Ejercicios de Seguridad en Redes
Escucha y análisis de tráfico + Escaneo de puertos

SSI 2018/19

20 de noviembre de 2018

Índice

1. Entorno de pruebas

1.1. Software de virtualización VIRTUALBOX

En estas prácticas se empleará el software de virtualización VIRTUALBOX para simular los equipos GNU/Linux sobre los que se realizarán las pruebas.

- Página principal: `http://virtualbox.org`
- Más información: `http://es.wikipedia.org/wiki/Virtualbox`

1.2. Imágenes a utilizar

1. Scripts de instalación
   - para GNU/Linux: ejercicio-nmap.sh
     `alumno@pc: $ sh ejercicio-nmap.sh`
   - para MS windows: ejercicio-nmap.ps1
     `Powershell.exe -executionpolicy bypass -file ejercicio-nmap.ps1`

Notas:

- Se pedirá un identificador (sin espacios) para poder reutilizar las versiones personalizadas de las imágenes creadas (usad por ejemplo el nombre del grupo de prácticas o el login LDAP)
- En ambos scripts la variable `$DIR_BASE` especifica donde se descargarán las imágenes y se crearán las MVs.
  Por defecto en GNU/Linux será en `$HOME/SSI1819` y en Windows en `C:/SSI1819`.
  Puede modificarse antes de lanzar los scripts para hacer la instalación en otro directorio más conveniente (disco externo, etc)
- Es posible descargar las imágenes comprimidas manualmente (o intercambiarlas con USB), basta descargar los archivos con extensión `.vdi.zip` de `http://ccia.esei.uvigo.es/docencia/SSI/1819/practicas/` y copiarlos en el directorio anterior (`$DIR_BASE`) para que el script haga el resto.
- Si no lo hacen desde el script anterior, se pueden arrancar las instancias VIRTUALBOX desde el interfaz gráfico de VirtualBOX o desde la línea de comandos con `VBoxManage startvm <nombre MV>_<id>`

2. Imágenes descargadas
   - **base_ssi.vdi** (1,2 GB comprimida, 4,8 GB descomprimida): Imagen genérica (común a todas las MVs) que contiene las herramientas a utilizar
     Contiene un sistema Debian 9 con herramientas gráficas y un entorno gráfico ligero LXDE (*Lighweight X11 Desktop Environment*) [LXDE].
   - **swap1GB.vdi**: Disco de 1 GB formateado como espacio de intercambio (SWAP)
3. Usuarios configurados e inicio en el sistema
   - Usuarios disponibles

login	password
root	purple
usuario	usuario

   - Acceso al entorno gráfico una vez logueado (necesario para poder copiar y pegar desde/hacia el anfitrión)

     ```
     root@datos:~# startx
     ```

   - Habilitar copiar y pegar desde/hacia el anfitrión en el menú `Dispositivos -> Portapapeles compartido -> bidir` de la ventana de la máquina virtual.

2. Intercepción de mensajes y escaneo de puertos

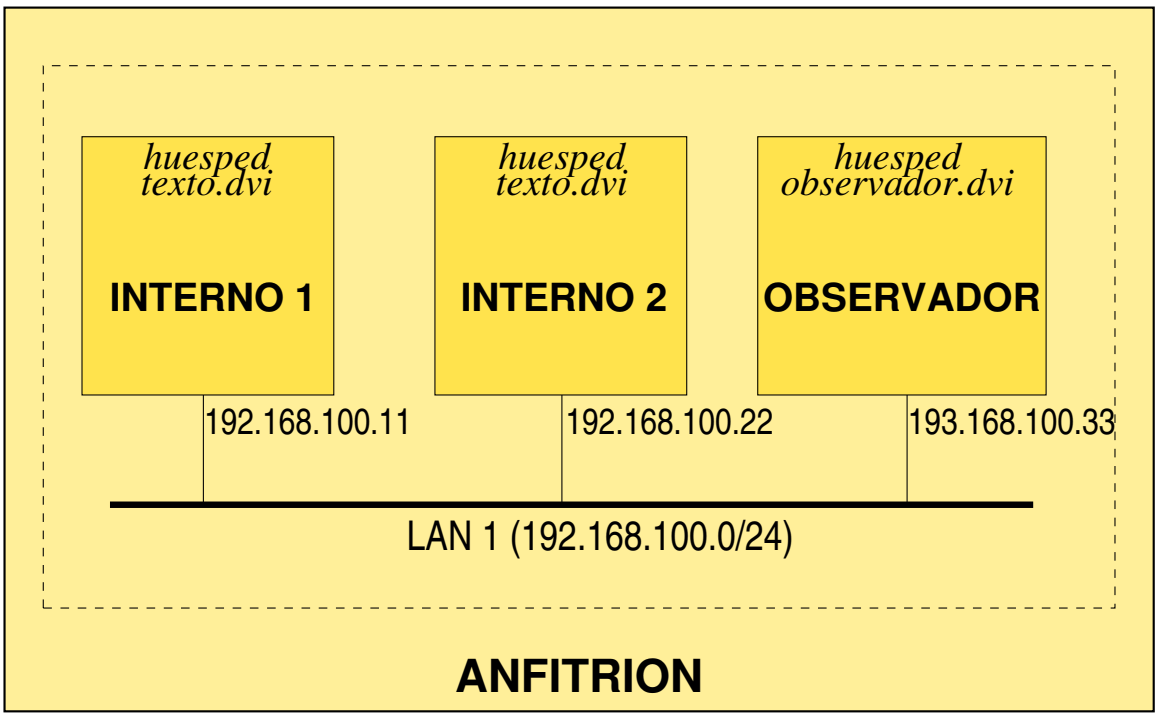
2.1. Descripción

El ejercicio consta de dos partes.

- Realizar una sesión de intercepción de mensajes utilizando el *sniffer*/analizador de redes WIRESHARK y comprobar la vulnerabilidad de los servicios que no usan cifrado.
- Realizar una sesión de recopilación de información empleando el escáner de puertos NMAP.

2.2. Desarrollo

Red donde se realizarán los ejercicios:



Servicios arrancados por defecto en todas las máquinas

- servidor web (Apache 2) [**Nota:** puede ser necesario reiniciarlo manualmente con `service apache2 restart`]
- servidor telnet (arrancado por `openbsd-inetd`)
- servidor SSH (openSSH)
- servidor ftp (arrancado por `openbsd-inetd`)
- servidor finger (arrancado por `openbsd-inetd`)
- servidor MySQL
- servidor SMTP (postfix)
- servidores POP3 e IMAP (dovecot)

2.3. Ejercicio 1

El primer ejercicio consistirá en el uso de la herramienta WIRESHARK desde el equipo **observador** para interceptar el tráfico TELNET, HTTP y SSH entre los equipos **interno1** e **interno2**.

WIRESHARK es un *sniffer* y analizador de protocolos que recopila los paquetes que fluyen por la red, los analiza, extrae el contenido de los campos de diferentes protocolos y los presenta al usuario.

- Página de WIRESHARK: http://www.wireshark.org
- Más información: http://es.wikipedia.org/wiki/Wireshark

2.3.1. Pasos:

1. En **observador (192.168.100.33)**: iniciar WIRESHARK
   - Iniciar el entorno gráfico:

```
# startx
```

- Arrancar WIRESHARK: `[Inicio] >Internet >Wireshark`

2. En **observador (192.168.100.33)**: iniciar la escucha de la red.
   - Menú `''Capture'' ->''Options''`
   - En `''Input''`, seleccionar el interfaz **enp0s3** + botón `[Start]` para iniciar la captura
3. En **interno1 (192.168.100.11)**: iniciar una conexión TELNET con **interno2 (192.168.100.22)**

```
interno1:~# telnet 192.168.100.22
Trying 192.168.100.22...
Connected to interno2.
Escape character is '^]'.

Linux 2.6.26-1-686 (192.168.100.22) (pts/18)

alqueidon login: usuario
Password: usuario
...
interno2:~$ ls -l
...
interno2:~$ exit
```

4. En **observador (192.168.100.33)**: analizar el tráfico recopilado
   - Detener captura con el botón `[Stop]`
   - Recorrer los paquetes capturados y comprobar los datos intercambiados
     - Puede filtrarse el tráfico TELNET capturado (poner `telnet` en el "buscador")
     - Puede verse el tráfico completo de la conexión `telnet` (sobre el primer paquete de la conexión dirigido al puerto 23, `[botón derecho] > Follow TCP stream`)

2.3.2. Tareas:

Tarea 1. Repetir el ejercicio de captura de tráfico, realizando una conexión SSH desde el equipo **interno1 (192.168.100.11)** al equipo **interno2 (192.168.100.22)**.

```
interno1:~# ssh usuario@192.168.100.22
usuario@192.168.100.22's password: usuario
...
interno2:~$ ls -l
...
interno2:~$ exit
```

Tarea 2. Repetir el ejercicio de captura de tráfico, realizando una conexión WEB desde el equipo **interno1 (192.168.100.11)** al equipo **interno2 (192.168.100.22)**.

- Poner de nuevo Wireshark a escuchar en el interfaz **enp0s3**
  (puede ser recomendable salir de Wireshark y volver a iniciarlo)
- **Opción 1:** usar el navegador web en modo texto LYNX o

  ```
  interno1:~# lynx 192.168.100.22
  ...
  ```

- **Opción 1:** usar el navegador MIDORI desde el interfaz gráfico [preferiblemente desde una *Ventana de navegación privada*]

Nota: En caso de no obtener respuesta, reiniciar el servidor `apache2` en la máquina **interno2 (192.168.100.22)**

```
interno2:~# service apache2 restart
```

Tarea 3. Habilitar el soporte SSL en el servidor Apache2 de **interno2 (192.168.100.22)** comprobar que sucede cuando se "escucha" una conexión SSL/TLS.

En el equipo **interno2 (192.168.100.22)**:

1. Crear un certificado autofirmado para el servidor web.

   ```
   interno2:~# mkdir /etc/apache2/ssl/
   interno2:~# make-ssl-cert /usr/share/ssl-cert/ssleay.cnf /etc/apache2/ssl/apache.pem
   ```

   - Cuando se solicite el nombre del servidor HTTP, indicar **interno2.ssi.net**
   - Cuando se solicite los *"nombres alternativos*, dejar el campo en blanco

   El fichero generado (`/etc/apache2/ssl/apache.pem`) contiene tanto el certificado del servidor como la clave privada asociada al mismo.

   ```
   interno2:~# cat /etc/apache2/ssl/apache.pem
   interno2:~# openssl x509 -text -in /etc/apache2/ssl/apache.pem
   interno2:~# openssl rsa -text -in /etc/apache2/ssl/apache.pem
   ```

   Nota: `make-ssl-cert` es una utilidad de Debian (incluida en el paquete DEB *ssl-cert*) para generar certificados autofirmados para pruebas (los datos de configuración del certificado a generar se indican en `/usr/share/ssl-cert/ssleay.cnf`). Internamente hace uso de las utilidades de la librería openSSL.

   Nota: En un servidor real se suele utilizar un certificado emitido por una autoridad de certificación (CA) reconocida (o bien una CA pública o una CA propia de la organización). No es recomendable utilizar certificados autofirmados en sistemas en producción ya que son fácilmente falsificables.

2. Editar la configuración SSL por defecto para indicar el certificado del servidor y su respectiva clave privada.

   ```
   interno2:~# nano /etc/apache2/sites-available/default-ssl.conf
   ```

   Asignar los siguientes valores a los parámetros (en caso de que estén comentados descomentarlos)

   ```
   ...
   SSLEngine on
   ...
   SSLCertificateFile /etc/apache2/ssl/apache.pem
   SSLCertificateKeyFile /etc/apache2/ssl/apache.pem
   ...
   ```

   Asegurar que el fichero `/etc/apache2/ports.conf` incluya el valor `Listen 443`

3. Habilitar soporte SSL en Apache2 y habilitar la configuracion SSL por defecto

   ```
   interno2:~# a2enmod ssl
   interno2:~# a2ensite default-ssl
   interno2:~# service apache2 restart
   ```

   Nota:

   - **a2enmod** es un comando (en Debian y derivados) para habilitar módulos de Apache2
     Los ficheros de configuración de los módulos disponibles están en `/etc/apache2/mods-available/` y al habilitarlos se crea un enlace simbólico desde `/etc/apache2/mods-enabled/`
   - **a2ensite** es un comando (en Debian y derivados) para habilitar configuraciones de "sitios web" en Apache2
     Los ficheros de configuración de los "sitios web" disponibles (normalmente son configuraciones de servidores virtuales Apache) están en `/etc/apache2/sitess-available/` y al habilitarlos se crea un enlace simbólico desde `/etc/apache2/sites-enabled/`

En el equipo **observador (192.168.100.33)**: Iniciar una sesión de escucha en WireShark.

En el equipo **interno1 (192.168.100.11)**:

1. Desde un navegador web Midori [preferiblemente desde una *Ventana de navegación privada*], indicar `https://interno` en la barra de direcciones.
2. Dará un aviso de que la CA que firma el certificado del servidor no está reconocida.
   Añadir la correspondiente excepción de seguridad (botón `[Confiar en este sitio]`) para permitir la descarga y aceptación del certificado

Comprobar en **observador (192.168.100.33)** el resultado de la escucha (aparecerán varios intentos de conexión, el fallido y el exitoso tras aceptar el certificado)

2.4. Ejercicio 2

El segundo ejercicio consistirá en el uso de la herramienta de escaneo de puertos NMAP para obtener información de los equipos y servicios de la red.

NMAP implementa diversas técnicas para extraer información de los equipos que forman parte de una red y para identificar los puertos y servicios que están disponibles en distintas máquinas. Algunos de los métodos disponibles realizan el escaneo sin dejar rastro, mientras que otros dejarán un rastro en los ficheros de log de las máquinas analizadas.

- Página de NMAP: `http://www.nmap.org`
- Más información: `http://es.wikipedia.org/wiki/Nmap`
- Manual en español: `http://nmap.org/man/es/`
- Tutorial en inglés: `http://www.nmap-tutorial.com`

2.4.1. Pasos:

1. Enumerar equipos de la red y sus servicios

   Desde la máquina **observador (192.168.100.33)**:

   a) Lanzar un escaneado Ping Sweeping [opción **-sP**] para identificar, mediante Ping, las máquinas que componen la red

   ```
   observador:~# nmap -sP 192.168.100.0/24
   ```

   b) Sobre cada uno de los equipos que aparezcan como activos (exluido **observador**) realizar un escaneo de tipo TCP connect scanning [opción **-sT**] para determinar que puertos están abiertos.

   ```
   observador:~# nmap -sT -v 192.168.100.11
   observador:~# nmap -sT -v 192.168.100.22
   ```

   c) Repetir el escaneado sobre INTERNO1(192.168.100.11), añadiendo la opción **-O** para que NMAP trate de identificar el Sistema Operativo que ejecuta y la opción **-sV** para determinar la versión concreta de los servicios que tiene activados.

   ```
   observador:~# nmap -sT -O -sV 192.168.100.11   (tarda unos segundos)
   ```

   Los escaneados anteriores dejan rastro. Comprobar los ficheros de log `''tail /var/log/syslog''` en las máquinas **interno1** e **interno2** y verificar que ha quedado constancia de las conexiones realizadas por NMAP.

   ```
   interno1:~# tail -100 /var/log/syslog | less
   ```

   Nota: El rastro del escaneo de tipo `-sT` que queda en `/var/log/syslog`

   - Fue guardado por el servidor TELNET en el momento en que se estableció la conexión Telnet

- Es necesario haber arrancado previamente el servidor Telnet (`/etc/init.d/openbsd-inetd start`).

2. Comprobar escaneos "silenciosos"

   Evaluaremos el comportamiento de los distintos tipos de escaneo sobre la máquina **interno1**(192.168.100.11)

   a) En la máquina INTERNO1(192.168.100.11) se habilitará una regla del firewall *netfilter* para hacer log de los paquetes SYN con intentos de conexión TCP.

   - Escribir el siguiente comando **iptables**

     ```
     interno1:~# iptables -A INPUT -i enp0s3 -p tcp \
                          --tcp-flags SYN SYN -m state --state NEW \
                          -j LOG --log-prefix "Inicio conex:"
     ```

   - Monitorizar continuamente el fichero de logs `/var/log/syslog`, con el comando `tail -f`

     ```
     interno1:~# tail -f /var/log/syslog
     ```

     (el terminal se libera con CONTROL+C)

   b) Desde la máquina **observador(192.168.100.33)** lanzar 3 tipos de escaneos nmap y comprobar en INTERNO1(192.168.100.11) como evoluciona el log.

   TCP connect scanning [opción `-sT`] Escaneo con conexiones TCP completas (opción por defecto)

   ```
   observador:~# nmap -sT 192.168.100.11
   ```

   SYN scanning [opción `-sS`] Escaneo con paquetes SYN (conexiones parcialmente iniciadas)

   ```
   observador:~# nmap -sS 192.168.100.11
   ```

   NULL scanning [opción `-sN`] Escaneo con paquetes "nulos" (todos los flags TCP a 0)

   ```
   observador:~# nmap -sN 192.168.100.11
   ```

Nota: Existe un interfaz gráfico para NMAP que se puede arrancar desde el entorno gráfico de **observador(192.168.100.33)** para probar otras opciones del escaner.

- Desde el menú principal: `[Inicio] > Internet > Zenmap`
- Desde un terminal:

  ```
  observador:~# zenmap &
  ```

2.5. Documentación y entrega

El material entregable de esta práctica constará de una pequeña memoria documentando los ejercicios realizados y los resultados obtenidos en cada paso realizado, junto con las conclusiones que se deriven de dichos resultados.

- Descripción de acciones realizadas y resultados obtenidos en ejercicio 1
- Resultados de las tareas 1, 2 y 3
- Conclusiones del ejercicio 1 (explicar la diferencia en el uso de `telnet` frente a `ssh` y de `http` respecto `https`)
- Descripción de acciones realizadas y resultados obtenidos en ejercicio 2
- Conclusiones del ejercicio 2 (explicar la diferencia entre los tres tipos de escaneo realizados [TCP Scan, SYN Scan, NULL Scan])

ENTREGA: en FAITIC, hasta **9/12/2018**