\input{header}

\begin{document}

\begin{center}

{\LARGE Snort -- Intrusion Detection}

\vspace{0.1in}\\

\end{center}

\copyrightnotice

\section{Overview}

This exercise introduces the use of the snort system

to provide intrusion detection within a

Linux environment. Students will configure simple

snort rules and experiment with a network

intrusion detection system, (IDS).

\section{Lab Environment}

This lab runs in the Labtainer framework,

available at http://my.nps.edu/web/c3o/labtainers.

That site includes links to a pre-built virtual machine

that has Labtainers installed, however Labtainers can

be run on any Linux host that supports Docker containers.

From your labtainer-student directory start the lab using:

\begin{verbatim}

labtainer snort

\end{verbatim}

\noindent A link to this lab manual will be displayed.

\section{Network Configuration}

This lab includes several networked computers as shown in Figure~\ref{fig:topology}.

When the lab starts, you will get virtual terminals, one connected to each

component. The gateway is configured with {\tt iptables} to use NAT to translate

sources addresses of traffic from internal IP addresses, e.g., 192.168.2.1, to

our external address, i.e., 203.0.113.10. The {\tt iptables} in the gateway also

routes web traffic (ports 80 and 443) to the web\\_server component by translating

the externally visible destination address to the internal web server address.

The gateway is also configured to mirror traffic that enters the gateway

via either the 203.0.113.10 link, or the link to the web server. This

mirrored traffic is routed to the {\tt snort} component. This mirroring allows

the snort component to reconstruct TCP sessions between the web server and external

addresses.

The snort component includes the Snort IDS utility. It also includes Wireshark

to help you observe traffic being mirrored to the snort component.

The web server runs Apache and is configured to support SSL for web pages in the

{\tt www.example.com} domain.

The {\tt remote\\_ws} component includes the Firefox browser, and a local

{\tt /etc/hosts} file that maps www.example.com to the external address of the

gateway, i.e., 203.0.113.10. The internal workstation (ws2) also includes Firefox

and an entry in {\tt /etc/hosts} for www.example.com. Both workstations also

include the nmap utility.

\begin{figure}[H]

\begin{center}

\includegraphics [width=0.8\textwidth]{snort.jpg}

\end{center}

\caption{Network topology for the snort lab}

\label{fig:topology}

\end{figure}

\section{Lab Tasks}

It is assumed that the student has received instruction or independent study on

the basic operation of Snort, and the general goals and mechanics of network intrusion detection.

Review the network topology. In particular, consider the {\tt iptables} settings on the gateway.

These can be seen by reviewing the commands in {\tt /etc/rc.local}, which are used to

define the NAT translations and, critically for this lab, mirror traffic to the snort component.

\subsection{Starting and stopping snort}

The Snort utility is installed on the snort component. The home directory includes a {\tt start\\_snort.sh}

script that will start the utility in \textit{Network Intrustion Dection Mode}, and display alerts

to the console. For this lab, you are required to start snort with:

\begin{verbatim}

./start\_snort.sh

\end{verbatim}

When it comes time to stop snort, e.g., to add rules, simply use {\tt CTL-C}.

\subsection{Pre-configured Snort rules}

The Snort utility includes a set of pre-configured rules that create alerts for known

suspicious network activity. The configuration on the snort component is largely as it

exists after initial installation of the snort utility. To see an example of

some of the preconfigured rules, perform an nmap scan of www.example.com from the remote

workstation:

\begin{verbatim}

sudo nmap www.example.com

\end{verbatim}

\noindent Note the alerts displayed at the snort console. The rules that generate these alerts can be seen,

along with all rules, in {\tt /etc/snort/rules/}

\subsection{Write a simple (bad) rule}

Custom rules are typically added to the file at {\tt /etc/snort/rules/local.rules}

Stop snort and add a rule that generates an alert for each packet within a TCP stream. For example:

\begin{verbatim}

alert tcp any any -> any any (msg:"TCP detected"; sid:00002;)

\end{verbatim}

\noindent That rule can be read as: ``Generate an alert whenever a TCP packet from any

address on any port is sent to any address on any port, and include the message tagged as

{\tt msg:}, and give the rule an identifier of 00002.''

Then restart snort. Test this rule by starting Firefox on the remote workstation:

\begin{verbatim}

firefox www.example.com

\end{verbatim}

As you can see, the rule you wrote will overwhelm you with useless information. So,

stop snort and delete the rule.

\subsection{Custom rule for CONFIDENTIAL traffic}

At the Firefox browser, which should be displaying the webpage from www.example.com,

we will display an unpublished web page that we know exists on the website. In particular,

we have heard that the keen minds at the startup company have placed their confidential

business plan at www.example.com/plan.html. Take a look at it.

Now add a rule to your local.rules file on snort that will generate an alert

whenever the text "CONFIDENTIAL" is sent out to the internet. Reference

the snort manual \url{https://www.snort.org/downloads/snortplus/snort\_manual.pdf}

or existing rules to understand how to qualify alerts based on \textit{content}. Be sure to include the

word "CONFIDENTIAL" in the alert message, and give the

rule its own unique sid. After adding the rule, restart snort.

On the browser at the remote workstation, clear your history (Menu / Preferences

Security \& Privacy), and then refresh the plan.html page. You should see an alert at

the snort console.

\subsection{Effects of encryption}

Back at the Firefox browser, again clear the browser history. Now alter the URL to make

use of the web server SSL function. Change the url to https://www.example.com/plan.html.

Do you see a new snort alert? Why?

One solution to this problem is to use a reverse proxy in front of the web server. This

reverse proxy would handle the incoming web traffic and manage the SSL connections.

The web server would then receive only clear-text HTTP traffic, and outgoing traffic from the web server

could then be mirrored to the IDS. We will not pursue that solution in this lab.

\subsection{Watching internal traffic}

Go to the ws2 (mary) component and run nmap:

\begin{verbatim}

sudo nmap www.example.com

\end{verbatim}

What do you see on the snort component? Does it include the {\tt ICMP PING NMAP}

alert that you saw when the remote workstation ran nmap? Why not?

Go to the gateway component and edit the {\tt /etc/rc.local} script so that traffic

from Mary's workstation is mirrored to the snort component. You can do this by adding

this line to the section of that file that defines the packet mirroring:

\begin{verbatim}

iptables -t mangle -A PREROUTING -i $lan2 -j TEE --gateway 192.168.3.1

\end{verbatim}

Then run the script to replace the {\tt iptables} rules with your new rules:

\begin{verbatim}

sudo /etc/rc.local

\end{verbatim}

Now restart snort and again run nmap from mary's ws2 computer.

\subsection{Distinguishing traffic by address}

Start Firefox on mary's ws2 computer to view the confidential business plan:

\begin {verbatim}

firefox www.example.com/plan.html

\end {verbatim}

Then observe the snort console. This will not do! The keen minds at the startup need to view their confidential

business plan without IDS alerts firing off. But they do want to monitor

internal computers for suspicious traffic, e.g., nmap scans. In this task, you

will adjust your snort rule so that the CONFIDENTIAL alert only fires when the plan

is accessed by addresses outside of

the site.

If you review rules found in the {\tt/etc/snort/rules} directory, you will see that rules

have the general form of:

\begin{verbatim}

alert <protocol> <source\_addr> <src\_port> -> \

<dest\_addr> <dest\_port> <rule options in parens>

\end{verbatim}

The snort rules include two address fields: {\tt source\\_addr} and {\tt dest\\_addr}.

These addresses are used to check the

source from which the packet originated and the destination of the packet. The address

may be a single IP address or a network address. You likely have used the \textit{any} keyword to apply a

rule on all addresses. For network addresses, the address is followed by a slash character

and number of bits in the netmask. For example, a network address of 192.168.2.0/24

represents C class network 192.168.2.0 with 24 bits in the network mask.

Note that as a result of our use of NAT, all traffic from the web server destined for an

external address will have a destination address of the gateway, (i.e., 192.168.1.10),

while web traffic destined for internal users will have destination addresses that match the internal user.

For this task, you must set your snort rules and traffic mirroring such that:

\begin{enumerate}

\item External access to the business plan generates an alert;

\item Internal access to the business plan does not generate an alert;

\item External or internal use of nmap will generate an ICMP NMAP PING alert.

\end{enumerate}

Your must test each of these criteria during a single snort session, i.e., if you change

a snort rule, or port mirroring, you must restart your tests.

\section{Submission}

After finishing the lab, go to the terminal on your Linux system that was used to start the lab and type:

\begin{verbatim}

stoplab snort

\end{verbatim}

When you stop the lab, the system will display a path to the zipped lab results on your Linux system. Provide that file to

your instructor, e.g., via the Sakai site.

\end{document}