Módulo 4.1

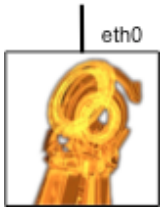
Redes y Seguridad
Gr. Ing. Informática
J.L. Vázquez Poletti

Objetivo

En este módulo sacaremos más rendimiento a la distribución de seguridad **Backtrack**, a la vez que realizamos ataques a diversos protocolos.

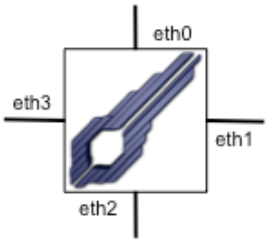
Punto de partida

Como parte de la formación del Cuerpo de Ingenieros de la Alianza, se realizarán diferentes maniobras con objetivos reales. Se ha transferido a nuestra omniherramienta una solución de seguridad actualizada (**RyS-Backtrack**), que se conectará a la red que se vayan indicando al comienzo de la maniobra (**Red1**, **Red2**, **Red3** o **Red4**) usando su interfaz **eth0**:



El resto de elementos que encontraremos son los siguientes:

1. Routers (**RyS-Router1/2**): Tienen sus cuatro interfaces dispuestas de la siguiente manera:

   Los interfaces se conectan de la siguiente manera: **eth0** a **Red1**, **eth1** a **Red2**, **eth2** a **Red3** y **eth3** a **Red4**.

2. Plataformas Geth (**RyS-Geth01/10**): Se conectan a la red a través de la interfaz **eth0**.

Máquinas Virtuales

- RyS-Backtrack (root:toor)
- RyS-Geth01 (ubuntu:reverse)
- RyS-Geth10 (ubuntu:reverse)
- RyS-Router1 (ubntu:reverse)
- RyS-Router2 (ubntu:reverse)

Más información

Backtrack: http://www.backtrack-linux.org/

Wireshark: http://www.wireshark.org/

ARPSpoof: http://arpspoof.sourceforge.net/

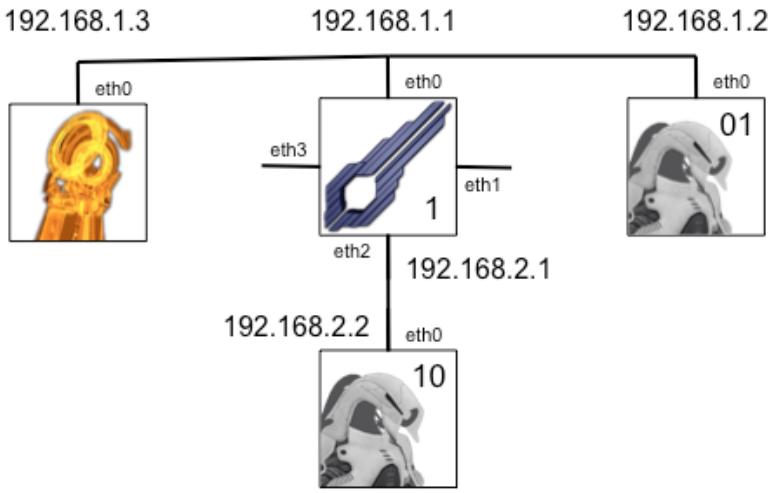
Fragroute(r): http://www.monkey.org/~dugsong/fragroute/

HPing: http://www.hping.org/

Objetivo 1: Escenario

El objetivo es interceptar las comunicaciones entre las dos plataformas Geth pero sin interferirlas, a fin de obtener información valiosa sin levantar sospechas. Al encontrarse una de ellas en la misma red que nosotros, podremos trabajar a nivel de enlace.

La red en cuestión tiene la siguiente topología:



1. El interfaz de red de **RyS-Backtrack** debe tener denegado el modo promiscuo (opciones avanzadas de **Red** en **VirtualBox**).
2. Tanto **RyS-Backtrack** como **RyS-Geth01/02** deben conectarse a la red indicada según su conexión con **RyS-Router1**.
3. **RyS-Router1** es el gateway por defecto del resto de elementos. Para asignarlo, ejecutar: **route add default gw <*IP_Router*> <*Interfaz*>**
4. **RyS-Router1** no tiene activado el IP forwarding. Para activarlo, ejecutar **sysctl -w net.ipv4.ip_forward=1**.

Antes de continuar, comprobar que todas las máquinas pueden verse entre sí usando el comando **ping <*IP*>**.

Objetivo 1: Manejo básico del *sniffer*

La distribución Backtrack cuenta con varias herramientas de análisis de tráfico. Para acceder a ellas, se debe ir al menú desplegable del icono con el logotipo de la distribución (). Y de ahí, **Information Gathering->Network Analysis->Network Traffic Analysis**.

1. Abrir **wireshark**.

   NOTA: las aplicaciones del menú de Backtrack son también ejecutables desde la terminal. El menú permite una búsqueda lógica de las herramientas.

2. Escoger el interfaz conectado y configurado.
3. Siempre desde **RyS-Backtrack**, hacer ping al resto de máquinas.

4. Usar el filtro para mostrar solamente los mensajes de los protocolos **icmp** o **arp**.
5. Identificar el contenido de cada mensaje según el protocolo (desplegar los campos).

¿Cuál es la dirección MAC de broadcast?

6. Reiniciar la captura (quinto icono desde la izquierda en la barra de herramientas de **wireshark**).
7. Desde **RyS-Geth01** hacer ping a **RyS-Geth10** y viceversa.

¿Qué diferencias hay en el tráfico ICMP y ARP entre los dos escaneos?
¿A qué se debe este fenómeno?

Objetivo 1: Suplantación del router

El protocolo escogido para el ataque es el ARP y éste es el plan de actuación:

1. Convencer a **RyS-Geth01** que **RyS-BackTrack** es el router.
2. Reenviar las tramas recibidas de **RyS-Geth01** al router legítimo una vez inspeccionadas.

Desde **RyS-Backtrack**:

1. Activar el forwarding normal con el comando **fragrouter**.
2. Suplantar a **RyS-Router1** con el comando **arpspoof**. El objetivo es **RyS-Geth01**.
3. Reiniciar la captura con **wireshark**.

Para verificar que el ataque ha sido un éxito, hacer ping desde **RyS-Geth01** a **RyS-Get02** y viceversa.

¿El tráfico en qué sentido está afectado con este ataque? ¿Por qué?
¿Por qué la repetición constante de mensajes generados por **arpspoof**?
Si no se hubiera especificado un objetivo, **arpspoof** mandaría mensajes a todas las máquinas de la red. ¿Es esto recomendable?
¿Por qué el forwarding?

Objetivo 1: Captura de información sensible

Una vez comprobado que el ataque ha sido un éxito, es el momento de obtener datos sensibles del tráfico entre las dos plataformas Geth.

1. Reiniciar la captura de tráfico en wireshark.
2. Desde **RyS-Geth01** conectarse a **RyS-Geth10** por **telnet** (cuenta de usuario **ubuntu**). Ejecutar algunos comandos.
3. Obtener el usuario y la contreseña introducidos desde wireshark.

CONSEJO: filtrar por protocolo.

4. Hacer la conexión inversa desde **RyS-Geth10** y buscar los mismos datos.

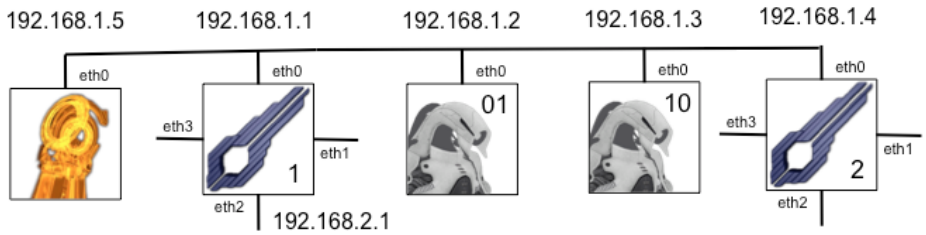
¿Qué diferencia hay entre los datos obtenidos por el tráfico capturado en ambos sentidos? ¿Por qué?

5. Reiniciar la captura de tráfico en **wireshark**.
6. Realizar la conexión ahora a través de **ftp** (cuenta de usuario **ubuntu**) y navegar por los directorios.

¿Qué diferencia hay con el tráfico anteriormente capturado?

Objetivo 2: Escenario

La plataforma **RyS-Geth10** tiene sospechas del ataque anterior por lo que se ha recolocado en la misma red que la otra plataforma y nosotros. Además, se ha añadido un nuevo router. Se trata de una ocasión estupenda para hacer pruebas a baja escala de ataques de denegación de servicio que pertenecen a capas superiores.



1. **RyS-Router2** solo tiene requiere que se le asigne la IP de **eth0**. No es necesario configuración de rutas.
2. **RyS-Router1** sigue siendo el router por defecto del resto de máquinas.
3. Ejecutar en todas las máquinas **menos RyS-Backtrack** el comando:

   sysctl -w net.ipv4.icmp_echo_ignore_broadcasts=0

Objetivo 2: Smurf

En este ataque se bombardeará la red con paquetes ICMP request falsificados. Estos paquetes tendrán como origen **RyS-Geth01** y como destino, la dirección de broadcast.

Para ello, utilizaremos una herramienta muy potente llamada **hping3** que ya está instalada en la distribución **Backtrack**.

1. Descubrir con el comando **ping** las máquinas de la red que responden a la dirección de broadcast. Comprobar que están todas las conectadas (menos **RyS-Backtrack**).
2. Desde **RyS-Geth10**:
   a. Hacer **ping** con 25 paquetes y anotar los datos estadísticos.
   b. Ejecutar **time telnet <RyS-Geth01>** y nada más iniciar sesión, salir de ella (**exit**). Anotar el tiempo.

3. Ejecutar **hping3** con los siguientes parámetros:
    a. Origen: **RyS-Geth01**.
    b. Destino: IP de broadcast.
    c. Envío de paquetes lo más rápido posible (**flood**)
4. Esperar unos instantes y desde **RyS-Geth10**:
    a. Volver a hacer ping y anotar los datos estadísticos.
    b. Volver a ejecutar **time telnet <RyS-Geth01>** y nada más iniciar sesión, salir de ella (**exit**). Anotar el tiempo.

¿Cuál es la contramedida a este ataque?

Objetivo 2: SYN Flood

Las plataformas han reaccionado al ataque con lo que han desplegado una mejora de seguridad (la contramedida de la solución a la anterior pregunta) en los elementos de red que impide la ejecución ataque. Por tanto, hay que realizar otro ataque.

Usando el comando **sysctl**, asignar los siguientes valores a los parámetros de red del Kernel:

- **net.ipv4.tcp_syncookies=”0”**
- **net.ipv4.tcp_fin_timeout=”999999”**
- **net.ipv4.tcp_max_syn_backlog="2”**
- **net.ipv4.tcp_synack_retries=”9999999999”**

Este ataque está dirigido a un puerto abierto, que en este caso es el del telnet (**23**). El objetivo es no permitir que **RyS-Geth10** se comunique con **RyS-Geth01** a través de ese servicio.

1. Ejecutar **hping3** con los siguientes parámetros:
    a. Origen: **RyS-Geth10**.
    b. Destino: **RyS-Geth01** (puerto **23**).
    c. Envío de paquetes lo más rápido posible (**flood**).
    d. Opción **SYN**.
    e. Packet count = 1.
2. Esperar un buen rato y desde **RyS-Geth10**:
    a. Comparar el retraso en el **ping** a **RyS-Geth01** con el del otro ataque.
    b. Medir los tiempos de conexión con **RyS-Geth01** por **telnet**. Realizar la conexión un mínimo de 4 veces seguidas.

¿Qué diferencias hay en este ataque y el anterior?
¿Cuál es la contramedida de este ataque?
A pesar de esta contramedida, ¿qué lo puede hacer imbatible?

NOTA: Hay que tener en cuenta que la ejecución de estos ataques se realiza en una red virtual, con lo que su efecto no es el mismo que el obtenido en una red real.