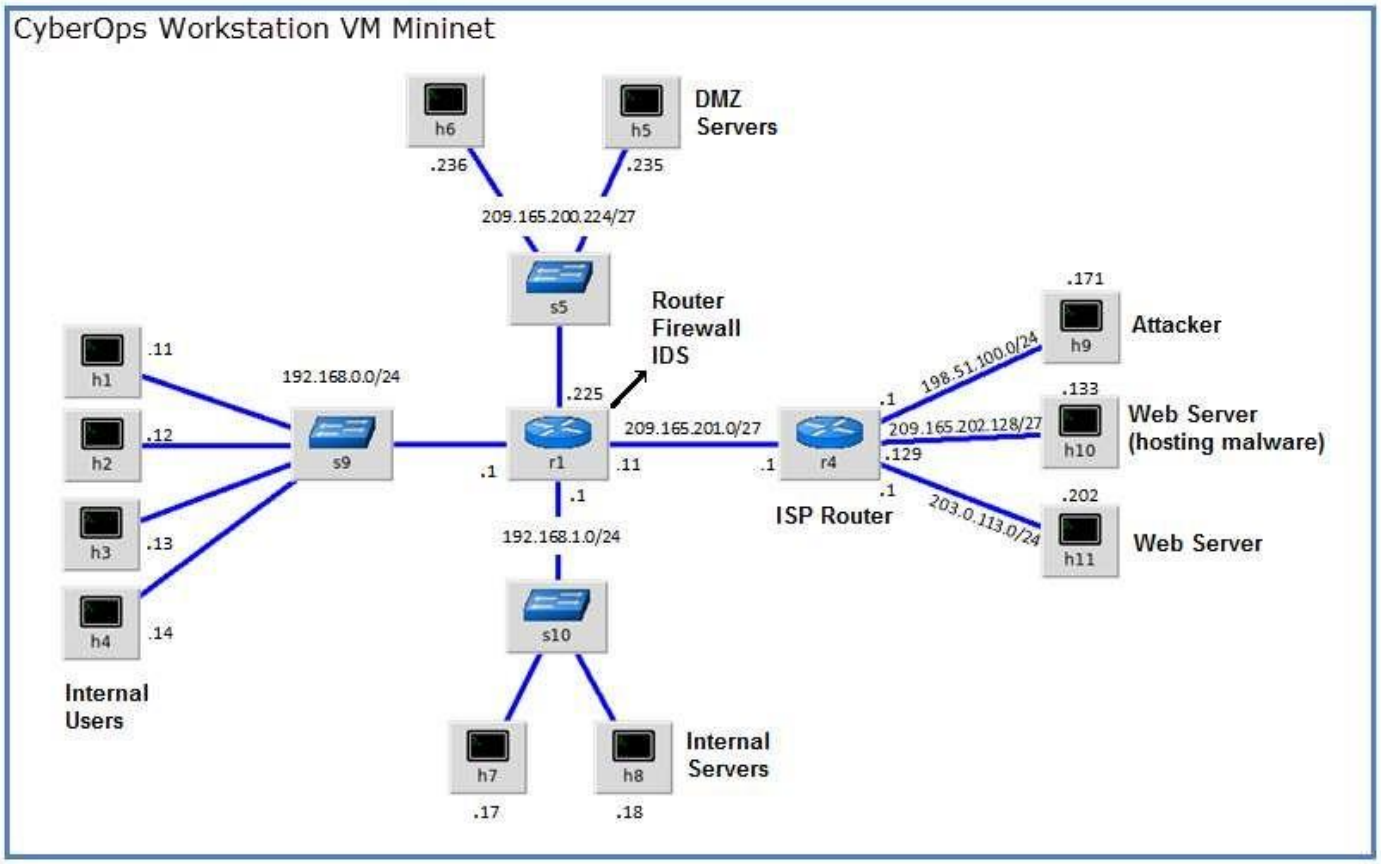
Práctica de laboratorio: Reglas de Snort y de firewalls
Topología



Objetivos

Parte 1: Preparar el entorno virtual

Parte 2: El firewall y los archivos de registro de IDS

Parte 3: Finalizar y desactivar el proceso de Mininet

Antecedentes / Escenario

En una red de producción segura, las alertas de red son generadas por diversos tipos de dispositivos como dispositivos de seguridad, firewalls, dispositivos IPS, routers, switches y servidores, entre otros. El problema es que no todas las alertas se crean de la misma manera. Por ejemplo: las alertas generadas por un servidor las generadas por un firewall serán diferentes y tendrán distintos contenidos y formatos.

En esta práctica de laboratorio, se familiarizará con las reglas de firewall y las firmas de IDS.

Recursos necesarios

- Máquina virtual CyberOps Workstation
- Conexión a Internet

Nota: En esta práctica de laboratorio, la VM CyberOps Workstation es un contenedor para alojar el entorno de Mininet que se muestra en la topología. Si se recibe un error de memoria en un intento de ejecutar cualquier comando, cierre el paso, vaya a la configuración de VM y aumente la memoria. El valor predeterminado es 1 GB; intente cambiarlo a 2 GB.

Instrucciones

Parte 1: Preparar el entorno virtual

a. Ejecutar **Oracle VirtualBox** y cambie la **CyberOps Workstation** a modo puente(Bridged), si es necesario Seleccionen **Machine > Settings > Network** (Máquina > Configuración > Red). En **Conectado a**, seleccione **Adaptador de puente** (o, si utiliza wifi con un proxy, puede que necesite **el adaptador NAT**), y haga clic en **Aceptar**.

b. Abran la **VM CyberOps Workstation**, después un terminal, y configuren su red; para ello, ejecuten el script **configure_as_dhcp.sh**.

Como el script requiere privilegios de usuario avanzado, introduzcan la contraseña correspondiente al usuario **analyst**.

```
[analyst@secOps ~]$ sudo ./lab.support.files/scripts/configure_as_dhcp.sh
[sudo] contraseña para analyst:
[analyst@secOps ~]$
```

c. Utilicen el comando **ifconfig** para verificar que la **VM CyberOps Workstation** ahora tenga una dirección IP en sus redes locales. También pueden probar la conectividad a un servidor web público si emiten un ping a www.cisco.com. Presionen **Ctrl+C** para detener los pings.

```
[analyst@secOps ~]$ ping www.cisco.com
PING e2867.dsca.akamaiedge.net (23.204.15.199) 56(84) bytes of data.
64 bytes from a23-204-15-199.deploy.static.akamaitechnologies.com
(23.204.15.199): icmp_seq=1 ttl=54 time=28.4 ms
64 bytes from a23-204-15-199.deploy.static.akamaitechnologies.com
(23.204.15.199): icmp_seq=2 ttl=54 time=35.5 ms
^C
--- e2867.dsca.akamaiedge.net ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 28.446/32.020/35.595/3.578 ms
```

Parte 2: El firewall y los archivos de registro de IDS

A menudo se implementan firewalls y Sistemas de detección de intrusiones (Intrusion Detection Systems, IDS) para automatizar parcialmente la tarea del monitoreo de tráfico. Tanto los firewalls como los IDS comparan el tráfico entrante con reglas administrativas. Los firewalls suelen comparar el encabezado de un paquete con un conjunto de reglas mientras que los IDS utilizan la carga útil del paquete para compararla con los conjuntos de reglas. Como los firewalls y los IDS aplican las reglas predefinidas a diferentes porciones del paquete IP, las reglas de los IDS y de los firewalls tienen estructuras diferentes.

Aunque hay una diferencia en la estructura de las reglas, sigue habiendo ciertas similitudes entre sus componentes. Por ejemplo: tanto las reglas de un firewall como las de un IDS contienen componentes de correspondencia y componentes de acciones. Las acciones se realizan una vez detectada una coincidencia.

- **Componente de correspondencia**: especifica los elementos de interés del paquete, por ejemplo: origen del paquete, destino del paquete, protocolos y puertos de la capa de transporte y los datos incluidos en la carga útil del paquete.

- **Componente de acciones**: especifica qué debe hacerse con el paquete que coincida con un componente, por ejemplo: aceptar y reenviar el paquete, descartar el paquete, o enviar el paquete a un conjunto de reglas secundario para profundizar su inspección.

Un diseño de firewall común es descartar paquetes de manera predeterminada y especificar manualmente el tráfico que se debe permitir. Conocido como descartar por defecto, este diseño tiene la ventaja de proteger la red de protocolos y ataques desconocidos. Como parte de este diseño, es común registrar los eventos de los paquetes descartados porque se trata de paquetes no permitidos explícitamente y, por lo tanto, infringen las políticas de la organización. Tales eventos deben registrarse para próximos análisis.

Paso 1: Monitoreo de archivos de registro de un IDS en tiempo real

a. En la **VM CyberOps Workstation VM**, ejecuten el script para iniciar **mininet**.

```
[analyst@secOps ~]$ sudo
./lab.support.files/scripts/cyberops_extended_topo_no_fw.py
[sudo] contraseña para analyst:
*** Adding controller
*** Add switches
*** Add hosts
*** Add links
*** Starting network
*** Configuring hosts
R1 R4 H1 H2 H3 H4 H5 H6 H7 H8 H9 H10 H11
*** Starting controllers
*** Starting switches
*** Add routes
*** Post configure switches and hosts
*** Starting CLI:
mininet>
```

El cursor de **mininet** debería aparecer en la pantalla; eso indica que **mininet** está listo para recibir comandos.

b. En el cursor de **mininet**, abran un shell en **R1** con el siguiente comando:

```
mininet> xterm R1
mininet>
```

El shell de **R1** se abre en una ventana del terminal con texto negro y fondo blanco. ¿Qué usuario ha iniciado sesión en ese shell? ¿Qué nos lo indica?

c. En el shell de **R1**, inicien el IDS basado en Linux: Snort.

```
[root@secOps analyst]# ./lab.support.files/scripts/start_snort.sh
Running in IDS mode
--== Initializing Snort ==--
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "/etc/snort/snort.conf"
<output omitted>
```

Nota: No verá ningún indicador porque Snort ahora se está ejecutando en esta ventana. Si por cualquier motivo, Snort deja de ejecutarse y aparece el indicador **[root@secOps analysts]#**, vuelva a ejecutar el script para iniciar Snort. Snort debe estar en ejecución para poder capturar alertas más adelante en este laboratorio.

d. En el **prompt de mininet de la máquina virtual CyberOps Workstation**, abra shells para los hosts **H5** y **H10**.

```
mininet> xterm H5
mininet> xterm H10
mininet>
```

e. **H10** simulará ser un servidor de Internet que aloja malware. En **H10**, ejecute el script **mal_server_start.sh** para iniciar el servidor.

```
[root@secOps analyst]# ./lab.support.files/scripts/mal_server_start.sh
[root@secOps analyst]#
```

f. En **H10**, utilicen **netstat** con las opciones **-tunpa** para verificar que el servidor web se esté ejecutando. Cuando se utiliza como se indica arriba, **netstat** genera una lista de todos los puertos asignados a servicios en este momento:

```
[root@secOps analyst]# netstat -tunpa
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name
tcp 0 0 0.0.0.0:6666 0.0.0.0:* LISTEN 1839/nginx: master
[root@secOps analyst]#
```

Tal como se ve en el resultado anterior, el servidor web ligero **nginx** se está ejecutando y está escuchando conexiones en el puerto TCP 6666.

g. Se está ejecutando una instancia de Snort en la ventana del terminal de **R1**. Para introducir más comandos en **R1**, abra otro terminal de **R1**; para ello, vuelva a introducir **xterm R1** en la ventana del terminal de la **máquina virtual CyberOps Workstation**. También es posible que quieran organizar las ventanas del terminal para poder ver e interactuar con cada dispositivo.

h. En la nueva ficha del terminal de **R1**, ejecuten el comando **tail** con la opción **-f** para monitorear el archivo **/var/log/snort/alert** en tiempo real. Es en este archivo que se configura Snort para registrar alertas.

```
[root@sec0ps analyst]# tail -f /var/log/snort/alert
```

Como todavía no se registró ninguna alerta, el archivo de registro debería estar vacío. Sin embargo, si ya han realizado esta práctica de laboratorio, es posible que aparezcan entradas de alertas extrañas. En cualquier caso, no verán ningún cursor después de escribir este comando. En esta ventana se mostrarán las alertas a medida que tengan lugar.

i. En **H5**, utilicen el comando **wget** para descargar un archivo de nombre **W32.Nimda.Amm.exe**. Diseñada para descargar contenido a través de HTTP, **wget** es una excelente herramienta para descargar archivos desde servidores web directamente desde la línea de comandos.

```
[root@secOps analyst]# wget 209.165.202.133:6666/W32.Nimda.Amm.exe
--2017-04-28 17:00:04-- http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... connected.
HTTP request sent, awaiting response... 200 OK
Length: 345088 (337K) [application/octet-stream]
Saving to: 'W32.Nimda.Amm.exe'

W32.Nimda.Amm.exe 100%[=========================================>] 337.00K --.-KB/s
in 0.02s
```

```
2017-04-28 17:00:04 (16.4 MB/s) - 'W32.Nimda.Amm.exe' saved [345088/345088]

[root@secOps analyst]#
```

¿Qué puerto se utiliza al comunicarse con el servidor web que aloja malware? ¿Qué nos lo indica?

¿Se descargó completamente el archivo?

¿El IDS generó alguna alerta relacionada con la descarga del archivo?

j. Como el archivo malicioso estaba transitando por **R1**, el IDS (Snort) pudo inspeccionar su carga útil. La carga útil coincidió con al menos una de las firmas configuradas en Snort y generó una alerta en la segunda ventana del terminal de **R1** (la ficha en la que se está ejecutando **tail -f**). A continuación se muestra la entrada de la alerta. Sus marcas de hora serán diferentes:

```
04/28-17:00:04.092153 [**] [1:1000003:0] Malicious Server Hit! [**] [Priority: 0]
{TCP} 209.165.200.235:34484 -> 209.165.202.133:6666
```

En función de la alerta que se muestra arriba, ¿cuáles fueron las direcciones IPv4 de origen y de destino que se utilizaron en la transacción?

En función de la alerta que se muestra arriba, ¿cuáles fueron los puertos de origen y de destino que se utilizaron en la transacción?

En función de la alerta que se muestra arriba, ¿cuándo tuvo lugar la descarga?

En función de la alerta que se muestra arriba, ¿qué mensaje registró la firma del IDS?

En **H5**, utilicen el comando **tcpdump** para capturar el evento y volver a descargar el archivo malicioso y así poder capturar la transacción. Emitan el siguiente comando para iniciar la captura de paquetes:

```
[root@secOps analyst]# tcpdump –i H5-eth0 –w nimda.download.pcap &
[1] 5633
[root@secOps analyst]# tcpdump: listening on H5-eth0, link-type EN10MB (Ethernet),
capture size 262144 bytes
```

El comando anterior le ordena a tcpdump que capture paquetes en la interfaz **H5-eth0** y que guarde la captura en un archivo de nombre **nimda.download.pcap**.

El símbolo **&** del final le indica al shell que ejecute **tcpdump** en segundo plano. Sin este símbolo, **tcpdump** impediría el uso del terminal mientras se está ejecutando. Observen el **[1] 5633**; indica que se envió un proceso al segundo plano y que su ID de proceso (PID) es 5366. Sus PID muy probablemente serán diferente.

k. Presionen **INTRO** un par de veces para recuperar el control del shell mientras **tcpdump** se ejecuta en segundo plano.

l. Ahora que **tcpdump** está capturando paquetes, vuelvan a descargar el malware. En **H5**, vuelvan a ejecutar el comando o utilicen la flecha hacia arriba para recuperarlo del centro del historial de comandos.

```
[root@secOps analyst]# wget 209.165.202.133:6666/W32.Nimda.Amm.exe
--2017-05-02 10:26:50-- http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... connected.
HTTP request sent, awaiting response... 200 OK
Length: 345088 (337K) [application/octet-stream]
Saving to: 'W32.Nimda.Amm.exe'

W32.Nimda.Amm.exe 100%[===================>] 337.00K --.-KB/s in 0.003s

02/05/2017 10:26:50 (105 MB/s) - 'W32.Nimda.Amm.exe' saved [345088/345088]
```

m. Lleve **tcpdump** al primer plano con el comando **fg** para detener la captura. Como **tcpdump** era el único proceso que se había enviado a segundo plano, no es necesario especificar el PID. Detengan el proceso de **tcpdump** con **Ctrl+C**. El proceso de **tcpdump** se detiene y exhibe una resumen de la captura. La cantidad de paquetes puede diferir en sus capturas.

```
[root@secOps analyst]# fg
tcpdump -i h5-eth0 -w nimda.download.pcap
^C316 packets captured
316 packets received by filter
0 packets dropped by kernel
[root@secOps analyst]#
```

n. En **H5**, utilicen el comando **ls** para verificar que el archivo pcap realmente se haya guardado en el disco y que su tamaño sea mayor que cero:

```
[root@secOps analyst]# ls -l
total 1400
drwxr-xr-x 2 analyst analyst 4096 Sep 26 2014 Desktop
drwx------ 3 analyst analyst 4096 Jul 14 11:28 Downloads
drwxr-xr-x 8 analyst analyst 4096 Jul 25 16:27 lab.support.files
-rw-r--r-- 1 root root 371784 Aug 17 14:48 nimda.download.pcap
drwxr-xr-x 2 analyst analyst 4096 Mar 3 15:56 second_drive
-rw-r--r-- 1 root root 345088 Apr 14 15:17 W32.Nimda.Amm.exe
-rw-r--r-- 1 root root 345088 Apr 14 15:17 W32.Nimda.Amm.exe.1
 [root@secOps analyst]#
```

Nota: Es posible que su lista de directorios tenga otra combinación de archivos, pero de todos modos debería ver el archivo **nimda.download.pcap**.

¿Qué beneficios puede aportar este PCAP al analista especializado en seguridad?

Nota: El archivo PCAP se analizará en otra práctica de laboratorio.

Paso 2: Afinar reglas de firewall en función de alertas de un IDS

En el paso 1, empezamos con un servidor malicioso de Internet. Para evitar que otros usuarios lleguen a ese servidor, se recomienda bloquearlo en el firewall de perímetro.

En la topología de esta práctica de laboratorio, **R1** no solo está ejecutando un IDS sino también un firewall basado en Linux muy popular: **iptables**. En este paso, bloquearán el tráfico al servidor malicioso identificado en el Paso 1; para ello, editarán las reglas de firewall presentes en **R1**.

Nota: Aunque el estudio completo de **iptables** está fuera del alcance de este curso, la estructura lógica y de reglas básica de **iptables** es relativamente sencilla.

El firewall **iptables** utiliza los conceptos de *cadenas* y *reglas* para filtrar tráfico.

El tráfico que ingresa al firewall y tiene como destino al propio dispositivo de firewall es manejado por la cadena **INPUT**. Como ejemplo de este tráfico podemos mencionar a los paquetes de ping provenientes de cualquier otro dispositivo en cualquier red y enviados a cualquiera de las interfaces del firewall.

El tráfico que se origina en el propio dispositivo de firewall y tiene como destino cualquier otro lugar es manejado por la cadena **OUTPUT**. Un ejemplo de este tráfico son las respuestas de ping generadas por el propio dispositivo de firewall.

El tráfico originado en cualquier otro lugar y que atraviesas el dispositivo de firewall es manejado por la cadena **FORWARD**. Un ejemplo de este tráfico son los paquetes que está enrutando el firewall.

Cada cadena puede tener sus propias reglas independientes que especifican cómo se debe filtrar el tráfico. Una cadena puede tener prácticamente cualquier cantidad de reglas, incluso ninguna.

Las reglas se crean para comprobar características específicas de los paquetes; eso permite que los administradores generen filtros muy integrales. Si un paquete no coincide con una regla, el firewall pasa a la siguiente y repite la comprobación. Si se encuentra una coincidencia, el firewall toma la medida definida en la regla pertinente. Si se han comprobado todas las reglas de una cadena y no se encontró ninguna coincidencia, el firewall toma la medida especificada en la política de la cadena, que suele ser permitir que el paquete circule o rechazarlo.

a. En la **VM CyberOps Workstation**, inicien una tercera ventana del terminal de R1.

```
mininet > xterm R1
```

b. En la nueva ventana del terminal de **R1**, utilicen el comando **iptables** para generar una lista de las cadenas y sus reglas en uso:

```
[root@secOps ~]# iptables -L -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination

Chain FORWARD (policy ACCEPT 6 packets, 504 bytes)
 pkts bytes target prot opt in out source destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination

[root@secOps ~]#
```

¿Qué cadenas está utilizando **R1** en este momento?

c. Las conexiones al servidor malicioso generan paquetes que deben atravesar el firewall **iptables** en **R1**. Los paquetes que atraviesan el firewall son manejados por la regla de FORWARD y, por lo tanto, esa es la cadena que recibirá la regla de bloqueo. Para impedir que las computadoras de los usuarios se conecten al servidor malicioso identificado en el Paso 1, agreguen la siguiente regla a la cadena FORWARD en **R1**:

```
[root@secOps ~]# iptables -I FORWARD -p tcp -d 209.165.202.133 --dport 6666 -
j DROP
[root@secOps ~]#
```

Donde:

- **-I FORWARD**: inserta una regla nueva en la cadena FORWARD.
- **-p tcp**: especifica el protocolo TCP.
- **-d 209.165.202.133**: especifica el destino del paquete.
- **-dport 6666**: especifica el puerto de destino.
- **-j DROP**: define la acción a descartar.

d. Vuelvan a utilizar el comando **iptables** para asegurarse de que la regla se haya agregado a la cadena FORWARD. La VM CyberOps Workstation puede demorar algunos segundos para generar la salida:

```
[root@secOps analyst]# iptables -L -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination
    0 0 DROP tcp -- any any anywhere 209.165.202.133 tcp dpt:6666

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target prot opt in out source destination
[root@secOps analyst]#
```

e. En **H5**, traten de descargar el archivo nuevamente:

```
[root@secOps analyst]# wget 209.165.202.133:6666/W32.Nimda.Amm.exe
--2017-05-01 14:42:37-- http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... failed: Connection timed out.
Retrying.

--2017-05-01 14:44:47-- (try: 2) http://209.165.202.133:6666/W32.Nimda.Amm.exe
Connecting to 209.165.202.133:6666... failed: Connection timed out.
Retrying.
```

Si es necesario, presionen **Ctrl+C** para cancelar la descarga.

¿Esta vez la descarga se completó correctamente? Explique.

¿Cuál sería un enfoque más agresivo (pero válido a la vez) cuando se está bloqueando el servidor malicioso?

Parte 3: Finalizar y desactivar el proceso de Mininet

a. Diríjanse al terminal que se utilizó para iniciar Mininet. Introduzca **quit** en la ventana del terminal principal de la VM CyberOps para finalizar Mininet.

b. Después de salir de Mininet, limpie los procesos que inició Mininet. Introduzcan la contraseña **cyberops** cuando el sistema se los solicite.

```
[analyst@secOps scripts]$ sudo mn -c
[sudo] contraseña para analyst:
```