot called "lab-start" or similar does not exist yet, create a Snapshot by following the instructions below:
   (a) Select the VM in *Oracle VM VirtualBox Manager*
   (b) Click on *Machine Tools* → Snapshots
   (c) Right-click on *Current State* → Take
   (d) Name the Snapshot *START_EDA491* and click OK.
7. Follow the lab instructions.

If using the lab machines, remember to delete, when you are done, your local copies of the virtual machines (from virtual box first and then from the extracted folder) to ensure the system has free space for the next student.

2 How to ask questions and demonstrate the lab

Labs will be performed on room ED4225. Attending the room physically to do the demonstration is required to pass the labs. All students in the group need to: be physically present; and have

understood, and be able to answer all the questions. Demos can take from 15 minutes to half an hour. Since there may be other groups before you, you should never consider coming to do your demonstration less than one hour before the lab pass ends.

You are also welcome to come to the labs to do the assignment, ask us questions, and even perform your demo once you are done. That said, other than for the demo, it is up to you to decide where and how to perform the assignment. During the lab, TAs will prioritize attention to students doing their demonstration during the time they booked.

If you need to do demos or plan on joining the lab session after 18:00 or before 10:00, make sure to perform your booking at least 24 hours in advance to ensure that a TA will wait for you. Please remove your bookings (or ask us to do it for you if it is already locked) if you cannot attend your booking. We have to plan our workload based on the number of students signing up and we would rather leave early than wait an hour for an student which will not come.

3 Purpose and Scope

In this assignment you will be introduced to network intrusion detection by analysing suspicious network packets and configuring the network intrusion detection system (NIDS) snort to detect and raise alerts for these packets.

In the first part of the exercises (7.1) you will perform some tasks to become familiar with snort. In the second part (7.2), you will analyze malicious traffic and write rules for snort to detect this traffic.

4 Preparation prior the lab

Look at the reference documents listed below before you continue to read the lab manual. It is important that you are well prepared in order to finish the assignment in time.

Reference documents:

- The Snort Manual, `http://manual.snort.org`.
  Important chapters (should be read): 1.1–1.3, and 3.

! IMPORTANT !
Since this assignment is more open than previous assignments, you must read the entire lab manual before you start! A summary of the lab flow is provided in Section 8.

5 Reporting

To pass this assignment, you need to demonstrate that your snort configuration successfully alerts on all suspicious packets, while not raising alerts on legitimate traffic. Also, there are questions throughout the lab manual that should be answered and discussed with a lab assistant when the practical part of the lab is done.

6 Lab setup

This assignment will be performed using two VirtualBox virtual machines (VMs). The VMs images can be downloaded from the course page on Canvas. A short step-by-step guide about importing the VMs is available on Canvas (see *Virtual Machines*).

6.1 System overview

You will use the VM `attacker` to generate malicious traffic and direct it against your host (`firewall`). You will use your host to capture and analyse the traffic and to create rules for Snort. You will then use Snort to detect and alert upon successive malicious packets.

The names, ip addresses, username and password are shown in Table 1 together with a short description. Keep in mind that the VMs are not connected to the internet and make sure you are only scanning one of the two VMs.

VM name	IP-Address	Description	username/password
firewall	10.0.0.3/24	Host running snort	eda491/EDA491!
attacker	10.0.0.4/24	Executes the attack scripts	eda491/EDA491!

Table 1: Name of the VMs, including their ip addresses and username/password combination.

6.2 Snort

Snort is a signature–based intrusion detection system (IDS) which maintains a database of *detection signatures* or *detection rules*. For each network packet that is captured, a comparison is made between the content of the packet and the available signatures in the database. Whenever a match is found, Snort raises an alert to notify the system administrator (or other specified parties) that malicious traffic was detected. Figure 1 illustrates the concept.

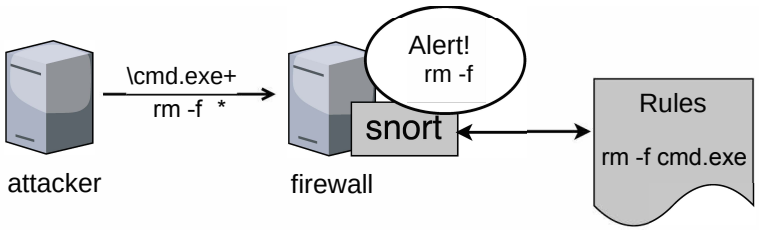

Figure 1: `attacker` sends an attack against the host `firewall` running snort

The Snort configuration file *snort.conf* provides Snort with runtime configuration data, e.g., network variables and rule file locations. The initial snort configuration `snort.conf` located in */home/eda491/netsec-lab4/snort.conf*.

6.3 Netcat

During this assignment you will encounter various network related tools, such as Wireshark, Netcat and Snort. *Whenever you are uncertain of how you should formulate a specific command or how a specific feature works, you should refer to the man pages and the manual.* Snort is the main focus of this lab, and you should already be familiar with Wireshark, so only Netcat will be introduced shortly.

Netcat, or `nc`, is the swiss army knife for networks. It can be used for connecting two peers, either by taking the role of the server and listening on incoming connections, or as the client, initiating connections.

Check out the manual for Netcat (`nc`). Make sure that you know the basics. Then use it to retrieve something, e.g., with `GET`, from the local web server. A useful way to transmit strings is to `echo` the string and then pipe it to `nc` like this:
`$ echo STRING | nc` *`options`*.

7 Lab Assignments

The following exercises are to be done in the lab. Discuss and write down the answers to the questions, and **be prepared to explain your answers** when you have finished the lab.

7.1 Snort in sniffer and packet logger modes

Snort is invoked with "`sudo snort -i <iface> -c <configfile>`", where `<iface>` is the network interface (enp0s3) and `<configfile>` the path to the snort.conf file. Snort can be used in several ways and now you will look at how to start Snort in the *sniffer* and *packet logger* modes.

NOTE: Always start snort in the foreground (i.e., **without** &): If there are errors in your rules, Snort will refuse to start and you will be able to see this on the terminal window. If you start Snort as a daemon, it will exit silently. Also, if Snort is running and it does not terminate when hitting `Ctrl-C`, you might want to use the `pkill -9` command instead.

Start Snort in sniffer mode and be sure to test the different options for capturing link, network and transport headers, and payload. To generate traffic to sniff, log in to `attacker` and transmit the message `"TESTMESSAGE SNORTLAB EDA491"` to your host `firewall` (remember, you need a client and a server).

Q1: To see the string, you had to add the payload option, but when the payload was captured, the output increased in size. Elaborate briefly whether payload data should be captured (consider log sizes, attack coverage, and amount of traffic passing the IDS).

The sniffer mode is seldom useful, unless you are looking for a very specific string and use heavy filtering to discard all other traffic. A more useful mode is the packet logger mode, where the packets are saved to a file and can be inspected later.

Store the logs in `/tmp`. To do this, create a log directory (where XX is your group number):

$ mkdir -p /tmp/nsecXX/log

After this you can give the argument **-l /tmp/nsecXX/log** to snort, to use this directory for logging.

NOTE: When running in packet logger or detection mode, you need to use specific options: add `-i enp0s3` and `-k none`. The parameter `-k none` disregards checksum issues (for the curious: search for "checksum offloading"). In addition, use the `-l` option to denote a directory where you want to store your logs.

Example: `sudo snort -i enp0s3 -c netsec-lab4/snort.conf -l /tmp/nsecXX/log -k none -K <log_format>`

Try out snort's packet logger mode and be sure to store packets in both ASCII and pcap format (in separate runs, only one type of logging is supported at a time). Then, inspect both the ASCII and the pcap files.

Q2: Elaborate briefly on storing log files in ASCII and pcap format. When would it be more suitable to store the files in ASCII, and when would it be more suitable to store the files in pcap format? Is there a difference in the way the log files are named?

So far you have looked at the sniffer and packet logger modes. The real power of Snort, however, lies in its ability to use a set of pattern-matching rules for known malicious network traffic to raise alerts whenever a rule matches. This happens in Snort's *intrusion detection* mode.

7.2 Snort in intrusion detection mode

Before you continue, make sure you have the initial `snort.conf`, and the programs `server1` and `server2` in */home/eda491/netsec-lab4*, which you will need in a little while.

This final part is all about writing Snort rules. Remember that in order for your rules to take effect, you need to restart Snort and provide the path to the configuration file as an option.

NOTE: This part of the lab is quite open. Be sure to **read chapter three in the snort manual**, or this assignment may be very difficult. A summary of the different steps in this assignment is provided in Section 8 for your convenience.

7.2.1 Attack scenario

In the following you will run several simulated attacks from `attacker` against your lab computer. Some of the attacks assume particular services to be running. So before you begin, start `server1` and `server2` located in */home/eda491/netsec-lab4* on `firewall` as follows:

$ sudo netsec-lab4/prepare.sh

You **must** have these services running in order for the attack scripts to work properly.

To launch the attacks against `firewall`, you use the `execute` program in your home folder on `attacker`. There are five simulated attacks, denoted `suspect1` to `suspect5`. Each attack will send one or more packets to your system.

The syntax for execute is as follows: `./execute suspectX ip_target`, where $X \in \{1..5\}$, and `ip_target` is the ip address of your VM running snort.

7.2.2 Lab flow

You begin with an empty rule set which you should expand with the rules you find necessary to alert on the attacks. Use Wireshark to analyse the attacks, and to figure out which packets (or part of a packet) to alert on. Consider what the essence of the attack is, and make sure that you alert on the correct abstraction level (neither too broad, nor too specific).

Rules should be named according to the attack script that they will match, and they should also include your group number. So when writing a rule for attack script 1, the `msg:`–clause of the rule should have the text `"ATTACK 1 nsecnYY"` (where YY is your group number).

To assess your rules you can manually inspect (or `tail -F`) the snort alert file, which is in the same directory as your logs and is called "alert". If nothing shows up in the alert file, you need to tune your configuration.

7.2.3 Lab validation

When you have written rules for all the attacks, answer the remaining lab questions, and contact a teaching assistant to look over your rules. After you demonstrate that your rules as expected, you will discuss your answers.

Q3: In this assignment, you have written specific rules to discover attacks. This paradigm is known as signature-based detection. Another paradigm for intrusion detection is called anomaly-based detection. Elaborate on the advantages and disadvantages of each.

Q4: If you remember only one thing from this lab, what should it be?

8 Lab progress action list

This section will provide an action list for how to proceed when creating your rules. Effectively, this section is a summary of the previous section.

1. Unless already started, start `server1` and `server2` on ports 3000, 8080 and 5555 as indicated earlier.
2. Start capturing packets with wireshark (and set appropriate filters).
3. Launch one attack from `attacker` against your host with the `execute` program.
4. Stop the packet capture.
5. Inspect the recorded traffic and try to identify what constitutes the attack.
6. Add the corresponding rule to the `snort.conf` file and restart snort.
7. Re-launch the attack and inspect the alert file. If there is an alert for the attack, you can move on to the next attack.

Finally, if you have answered all the questions, report the lab:

1. Make sure everyone in the group understood (and is able to answer to) all the questions.
2. Contact a lab supervisor to check your results.