CYBR 8410 – Distributed System Security – 2021 Spring

Lab 3 Part 1

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

NUID: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Please follow the requirements and due date to submit your work.

**Lab Description**: Students will learn Snort architecture and alerts. Students will identify specific attacks with a given capture file. Students will also identify incoming attacks and be able to create custom alerts. Students need to answer the specified questions in the following lab content and steps, and submit your answers in a PDF or Word file.

During the lab, if anything is not clear to you, please contact the instructor.

**Lab Content and Steps:**

1. Updating your machine
   1. Know about this system, e.g., run the following commands in the terminal:

whoami

ifconfig

uname -a

echo $SHELL

cat /proc/cpuinfo

cat /proc/meminfo

Note: press CTRL+ALT+T to open a terminal in Ubuntu.

* 1. Run the following commands and type in your password if you are asked so

sudo apt-get update

sudo apt-get upgrade

Question 1: (15 pts)

1. What do these two commands in 1.2 do? (10 pts)
2. Why do we run them? (5 pts)
3. Installation of Snort

Run the command:

sudo apt-get install snort

! Use TAB key to move focus from one button to another.

! When you are asked for the interface(s) which Snort should listen on, open another terminal and run ifconfig, find out the other name than the “lo”. In my machine, it is “ens33”.

! Leave all other settings in their default values.

Question 2: (25 points)

1. What version of snort is running? (5 pts)
2. Is this a pretty old version, why is it important to install the latest version? (5 pts)
3. Give one example of a really chatty rule. And how can we block these alerts from showing up? (10 pts)
4. Where do we put custom rules? (5 pts)
5. Working with custom rules
   1. Install LAMP server, run the commands:

sudo apt-get install apache2

sudo apt-get install mysql-server

sudo apt-get install php libapache2-mod-php php-mcrypt

sudo /etc/init.d/apache2 restart

Open a web browser and navigate to http://localhost/. You should see a message saying It works!

Run the command to check your PHP:

php -r 'echo "\n\nYour PHP installation is working fine.\n\n\n";'

Now your Apache + SQL + PHP is ready for use!

* 1. Create a username.html webpage under /var/www/html, run the commands:

cd /var/www/html

sudo gedit username.html

Put your full name in that file, save it, and close the editor.

Open a web browser on your own OS (NOT on the Ubuntu virtual machine) and navigate to http://<Your Virtual Machine IP>/username.html. For example, I am using http://192.168.65.128/username.html. (note: in Ubuntu, use the command ifconfig to check the IP address of your Ubuntu)

* 1. Create a rule that will fire when someone access that webpage from the URL in 3.2.

(note: you can simply check URL, or you can try to check your full name in the webpage)

* 1. Restart snort and make that rule fire

Question 3: (30 points)

1. What is the rule that you created? (10 pts)
2. How did you make your rule fired? (10 pts)
3. What is the alert message from the log? (10 pts)

1. Working on SQL injection attacks

Suppose the webpage, username.html, will submit forms to our Apache server, and later make queries to the SQL database. And the forms are going to be submitted in the URL of

http://192.168.65.128/username.html?id=ruizhao where “ruizhao” may be replaced with any other value.

Unfortunately, adversaries may submit the forms in the following way:

http://192.168.65.128/username.html?id=1%E2%80%99%20OR%201=1%20--

Question 4: (30 points)

1. What attack do you think the adversaries was performing in the last URL? (10 pts)
2. What is the rule that you created? (10 pts)
3. What is the alert message from the log? (10 pts)