Seguridad en Sistemas de Información: Tarea 3

Jorge E. Chávez Saab

CINVESTAV-IPN

12 de Junio 2020

En esta práctica realizamos los pasos para configurar una red virtual que consiste en una máquina cliente y otra que actúa como puerta y cortafuegos, conectada a internet a través de la conexión NAT de VirtualBox.

1. Crear la máquina virtual para la puerta

Descargamos el archivo .iso disponible para la puerta y creamos una nueva máquina virtual llamada "Mi Puerta" con la misma configuración que en la primera práctica.

ſ		Create Virtual Machine	×	•	×	l
u	File	Arme and operating system Please choose a descriptive name and destination folder for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox Name: Mi Puerta Machine Folder: /home/kali/VirtualBox VMs Type: Linux Version: Linux 2.4 (64-bit)	4			
		Expert Mode < Back Next > Cancel Remote Desktop Server: Disabled Recording: Disabled				

En la configuración de la nueva máquina, vamos a la pestaña Storage>Controller:IDE y elegimos el .iso que descargamos para arrancar la máquina.

	MiPuer	ta - Settings	×		
General System Display Storage Audio Network Serial Ports USB Shared Folders User Interface	Storage Devices Controller: IDE Controller: SATA Controller: SATA	 < a kali Places Search Recently Used kali Desktop File System Kali Linux am Documents Music Pictures Videos Downloads 	Pease choose a virtual optical disk file. Desktop iunux1:so MiLinux1:so	 Size 13.3 MB 14.9 ME 10.8 MB 	Modified 05/18/2020 05/22/2020 Wednesday
				All virtual optio	al disk files 🗸 I 🛨 Open

Después vamos a Network>Advanced y elegimos el adaptador virtio. Este adaptador está conectado con NAT, ya que será el que se conecte a internet a través de la interfaz de VirtualBox.

Necesitaremos un segundo adaptador para que se conecte con la máquina cliente, así es que vamos a la pestaña de "Adapter 2" y lo habilitamos. Ya que el cliente también es una máquina en VirtualBox, para este adaptador elegimos la opción "internal network" en vez de NAT.

		Mi Puerta - Settings		×
📃 General				
 System Display Storage 	Adapter 1 Adapter 2 A	dapter 3 Adapter 4 oter Internal Network V		
V Audio V V Network Serial Ports V USB	Name: Advanced Adapter Type: Promiscuous Mode:	intnet Paravirtualized Network (virtio-net) Denv		
Shared Folders	MAC Address:	Cable Connected Port Forwarding		•
			Cancel	✓ ОК

La máquina virtual que creamos en la práctica anterior, llamada "Mi Linux Miniatura", es la que nos servirá de cliente. Como sólo se conectará a la puerta, necesitamos un solo adaptador y este debe estar en "internal network" también. Debemos asegurarnos que el nombre de la red sea el mismo para que las máquinas estén conectadas a la misma red virtual:

	MiLi	nux Miniatura - Settings	×
📃 General			
System	Adapter 1 Adapter 2 Ad	dapter 3 Adapter 4	
Display Storage	🐱 Enable Network Adapt	ter	
Audio	Attached to: Name: i	Internal Network 🗸	
Serial Ports	 Advanced Adapter Type: Promiscuous Mode: MAC Address: 	Paravirtualized Network (virtio-net)	~
USB Shared Folders		Deny	¥
📰 User Interface		0800271E3607	•
		Port Forwarding	
			e Cancel OK
			• Cancer V VK

2. Configurar la red

Iniciamos ambas máquinas virtuales (cliente y puerta), y modificamos sus archivos /etc/resolv.conf para incluir la dirección de nuestro DNS como en la práctica anterior.

Usando el comando ifconfig eth0 & ifconfig eth1 & route en la máquina puerta, podemos verificar que eth0 está conectado a la interfaz de VirtualBox, que como vimos en la práctica anterior, tiene el Default Gateway 10.0.2.2, mientras que eth1 está conectado a la red virtual con dirección 192.168.1.1, que es la misma red que al red física en mi hogar.

	Mi Puerta [Running] - Oracle VM VirtualBox _ 🗖 🗙									
File Machine	View Input Devic	es Help								
<pre>~ # ifconfig eth0 && ifconfig eth1 && route eth0 Link encap:Ethernet HWaddr 08:00:27:F0:B3:6F inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0 inet6 addr: fe80::a00:27ff:fef0:b36f/64 Scope:Link UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1 RX packets:0 errors:0 dropped:0 overruns:0 frame:0 TX packets:12 errors:0 dropped:0 overruns:0 carrier:0 collisions:0 txqueuelen:1000 RX bytes:0 (0.0 B) TX bytes:936 (936.0 B)</pre>										
eth1 Link encap:Ethernet HWaddr 08:00:27:81:37:9F inet addr:192.168.1.1 Bcast:192.168.1.255 Mask:255.255.255.0 inet6 addr: fe80::a00:27ff:fe81:379f/64 Scope:Link UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1 RX packets:0 errors:0 dropped:0 overruns:0 frame:0 TX packets:12 errors:0 dropped:0 overruns:0 carrier:0 collisions:0 txqueuelen:1000 RX bytes:0 (0.0 B) TX bytes:956 (956.0 B)										
Kernel IP rout:	Kernel IP routing table									
Destination	Gateway	Genmask	Flags Metric	: Ref Use	e Iface					
default	10.0.2.2	0.0.0.0	UG O	0 0) eth0					
10.0.2.0	*	255.255.255.0	U O	0 0) eth0					
192.168.1.0	*	255.255.255.0	U O	0 0) eth1					
~ # _										
💿 🎉 🗗 🖉 🗐 🖳 🖉 Right Ctrl										

Para evitar confusiones asignamos otro número a la red virtual. Para esto hemos elegido la red 10.0.3.0/24. En donde la puerta tendrá la dirección 10.0.3.1 y el cliente la dirección 10.0.3.2.

Para implementar esto usamos el comando

ifconfig eth1 10.0.3.1 netmask 255.255.255.0 broadcast 10.0.3.255

en la puerta para asignarle dicha dirección:

	MiP	Puerta [Runnin	g] - Oracle VM Virtua	lBox			_ = ×
File Machine	View Input	Devices	Help				
~ # ifconfig e	th1 10.0.3.1	netmask	255.255.255.	0 broad	lcast 10	0.0.3.255	
~ # route							
Kernel IP rout	ing table						
Destination	Gateway	Gei	nmask	Flags	Metric	Ref Us	se Iface
default	10.0.2.2	0.0	9.0.0	UG	0	Θ	0 eth0
10.0.2.0	*	255	5.255.255.0	U	0	Θ	0 eth0
10.0.3.0	*	255	5.255.255.0	U	Θ	Θ	0 eth1
# -							
				🕢 🔐	-	I 🔚 🗑 🖉 🖪	Right Ctrl
							Right Cut

Ahora debemos ir a la máquina cliente y conectar su única interfaz a la misma red. Para esto, usamos

ifconfig eth0 10.0.3.2 netmask 255.255.255.0 broadcast 10.0.3.255

y además registramos la dirección de la puerta como default gateway con el comando

route add default gw 10.0.3.1:

Mi Linux Miniatura [Running] - Oracle VM VirtualBox _ 💷 🗙								
File Machine	View Input Dev	ices Help						
~ # ifconfig e	eth0 10.0.3.2 ne	tmask 255.255.255.	.0 broadcast	10.0.3	.255			
~ # route add	default gw 10.0	.3.1						
Kernel IP rout	ting table							
Destination	Gateway	Genmask	Flags Meti	ic Ref	Use Iface			
default	10.0.3.1	0.0.0.0	UG O	Θ	0 eth0			
10.0.3.0	*	255.255.255.0	U O	Θ	0 eth0			
#								

Para asegurarnos de que la conexión es correcta, podemos hacer un ping desde la máquina cliente a la máquina puerta usando ping 10.0.3.1:

		Mi Linu	x Miniatu	ura (Runnii	ng] - Oracle VM Vir	tualBox				_ 0	×
File Machine	View	Input	Devic	es He	lp						
~ # ifconfig	eth0 10	9.0.3.	2 netr	mask 25	55.255.255.0	9 broad	lcast 10	9.0.3.	255		
~ # route add	defaul	lt gw	10.0.3	3.1							
~ # route											
Kernel IP rou	ting ta	able									
Destination	Gate	eway		Genma	ısk	Flags	Metric	Ref	Use	Iface	
default	10.0	9.3.1		0.0.0	9.0	UG	Θ	Θ	Θ	eth0	
10.0.3.0	×			255.2	255.255.0	U	Θ	Θ	Θ	eth0	
~ # ping 10.0	.3.1 -0	: 10									
PING 10.0.3.1	(10.0.	.3.1):	56 di	ata by1	tes						
64 bytes from	10.0.3	3.1: s	eq=0 t	ttl=64	time=1.256	ms					
64 bytes from	10.0.3	3.1: s	eq=1 i	ttl=64	time=1.065	ms					
64 bytes from	10.0.3	3.1: s	eq=2 t	ttl=64	time=0.559	ms					
64 bytes from	10.0.3	3.1: s	eq=3 t	ttl=64	time=1.080	ms					
64 bytes from	10.0.3	3.1: s	eq=4 i	ttl=64	time=0.595	ms					
64 bytes from	10.0.3	3.1: s	eq=5 t	ttl=64	time=0.707	ms					
64 bytes from	10.0.3	3.1: s	eq=6 i	ttl=64	time=0.532	ms					
64 bytes from	10.0.3	3.1: s	eq=7 t	ttl=64	time=1.067	ms					
64 bytes from	10.0.3	3.1: s	eq=8 t	ttl=64	time=0.657	ms					
64 bytes from	10.0.3	3.1: s	eq=9 t	ttl=64	time=0.996	ms					
10.0.3.1	ping st	tatist	ics								
10 packets tr	ansmitt	ted, 1	0 pacl	kets re	ceived, 0%	packet	t loss				
round-trip mi	n/avg/n	nax = 1	0.532/	/0.851/	/1.256 m s						
~ #											

3. Configuración de la puerta y cortafuegos

Ahora configuramos la puerta. Debemos editar el script /etc/fw_nat2.sh que hace la configuración del firewall usando iptables para habilitar los servicios de ssh (22), http (80), dns (53) y https (443) solamente. En el encabezado del archivo, modificamos el valor de IPADDR para que sea la dirección que se conecta a la interfaz de VirtualBox (es decir 10.0.2.15) así como el valor de REDLOCAL, DNS (donde insertamos la dirección del DNS que da nuestro ISP) y REDINTERNA (donde introducimos la red 10.0.3.0/24 que creamos).



Finalmente, usamos el comando echo 1 > /proc/sys/net/ipv4/ip_forward para habilitar el traspaso de paquetes y ejecutamos nuestro script:

Mi Puerta [Running] - Oracle VM VirtualBox	_ = ×
File Machine View Input Devices Help	
<pre>"# echo 1 > /proc/sys/net/ipv4/ip_forward " # sh /etc/fw_nat2.sh " # _</pre>	

4. Prueba de conexión y del cortafuegos

Ahora nuestra red virtual está completamente configurada. Para verificar que la máquina cliente tiene acceso a internet, podemos usar un telnet 172.217.7.46 80 (que es la dirección de google.com) lo cual es válido porque el cortafuegos admite conexiones a direcciones externas por el puerto 80 (http), pero si usamos telnet 172.217.7.46 1025 entonces no hay respuesta ya que el cortafuegos no permite conectarse a ningún puerto de usuario de una dirección externa.

MiLinux Miniatura (Running) - Oracle VM VirtualBox _ O X File Machine View Input Devices Help ⁷ # telnet 172.217.7.46 80 Connected to 172.217.7.46 ⁷ C ⁸ Console escape. Commands are: ¹ go to line mode ² go to character mode ² suspend telnet ^e exit telnet ⁷ # telnet 172.217.7.46 1025 ⁸ C ⁹ I I I I I I I I I I I I I I I I I I I				rpeables -n	ronwinnb -
File Machine View Input Devices Help * # telnet 172.217.7.46 80 Connected to 172.217.7.46 Console escape. Commands are: a 1 go to line mode c go to character mode z suspend telnet e exit telnet * # telnet 172.217.7.46 1025 R R R R R R R R R R R R R	. N	1i Linux Miniatura (F	Running] - Oracle VM VirtualBox		- • ×
<pre>* # telnet 172.217.7.46 80 Connected to 172.217.7.46 *C B Console escape. Commands are: a 1 go to line mode c go to character mode z suspend telnet e exit telnet * # telnet 172.217.7.46 1025 ************************************</pre>	File Machine View In	put Devices	Help		1
Console escape. Commands are: 1 go to line mode c go to character mode z suspend telnet e exit telnet 7 # telnet 172.217.7.46 1025 8 8 8 8 8 8 9 10 10 10 10 10 10 10 10 10 10	" # telnet 172.217.7.4 Connected to 172.217.7	16 80 7.46			3
a l go to line mode c go to character mode z suspend telnet e exit telnet 7 # telnet 172.217.7.46 1025 R R R R R R R R R R R R R R R R R R R		de anel			
t go to line mode c go to character mode z suspend telnet e exit telnet " # telnet 172.217.7.46 1025 R R R R R	sconsole escape. Comman	ids are;			
z suspend telnet e exit telnet ⁷ # telnet 172.217.7.46 1025 R R R R	A l go to line mod c go to characte	le er mode			
e exit telnet [~] # telnet 172.217.7.46 1025 R R R R R R B B B B B B B B B B B B B	z suspend telnet				
# ternet 172.217.7.46 1925	e exit telnet	6 1025			
R R I III III IIII IIII IIII IIII IIII IIIII IIIII IIIII IIIII IIIII IIIII IIIII IIIII IIII	/ # telnet 172.217.7.5	1025			
R 2 9 M 🗗 🖉 🖉 🖾 Right Ctrl					
S M P D III S V S Right Ctrl	* 6				
े । 🖓 🏕 🗐 🖽 🖉 🖉 Right Ctrl					
o 🔰 🗗 🖉 🖉 🖸 Right Ctrl					
o 🔰 🗗 🖉 🖉 🖸 Right Ctrl					
3 🔰 문 🏕 🗐 📴 🖉 🖉 Right Ctrl					
💿 🂵 🗗 🏕 🔲 🗐 🗗 🗷 🥔 🖸 Right Ctrl					
💿 🂵 🗗 🏕 🔲 🗐 🗗 🗵 🖉 🖸 Right Ctrl					
🌀 🂵 🗗 🏕 🔲 🗐 🚰 🗷 Right Ctrl					
💿 🄰 🗗 🖉 🔛 🔀 Right Ctrl					
			O H	🗗 🤌 📖 🖾 🚰 🖉 🥝 Rig	ht Ctrl

Por otro lado, si usamos telnet 192.168.1.1 53 entonces logramos conectarnos a nuestro DNS por el puerto 53, pero con telnet 172.217.7.48 53 no hay respuesta ya que el cortafuegos solo permite trafico por el puerto 53 si va hacia la dirección del DNS que introdujimos.

