SPRING 18

UNIDAD 1

EJERCICIOS 1, 2 Y 3

TÉCNICAS DE TUNNELING-SSH

-- EJERCICIO_1.- Realizar un túnel dinámico, a través de SSH Tuneling, para descubrir la IP de la máquina Metasploitable2.

- INTRODUCCIÓN.- Un **túnel dinámico** se establece con la finalidad de descubrir hosts, servicios y puertos que están mas allá de alcance de nuestra kali con la máquina comprometida, con la intención de ampliar nuestra explotación realizando movimientos laterales.
- Esta técnica esta dentro del **método tuneling-SSH**, por lo que debemos contar con esta conexión. En un caso real que no la tuvieramos, tendriamos que explotar alguna vulnerabilidad o usar *"fuerza bruta*" u otras gestiones que lleven a conocer credenciales válidas del SSH de la máquina comprometida, que en este caso, la usaremos como *"pivoting"* para después hacer el movimiento lateral.
- Para este ejercicio, se ha establecido un **host-anfitrion de kali linux** con IP 192.168.1.134, habiendo configurado en modo 'post -up', un enrutamiento de red en el rango 10.0.0.0/24, a través de la 1º interfaz con rango de IP - 192.168.1.0/24 de "pivotingVM".

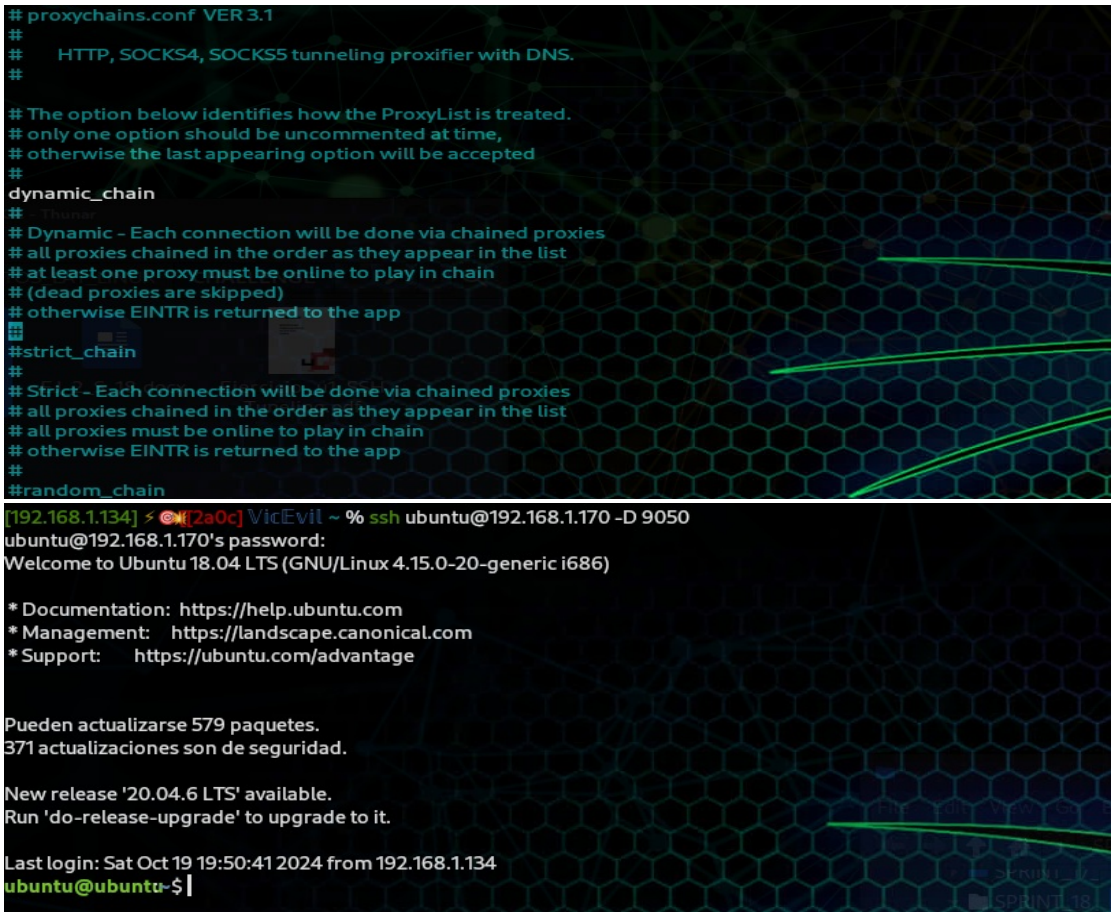
La máquina "pivotingVM" ha sido configurada en VirtualBox, con 2 interfaces de red: una en modo bridge por la red eth0 y en dhcp, la cual está en conexión directa con la kali, y otra interfaz de red conectada en modo red privada, la cual, es la que conectará con la "Metasploitable2" en un rango de IP - 10.0.0.0/24. En último lugar y no menos importante, se ha configurado la máquina "Metasploitable2", conectada con una sola interfaz de red en modo privado, similar a la 2 interfaz de la "pivotingVM".

Una vez establecido el ámbito de este ejercicio, se ha establecido de **manera práctica** de la siguiente forma:

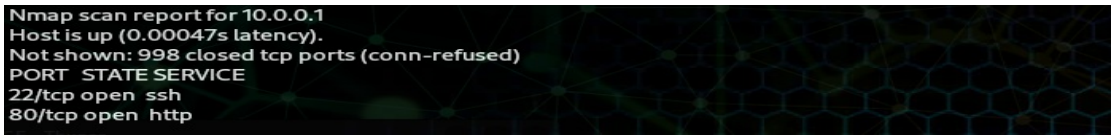
1) En nuestra máquina Kali, ejecutaremos el comando ***'IP a'***, con el que sabremos la IP de nuestra Kali. Despues ejecutaremos un "***sudo arp-scan --localnet***" para conocer todas las IPs conectadas, y, en mi caso, al ser la máquina kali host (anfitrión principal), consulto paralelamente en VirtualBox, dentro del apartado red de cada maquina **la MAC asignada** a cada una de ellas, para asegurarme que la IP elegida es la correcta.
2) Abrimos una terminal en nuestra kali, para conectarnos mediante **SSH a la maquina pivoting** (Interfaz 1: 192.168.1.170), de la cual conocemos las credenciales *"ubuntu:ubuntu"* , incluyendo al final el flag -D(dinámico) y el puerto por defecto de la **herramienta "*proxychains***", siendo ésta última, una herramienta que permite redirigir el tráfico de una aplicación, a través de uno o más proxies, como SOCKS5, HTTP, TCP o TOR, para enmascarar tu dirección IP y ocultar tu tráfico.

Proxychains va canalizando todo el tráfico de red y las peticiones que se realicen por el puerto establecido (se puede cambiar), siendo la información devuelta por ese mismo puerto, provocando "*cuello de botella*" que ralentizan el tráfico de la red. Se pueden usar en 3 modos diferentes::

- *dynamic_chain.*- intenta usar los proxies en orden, saltando los que fallen (no se corta la conexión).
- *strict_chain.*- Usa los proxies en el orden establecido, cayendo la conexión si uno falla.
- *random_chain.*- Se conecta a proxies al azar.



3) Una vez establecido el túnel dinámico, ahora abrimos otra nueva terminal en nuestra kali y desde allí, ejecutaremos el comando "**nmap"** el cual sera lanzada con el flag **"-sT"**, siendo este escaneo mas compatible con el tráfico enrutado a traves de proxies, al no usar paquetes "SYN" (como hace el flag -sS). Todo esto, enviado a través de "proxychains" en modo "**dynamic_chain**", en búsqueda de nuevos hosts en el rango de IPs seleccionados, en nuestro caso: 10.0.0.0/24.



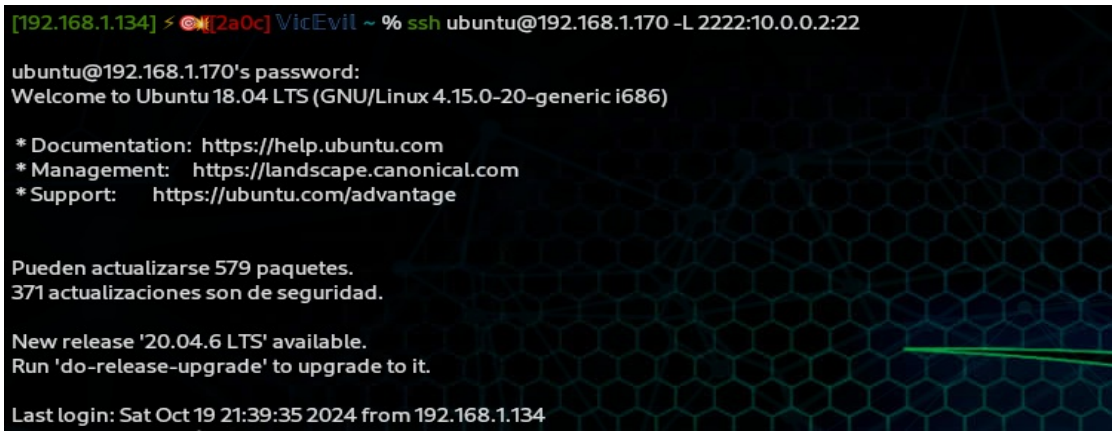
Como resultado ha encontrado una nueva IP, que es la **10.0.0.2**, la cual, muestra todos los puertos y servicios asociados a los mismos de nuestra maquina **“Metaesploitable2**”, junto con la 2ºinterfaz de la maquina pivoting con IP 10.0.0.1.

EJERCICIO_2

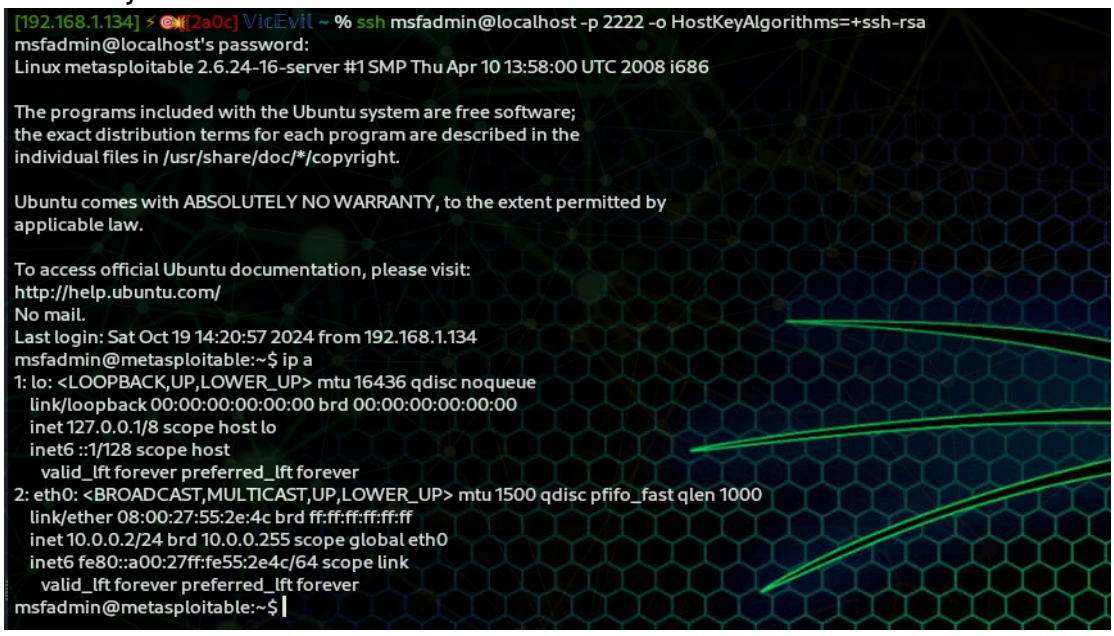
2.1- Realizar un túnel local, a través de SSH Tuneling, para conectarnos directamente a la máquina Metasploitable2 desde la Kali.

- INTRODUCCIÓN.- Un túnel “Local” es utilizado para redirigir el tráfico desde un puerto de tu máquina local hacia un destino fijo a través de un servidor remoto seguro, encapsulando el tráfico por un túnel cifrado (SSH), siendo utilizado para hacer un movimiento lateral en una explotación real, una vez descubierto el host mediante el túnel dinámico. La infraestructura usada es la misma que la explicada anteriormente, por lo que procederemos a su realización práctica:

I. Se procede a elegir el puerto que vamos a usar en nuestra máquina Kali (2222) y el puerto donde vamos a contectarnos en destino a la metaexploitable2 (22).
II. Una vez hecho esto, se realiza conexión a través de SSH desde nuestra máquina Kali, pivotando o utilizando la conexión de la máquina “pivoting”, la cual tiene la capacidad de comunicarse tanto con Kali como con meta2, para llegar finalmente a su destino final:



III. Con el túnel local establecido, ahora podemos abrir una nueva conexión a través de SSH, directamente desde Kali a meta 2, únicamente cambiando el puerto por defecto 22, por el 2222 en su ejecución:

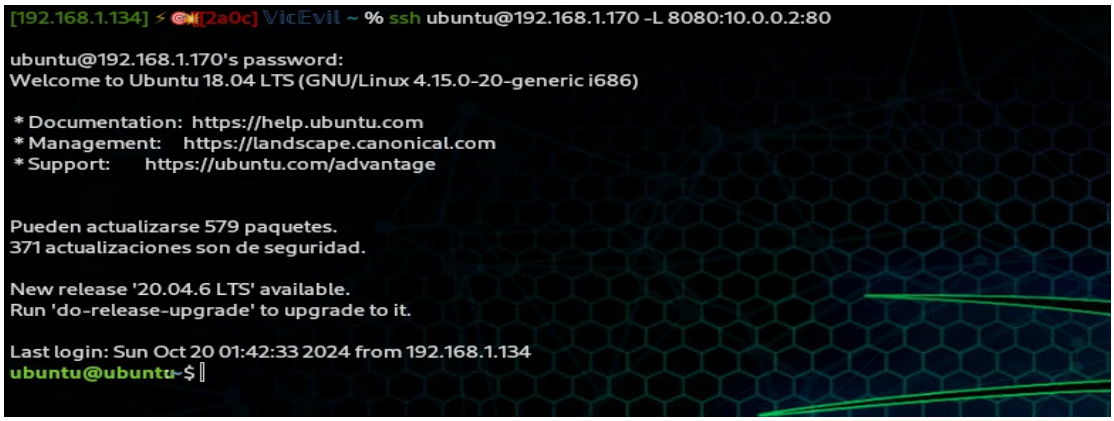


En este caso, se ha usado el flag *“-o HostKeyAlgorithms=+ssh-rsa”*, al realizar el comando SSH, para especificar qué algoritmos de clave de host, debe aceptar el cliente SSH, cuando se conecta a un servidor, siendo importante cuando se trata de compatibilidad con ciertos tipos de claves de hosts que pueden estar obsoletos, como el caso de la “metaexploitable2”.

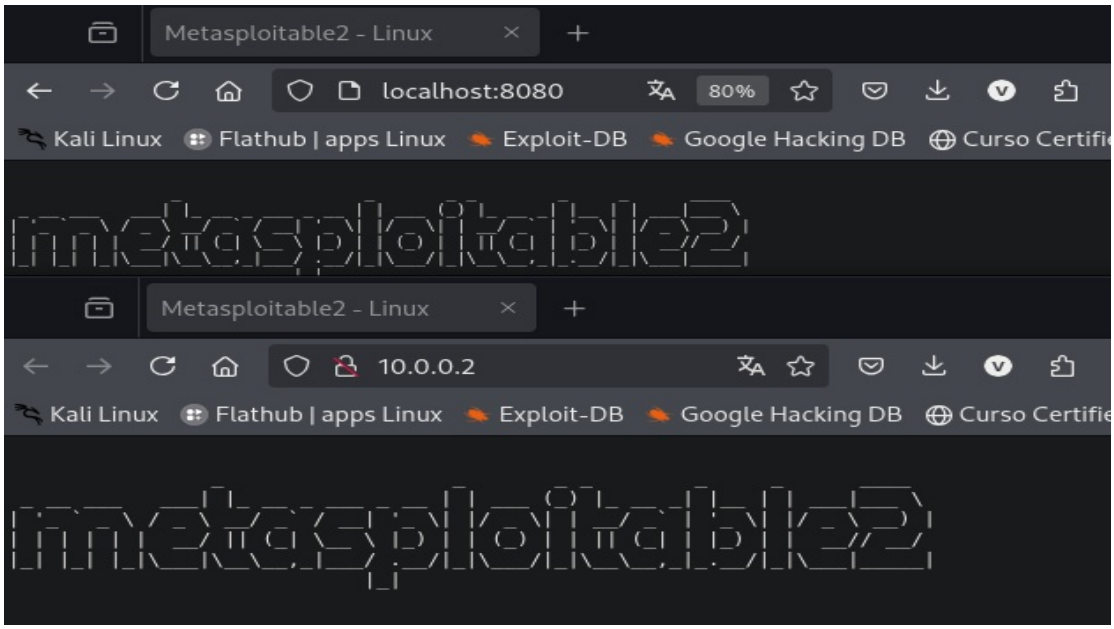
2.2- Realizar un túnel local, a través de SSH Tuneling, para acceder desde la Kali a la aplicación web de la máquina Metasploitable2.

- Este caso es muy similar al anterior, solamente cambiando los puertos y los servicios a los que podremos acceder.

❖ Se establece el túnel local, utilizando como intermediario la máquina pivoting, la cual comunica el host Kali y la máquina meta 2, especificando el puerto 8080 en la Kali y el puerto 80 para la meta 2:



❖ Este comando crea un túnel local, que redirige todo el tráfico que llega al puerto 8080 de tu máquina local, hacia la dirección 10.0.0.2 en el puerto 80 (servidor web), a través de la máquina pivoting, por lo que podemos acceder al contenido del servidor web directamente a través de nuestro localhost:8080, recibiendo la misma información que si ejecutas 10.0.0.2:80 en la máquina meta 2:



-- EJERCICIO_3.- Realizar un túnel reverso para brindar un archivo llamado *"test.txt"*, desde la Kali a Metasploitable2.

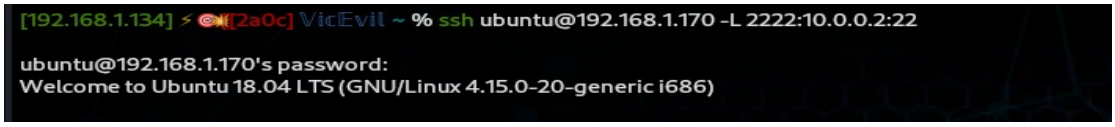
- INTRODUCCIÓN.- Un **túnel reverso o remoto**, es técnica de red dentro del tuneling -SSH, en la que el tráfico de un servidor remoto se redirige hacia un puerto del host local u a otro destino que especifiques, mediante una conexión SSH, y a través de la máquina que mantiene la comunicación entre servidor remoto y host local (pivoting).

-Este recurso puede ser muy útil en explotaciones de pentesting, donde una vez has comprometido una máquina de una red organizacional, comienzas a realizar movimientos laterales en ella, necesitando en algun momento recursos que no se encuentran en la máquina atacada, abriendo la posibilidad de acceder a servicios locales desde esa máquina.

- En nuestro caso, es el **tráfico inverso** al que se produce con un **túnel local**, es decir, en este caso la metaexploitable2 tendrá acceso a un servicio o recursos de la máquina host Kali, usando para ello el método "*servidor python -wget",* teniendo la salvedad que hay que **utilizar la IP real asignada al host local, no el localhost** como usamos en un puerto local, ya que no se podría conectar por estar el puerto cerrado.

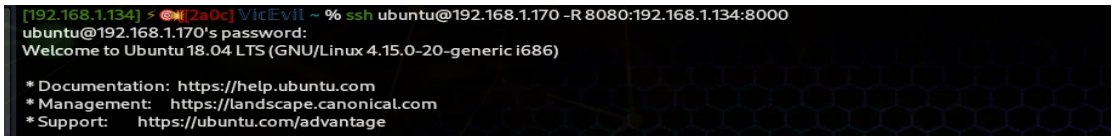
- Se procede con la parte práctica:

- En primer lugar, vamos a realizar un archivo de texto a modo de pruebas ubicado en /documents/mov-laterales_S18/ llamado tunelR.txt conteniendo en su interior el siguiente texto: "*prueba túnel remoto parea que meta 2(atacada por mov lateral) use un servicio de la kali".*
- Ahora elegimos los puertos a utilizar, siendo el 8000 en la Kali y el 8080 en la máquina pivoting.
- Aperturamos un túnel local desde kali hacia meta2, como antes hemos establecido, previamente a establecer el túnel dinámico:

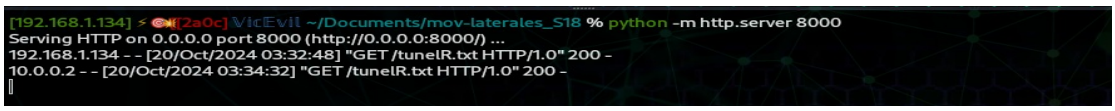


- Aprovechando el túnel local abierto, desde el host Kali, se procede a la conexión por SSH directamente a la máquina meta 2:

- Se realiza el túnel reverso o remoto a través de SSH, usando como siempre, la máquina pivoting como intermediaria, junto al flag -R. En este punto, se agregaran las IPs y puertos, en orden inverso a un túnel local, primero iría la IP de meta 2. En este sentido, se significa que se ha omitido la Ip de meta 2 en el comando, ya que no participa directamente en el túnel y ademas el puerto 8080 se encuentra en la máquina pivoting:



- Se apertura en la host Kali, en el directorio donde se encuentra el archivo a trasladar, un servidor python por el puerto 8000:

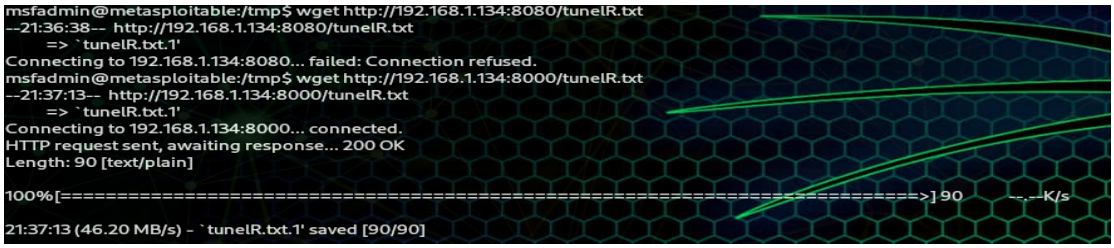


- Finlamente, desde la conexión SSH establecida desde la Kali a meta 2 a traves del túnel local, se procede a usar el método wget, habiéndose usado de dos formas diferentes con el mismo resultado:

- Primera:

Aquí, hemos usado la máquina intermediaria pivoting para establecer la conexión entre meta 2 y Kali host, ya que el puerto remoto se establece a través de pivoting, la cual, mantiene entrelazadas ambas máquinas, habiendo sido positiva la transacción.

- Segunda:



En este caso, hemos usado directamente la IP del host Kali junto a su puerto de conexión del servidor python (8000), resultando también, la transacción correcta, ya que nos conectamos directamente al servidor de python donde se recibe el tráfico desde el puerto 8080 de la máquina pivoting.