Seguridad en Redes de Ordenadores
Práctica 4: OpenVPN en mininet

GSyC
Departamento de Teoría de la Señal y Comunicaciones y
Sistemas Telemáticos y Computación

Marzo de 2018

Resumen

El objetivo de esta práctica es configurar VPNs utilizando la herramienta OpenVPN. Esta práctica se va a realizar en el entorno de emulación de redes mininet. Para ello usaremos una máquina virtual que contiene todos los paquetes necesarios para realizar la práctica. Primeramente, debes importar en VirtualBox el siguiente fichero que contiene una máquina virtual Linux 16.04: `/var/lib/vms/mininet-sro/mininet-sro.ova`. Asegúrate antes de importar la máquina, que en las opciones de virtualbox tienes algo como:

`General -> Avanzado -> Carpeta instantáneas = /var/tmp/miusuario`

De esta forma la máquina virtual se quedará almacenada en una carpeta local en la máquina del laboratorio y no en tu cuenta personal. Es muy importante esta configuración ya que tu cuenta personal es una carpeta en la red y si no almacenas la máquina virtual en una carpeta en local, virtualBox no funcionará bien. Por otro lado, siempre que necesites acceder a los ficheros de esta práctica deberás sentarte en el mismo ordenador donde realizaste la importación, ya que estos ficheros se quedarán localmente almacenados en esa carpeta.

El usuario de la maquina virtual es `sro` contraseña `SRO-URJC`.

1. Configuración IP del escenario

En la figura 1 se muestra una empresa que tiene 2 sucursales que se desean comunicar a través de una red insegura.

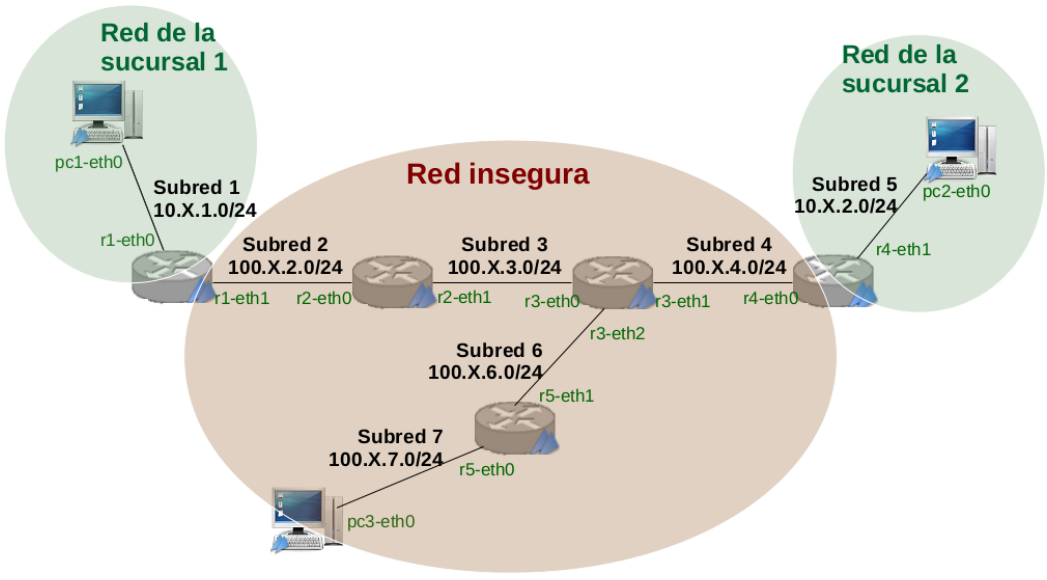

Figura 1: Escenario de red para la práctica openVPN

Descarga de aulavirtual el fichero `lab-openvpn.tgz` dentro de la máquina de virtualBox y descomprímelo.

Las direcciones asignadas son las siguientes. Cada alumno deberá sustituir el valor de X por el número que se le haya asignado:

Subred 1 (privada)	`10.X.1.0/24`	pc1-eth0=10.X.1.10 y r1-eth0=10.X.1.1
Subred 2 (pública)	`100.X.2.0/24`	r1-eth1=100.X.2.1 y r2-eth0=100.X.2.2
Subred 3 (pública)	`100.X.3.0/24`	r2-eth1=100.X.3.2 y r3-eth0=100.X.3.3
Subred 4 (pública)	`100.X.4.0/24`	r3-eth1=100.X.4.3 y r4-eth0=100.X.4.4
Subred 5 (privada)	`10.X.2.0/24`	r4-eth1=10.X.2.4 y pc2-eth0=10.X.2.20
Subred 6 (pública)	`100.X.6.0/24`	r3-eth2=100.X.6.3 y r5-eth0=100.X.6.5
Subred 7 (pública)	`100.X.7.0/24`	r5-eth1=100.X.7.5 y pc3-eth0=100.X.7.30

Edita el fichero `vpn.py` para sustuir el valor de "X"por el que corresponde a cada alumno, ten en cuenta que debes sustituirlo en todo el fichero, en la configuración de los pcs, los routers y en la configuración de las rutas.

OpenVPN necesita que los nombres de las máquinas para las que vamos a crear certificados (extremos de túnel OpenVPN) estén dados de alta en el servicio de resolución de nombres. Para ello, edita el fichero `lab-openvpn/<máq>/etc/hosts` (siendo `<máq>`, la carpeta de `r1`, `r4` y `pc3`) para añadir las siguientes líneas (sustituyendo previamente el valor de X por el que se le ha asignado a cada alumno). Recuerda que para editar este fichero lo deberás realizar con `sudo`:

```
100.X.4.4       r4
100.X.2.1       r1
100.X.7.30      pc3
```

Están configuradas las rutas para que:

- Todos los routers tengan conectividad con todas las subredes de la red insegura.
- Los pcs tengan una ruta por defecto al router al que están directamente conectados.

2. Configuración openVPN

Supón que los trabajadores de la Sucursal1 quieren comunicarse con los trabajadores de la Sucursal2 a través de una red insegura. Para ello, se va a establecer la configuración de openVPN en el que se va instalar un servidor openVPN en `r4` y un cliente openVPN en `r1`.

Adicionalmente hay un trabajador que se encuentra trabajando fuera de las oficinas de la empresa en `pc3` y que desea comunicarse con los ordenadores de la Sucursal2. Por ello, `pc3` también tendrá configurado un cliente openVPN que se comunique con el servidor de `r4`.

2.1. Generación de certificados y parámetros Diffie-Hellman

La autenticación de los extremos del túnel se va a realizar con certificados X.509. Para ello, la empresa decide crear los certificados necesarios para todos los extremos del túnel.

Todos los certificados los vamos a generar en la máquina virtual `mininet`. Para ello vamos a utilizar las herramientas `easy-rsa`. Descárgate del `aulavirtual` el fichero `easy-rsa.tgz` [1] y descomprímelo dentro de la máquina `mininet`.

En un entorno real, los certificados se crearían en cada máquina, independientemente. Es decir, en cada máquina que interviene en la comunicación a través de OpenVPN se generarían las claves pública y privada: en la máquina de la autoridad de certificación, en `r4` (servidor), en `r1` (cliente) y en `pc3` (cliente). Y después, la máquina que ejerce de autoridad de certificación generaría los certificados, firmándolos, tanto para el servidor como para los clientes de OpenVPN: `r4`, `r1` y `pc3`. Como nosotros vamos a crear todos los certificados en la misma máquina (en la máquina `mininet` de virtualBox), para simular que estamos en máquinas diferentes vamos a necesitar crear una copia de esta carpeta (`easy-rsa`) por cada máquina. Por ejemplo copiaremos la carpeta `easy-rsa.tgz` en diferentes carpetas, una por cada máquina:

- `easy-rsa-CA`: copia de la carpeta `easy-rsa` para la autoridad de certificación.
- `easy-rsa-r4`: copia de la carpeta `easy-rsa` para el servidor `r4`.
- `easy-rsa-r1`: copia de la carpeta `easy-rsa` para el cliente `r1`.
- `easy-rsa-pc3`: copia de la carpeta `easy-rsa` para el cliente `pc3`.

[1] Se encuentra público en: https://github.com/OpenVPN/easy-rsa

2.1.1. Creación de las claves para la CA

Dentro de la carpeta `easy-rsa-CA` generaremos las claves y el certificado autofirmado de la autoridad de certificación (CA). Observa cómo se han creado en la carpeta `pki` el certificado de la autoridad de certificación (`ca.crt`, su certificado autofirmado) y en la carpeta `pki/private` la clave privada de la autoridad de certificación (`ca.key`).

1. Incluye el contenido del certificado de la CA en formato legible en la memoria.
2. ¿Cómo sabes que es un certificado autofirmado?
3. ¿El certificado lleva algún campo que indique que pertenece a una CA?

2.1.2. Creación de las claves y el certificado firmado para el servidor r4

Dentro de la carpeta `easy-rsa-r4`, vamos a generar una pareja de claves para la entidad `r4` y una solucitud de certificado para que después lo firme la autoridad de certificación. En la carpeta `pki/req` estará la petición de un certificado (`r4.req`, contiene principalmente el nombre de la entidad y su clave pública) y en la carpeta `pki/private` la clave privada de r4 (`r4.key`).

1. Incluye en la memoria el contenido de la solicitud del certificado en formato legible.
2. Genera los parámetros Diffie-Hellman que después se utilizarán para configurar openVPN. Copia el resultado de visualizar el fichero en modo legible.
3. Importa la solicitud de certificado desde la carpeta de la autoridad de certificación y comprueba que se ha importado correctamente, visualizando la solicitud que acabas de importar. Copia el resultado en la memoria (el campo `Attributes: a0:00` significa que no hay atributos).
4. Firma el certificado del servidor con la CA e incluye en la memoria el certificado del servidor firmado en modo legible.
5. Fíjate en el certificado, ¿se puede saber que no pertenece a una autoridad de certificación?

2.1.3. Creación de las claves y certificado firmado para el cliente pc3

Dentro de la carpeta `easy-rsa-pc3`, vamos a generar una pareja de claves para la entidad `pc3` y una solucitud de certificado para que después lo firme la autoridad de certificación. En la carpeta `pki/req` estará la petición de un certificado (`pc3.req`, contiene principalmente el nombre de la entidad y su clave pública) y en la carpeta `pki/private` la clave privada de pc3 (`pc3.key`).

1. Incluye en la memoria el contenido de la solicitud del certificado en formato legible.
2. Importa la solicitud de certificado desde la carpeta de la autoridad de certificación y comprueba que se ha importado correctamente, visualizando la solicitud que acabas de importar. Copia el resultado en la memoria.
3. Firma el certificado del cliente con la CA e incluye en la memoria el certificado del cliente firmado en modo legible.

2.1.4. Creación de las claves y certificado firmado para el cliente r1

Dentro de la carpeta `easy-rsa-r1`, vamos a generar una pareja de claves para la entidad `r1` y una solucitud de certificado para que después lo firme la autoridad de certificación. En la carpeta `pki/req` estará la petición de un certificado (`r1.req`, contiene principalmente el nombre de la entidad y su clave pública) y en la carpeta `pki/private` la clave privada de r1 (`r1.key`).

1. Incluye en la memoria el contenido de la solicitud del certificado en formato legible.
2. Importa la solicitud de certificado desde la carpeta de la autoridad de certificación y comprueba que se ha importado correctamente, visualizando la solicitud que acabas de importar. Copia el resultado en la memoria.
3. Firma el certificado del cliente con la CA e incluye en la memoria el certificado del cliente firmado en modo legible.

2.2. Almacenar los ficheros adecuadamente para cada máquina

Copia sólo los ficheros de claves y certificados necesarios dentro de la carpeta del escenario:
`lab-openvpn/<nombreMáq>/etc/openvpn`
Indica en la memoria qué ficheros has almacenado en la carpeta del escenario y en qué carpetas los has almacenado.

2.3. Configuración del extremo servidor r4

Configura el fichero `server.conf` dentro de `lab-openvpn/r4/etc/openvpn` de `r4` para crear una configuración UDP en el puerto 1194. El servidor openVPN deberá asignar a las máquinas que se comunican a través de la VPN el siguiente rango de subred: `10.X.8.0/24`

Incluye en la memoria las líneas que no estén comentadas del fichero `server.conf`.

2.4. Configuración del extremo cliente r1

Configura el fichero `client.conf` dentro de `lab-openvpn/r1/etc/openvpn` de `r1` para que se conecte al servidor de `r4`.

Incluye en la memoria las líneas que no estén comentadas del fichero `client.conf` de `r1`.

2.5. Configuración del extremo cliente pc3

Configura el fichero `client.conf` dentro de `lab-openvpn/pc3/etc/openvpn` de `pc3` para que se conecte al servidor de `r4`.

Incluye en la memoria las líneas que no estén comentadas del fichero `client.conf` de `pc3`.

2.6. Túnel entre pc3 y r4

En este apartado se va a poner a prueba el túnel OpenVPN entre `pc3` y `r4`.

Desde la carpeta donde has descomprimido el escenario de mininet (`lab-openvpn`), arranca el escenario en mininet:

```
sudo ./vpn.py
```

Comprueba que todas las interfaces de las máquinas tienen configuradas sus direcciones IP y rutas correctamente, de acuerdo a los cambios que realizaste en `vpn.py`.

- Desde la interfaz CLI lanza un xterm en r4 y arranca el servidor OpenVPN (recuerda que deberás introducir la clave que pusiste para cifrar la clave privada de `r4`). Lanza un segundo xterm en `r4` e inicia una captura de tráfico en `r4-eth0` de forma que se capture todo el tráfico en el fichero `openvpn-01.cap`[2].
- Arranca el cliente de openvpn en `pc3`.

Responde a las siguientes cuestiones en la memoria:

1. Copia en la memoria la configuración de la interfaz `tun0` que se ha creado en el lado servidor y explica las direcciones IP que se muestran ¿para qué sirven?
2. Mira la tabla de encaminamiento de `r4` y copia las entradas que hacen referencia a la interfaz `tun0`. Explica su significado.
3. Copia en la memoria la configuración de la interfaz `tun` que se ha creado en el lado cliente y explica las direcciones IP que se muestran ¿para qué sirven?
4. Mira la tabla de encaminamiento de `pc3` y copia las entradas que hacen referencia a la interfaz `tun0`. Explica su significado.
5. Realiza un `ping` desde `pc3` hacia `r4` enviando 3 paquetes ICMP Echo Request sin usar el túnel. ¿Qué dirección IP destino has especificado para enviar los paquetes sin utilizar el túnel? ¿Por qué?
6. Realiza un `ping` desde `pc3` hacia `r4` enviando 3 paquetes ICMP Echo Request usando el túnel. ¿Qué dirección IP destino has especificado para enviar los paquetes utilizando el túnel? ¿Por qué?
7. Interrumpe la captura que has realizado (Ctrl+C). Abre la captura `openvpn-01.cap` en Wireshark y explica el contenido:
   a) Indica el identificador de sesión local y remoto que se establece en los mensajes `P_CONTROL_HARD_RESET` y fíjate como aparece en todos los mensajes posteriores.
   b) Observa el campo `Message Packet-ID` indica cuál es su valor incial para el cliente y para el servidor. Explica cómo va cambiando en cada uno de los paquetes `P_CONTROL` que se envían.
   c) ¿Por qué los mensajes `P_ACK` no llevan el campo `Message Packet-ID`?
   d) ¿En qué paquete se asiente el mensaje `P_CONTROL_HARD_RESET_CLIENT_V2`? ¿Cómo lo sabes?
   e) ¿En qué paquete se asiente el mensaje `P_CONTROL_HARD_RESET_SERVER_V2`? ¿Cómo lo sabes?
   f) El primer mensaje que envía el cliente al servidor del SSL/TLS handshake es `Client Hello`. ¿Alguno de los campos viaja cifrado?

[2] r4: tcpdump -i r4-eth0 -s 0 -w openvpn-01.cap

g) Localiza el primer mensaje que envía el servidor al cliente para establecer SSL/TLS handshake, es `Server Hello`. ¿Alguno de los campos viaja cifrado? Indica el algoritmo de cifrado que se va a usar.

h) A continuación verás en la captura que Wireshark no reconoce como mensajes de tipo openVPN: un mensaje del servidor y 2 del cliente. Estos mensajes son el intercambio de certificados y parámetros Diffie Hellman. La negociación TLS termina con el mensaje del servidor `Change Cipher Spec` y 4 mensajes `Application Data`, 2 del cliente y 2 del servidor, que van cifrados.

i) Observa el contenido de los mensajes `P_DATA`. ¿Por qué no llevan el campo `Message Packet-ID`?

j) ¿Qué crees que hay dentro de los mensajes `P_DATA`?

8. Comprueba el contenido del fichero `ipp.txt` en el servidor y explica su contenido.

2.7. Conectividad entre pc3 y pc2

Responde razonadamente a las siguientes cuestiones en la memoria:

1. ¿Qué crees que ocurrirá si desde `pc3` se envía un `ping` a la dirección IP de `pc2`?
2. Modifica la configuración de `r4` para que anuncie sus subredes internas y desde `pc3` funcione un `ping` a la dirección IP de `pc2` a través del túnel OpenVPN. Explica en la memoria qué has modificado.
3. Realiza una captura en `r4(eth0)` (fichero `openvpn-02.cap`) y en `pc2` (fichero `openvpn-03.cap`) mientras ejecutas el `ping` y explica las capturas.
4. Explica la tabla de encaminamiento que tiene `pc3` y las diferencias con la tabla de encaminamiento que viste en el apartado anterior.
5. ¿Crees que `pc2` sabe que se está usando un túnel openVPN? ¿Por qué?

2.8. Túnel entre r1 y r4

En este apartado se va a poner a prueba el túnel OpenVPN entre `r1` y `r4`. Mantén la configuración realizada anteriormente y ahora:

- Inicia una captura de tráfico en `r4(eth0)` de forma que se capture todo el tráfico en el fichero `openvpn-04.cap`.
- Arranca el cliente en `r1`.
- Realiza un `ping` desde `r1` hacia `r4` enviando 3 paquetes ICMP Echo Request sin usar el túnel.
- Realiza un segundo `ping` desde `r1` hacia `r4` enviando 3 paquetes ICMP Echo Request usando el túnel.
- Interrumpe la captura que has realizado (Ctrl+C)

Responde razonadamente a las siguientes cuestiones en la memoria:

1. Comprueba el contenido del fichero `ipp.txt` en el servidor y explícalo.
2. ¿Qué crees que ocurrirá si desde `r1` se envía un `ping` a `pc2`? Incluye la tabla de encaminamiento de `r1`.
3. ¿Qué crees que ocurrirá si desde `pc1` se envía un `ping` a `pc2`? ¿Por qué?
4. Modifica la configuración en `r4` para permitir que `r4` alcance las direcciones IP de la Subred1. Indica las modificaciones en la memoria. Comprueba que ahora funciona.

2.9. Conectividad entre entre pc3 y pc1

Responde razonadamente a las siguientes cuestiones en la memoria:

1. ¿Qué crees que ocurrirá si desde `pc3` se envía un `ping` a `r1`? ¿Por qué?
2. ¿Qué crees que ocurrirá si desde `pc3` se envía un `ping` a `pc1`? ¿Por qué?
3. Modifica la configuración en `r4` para permitir que `r4` anuncie las subredes internas del cliente `r1` (Subred1) al cliente `pc3`. No olvides guardar el fichero que has modificado en la máquina virtual.
4. Realiza una captura de tráfico (`openvpn-05.cap`) en `r4(eth0)` que muestre que funciona un `ping` desde `pc3` a `pc1`. Explica los paquetes capturados.
5. Explica la tabla de encaminamiento que tienen `pc3` y `r1` y las diferencias con la tablas de encaminamiento que tenían previamente.

3. Normas de entrega

Deberás subir al `aulavirtual` un fichero `openvpn.tgz` que contenga los siguientes archivos:

- La memoria en formato pdf.
- Un archivo `openvpn-caps.tgz` que contenga los ficheros con las capturas de `openvpn-01.cap` a `openvpn-05.cap`