Temat: Ćwiczenie 3 Routing i zdalne zarządzanie serwerem

W jakim celu?

Routing to usługa która pozwala przekazywać Internet z serwera klientowi nie tracąc go na serwerze. Wykorzystujemy translację adresów NAT czasem nazywaną masqarade by mając do dyspozycji jeden adres IP obsłużyć więcej niż jeden komputer. Zdalne zarządzanie zaś pozwoli zasymulować iż nie musimy być fizycznie przy serwerze i większość prac tak jest wykonywanych.

Co się nauczymy?

Sprawimy iż zacznie działać Internet na kliencie oraz zdalne zarządzanie serwerem. Wykonamy to poprzez usługę ssh czyli secure shell (zdalną konsolę). Rozważymy też opcjonalne zdalne połączenie do graficznego interfejsu użytkownika przychwytując zdalnie ekran poprzez protokół VNC.

Co mamy?

Na wstępie mamy :

- maszynę wirtualną klienta Ubuntu 21.10 w wersji z domyślnym GUI czyli GNOME zainstalowanym na Virtualboxie jako legacy BIOS z kontem administracyjnym ad mającym hasło 123.Dysk twardy maszyny jest w formacie vdi i ma 50GB.
- maszynę wirtualną serwera Ubuntu 21.10 w wersji bez GUI zainstalowanym na Virtualboxie jako legacy BIOS z kontem administracyjnym ad mającym hasło 123.Dysk twardy maszyny jest w formacie vdi i ma 40GB.

Maszyny startowe (początek ćwiczenia): klient Linux serwer Linux

Wynikowe maszyny (koniec ćwiczenia): klient Linux serwer Linux

Polecenia ćwiczenia:

• Mając skonfigurowane interfejsy serwera w wirtualboxie i systemie operacyjnym wg poniższego diagramu sprawdź komunikację klient serwer, serwer klient, serwer internet

Klient	Server		
enp0s3	enp0s3	enp0s8	
Sieć wewnętrzna IP i pozostałe parametry z DHCP	Sieć wewnętrzna IP 10.0.0.2/24	sieć : NAT	
	Brama 10.0.0.1 DNS 10.0.0.1, 8.8.8.8 DHCP client (auto)		

Sieć wewnętrzna inet

Internet

- Sprawdź czy mamy zainstalowany serwer ssh. Jeśli go nie ma zainstaluj.
- Połącz się zdalnie z klienta na serwer poprzez ssh
- (opcjonalnie) Zainstaluj na serwerze GUI i usługę VNC i połącz się do serwera graficznie poprzez protokół VNC uruchamiając na kliencie Remmina i podając nazwę komputera do którego się łączymy - server
- Zainstaluj i skonfiguruj usługę routingu
- Sprawdź działanie Internetu na serwerze i kliencie

Jak zrobić

Uruchamiamy maszyny wirtualne serwera i klienta. Po uruchomieniu logujemy się na użytkownika ad z hasłem 123.

Połączenie przez ssh

Serwer ma już uprzednio zainstalowaną usługę sshd której status możemy sprawdzić

sudo systemctl status sshd

ad@server:~\$	sudo systemati status sshd					
[sudo] passwo	ord for ad:					
 ssh.service 	e – OpenBSD Secure Shell server					
Loaded:	<pre>loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)</pre>					
Active:	active (running) since Sun 2022–05–15 12:04:49 UTC; 48s ago					
Docs:	man:sshd(8)					
	man:sshd_config(5)					
Process:	632 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=O/SUCCESS)					
Main PID:	679 (sshd)					
Tasks:	1 (limit: 2241)					
Memory:	4.5M					
CPU:	16ms					
CGroup:	/system.slice/ssh.service					
	└─679 ``sshd: /usr/sbin/sshd –D [listener] O of 10–100 startups``					
mai 15 10.01	40 conversion during fill, Ctarting Occurrence Chall convers					
maj 15 12:04) maj 15 12:04	:49 server systemu[1]: Starting openBSD Secure Shell server					
maj 15 12:04 maj 15 12:04	49 Server SShu[075]: Server listening on 0.0.00 port 22.					
maj 15 12:04. maj 15 12:04	19 server sshulo(2), server iistening on ,, port 22.					
ad@server:~\$	A Server Systema[1]. Started openbod secure shell server.					

Łaczymy się na serwer:

W konsoli na kliencie wpisujemy

ssh <u>ad@10.0.0.1</u>

gdzie ad jest nazwą użytkownika serwera

(]+L			admserver.~			_		4
ad@PC:~\$ s: The authen ECDSA key i Are you sur Please type Warning: Pe ad@192.168 Welcome to * Document	sh ad@192.16 ticity of ho fingerprint e you want e 'yes', 'no ermanently ac 5.13's pass Ubuntu 22.0 tation: http	8.5.13 st '192.168. is SHA256:m8 to continue ' or the fin dded '192.16 word: 4 LTS (GNU/L ps://help.ub	5.13 (192.168.5.13 5mBZxEak64HkQ7WQ/(connecting (yes/no gerprint: yes 8.5.13' (ECDSA) to inux 5.15.0-27-ger untu.com	3)' can' Q6ePezzV p/[finge p the li neric x8	t be esta +7HURMg+T rprint])? st of kno 6_64)	blish rxo10 y wn ho	ued. DQk. Dosts.	
* Manageme * Support System in	nt: ntt http nformation a:	ps://landsca ps://ubuntu. s of nie, 15	pe.canonical.com com/advantage maj 2022, 12:08:3	35 UTC				
System lo	oad: 0.0019	53125 f 58.75GB	Processes: Users logged in:		121 1			

(na zrzucie ekranu jest inny adres IP gdyż został wykonany w innym środowisku)

(Opcjonalnie) Instalujemy serwer graficznego środowiska na serwerze Korzystamy z poradnika np. : <u>https://linuxconfig.org/ubuntu-22-04-gui-installation</u>

Jeśli chcemy środowisko xfce: sudo apt-get install tasksel

sudo tasksel

sudo tasksel (wybieramy Xubuntu desktop czyli xfce) Po instalacji weryfikujemy środowisko graficzne komendą xstart

na końcu sudo reboot i opcjonalnie instalujemy dodatki gościa.

(Opcjonalnie) Instalujemy serwer VNC na serwerze

Korzystamy z poradnika np. : <u>https://www.server-</u> world.info/en/note?os=Ubuntu_22.04&p=desktop&f=6

Serwer VNC instalujemy komendą

sudo apt install tigervnc-standalone-server

vncpasswd (nadajemy hasło zdalnego połączenia min.6znaków np. 12345678)

Po ustawieniu hasła uruchamiamy serwer: tigervncserver -xstartup /usr/bin/xfce4-session -geometry 800x600 -localhost :1 ==

Jeśli chcemy środowisko Mate :

sudo apt -y install tigervnc-standalone-server (Instalujemy VMC Serwer) sudo apt install tasktel

(Instalujemy tasksel)

Wybieramy Desktop Ubuntu Mate. W folderze ~/.vnc tworzymy plik "xstartup". Edytujemy plik wpisując do niego podany skrypt:

#!/bin/sh

===

Start up the standard system desktop unset SESSION MANAGER unset DBUS SESSION BUS ADDRESS

/usr/bin/mate-session

[-x/etc/vnc/xstartup] && exec /etc/vnc/xstartup [-r \$HOME/.Xresources] && xrdb \$HOME/.Xresources *x-window-manager* & ____

(Zmieniamy uprawnienia pliku komendą) *chmod* +*x* ~*/.vnc/xstartup*

Otwieramy sesję VNC komendą:

tigervncserver -xstartup ~/.vnc/xstartup -geometry 800x600 -localhost no :1

Łączymy się z VNC za pomocą innej maszyny wpisując jej nazwę i port np: server:5901

(Opcjonalnie) Łączymy się z klienta programem Remmina do serwera poprzez protokół VNC przechwytując ekran serwera

uruchamiamy program Remmina i wpisujemy adres ip serwera i port Na kliencie domyślny 5901, wpisujemy hasło ustawione wcześniej.

Na kliencie poprzez program Remmina łączymy się z serwerem wpisując 10.0.0.1:5901

	Klien	ıt zdalnego	pulpitu Re	emmina 🗄 🗏	_	\otimes
VNC ~	10.0.0.1	1				8
Nazwa 🗸	Grupa	Serwer	Plugin	Last used		

Na zrzucie brak jest portu-wpisz to co w tekście!

W efekcie zdalnie zarządzamy serwerem

🖉 Menu 🍅 🕴 🕴	L
🖾 ad@server: ~ 🛛 👘 📈	
File Edit View Search Terminal Help	
Use xtigervncviewer -SecurityTypes VncAuth -passwd /home/ad/.vnc/passwd :1 to co nnect to the VNC server.	
======================================	
Session startup via '/home/ad/.vnc/xstartup' cleanly exited too early (< 3 secon ds)!	
Maybe try something simple first, e.g., tigervncserver -xstartup /usr/bin/xterm The X session cleanly exited! Killing Xidervac process ID 3117 success!	
ad@server:~\$ tigervncserver -xstartup /usr/bin/mate-session -geometry 800x600 -l ocalhost no :1	
New Xtigervnc server 'server:1 (ad)' on port 5901 for display :1. Use xtigervncviewer -SecurityTypes VncAuth,TLSVnc -passwd /home/ad/.vnc/passwd s erver:1 to connect to the VNC server.	
ad@server:~\$	
Image: Second	Ē

Routing

Korzystamy z poradnika <u>https://www.networkreverse.com/2020/06/how-to-build-linux-router-with-ubuntu.html</u>

sudo nano /etc/sysctl.conf	(edytujemy plik)
znajdź i odkomentuj	net.ipv4.ip	forward=1

sudo sysctl –p

(wdrażamy konfigurację)

sudo iptables - A FORWARD - i enp0s8 - o enp0s3 - j ACCEPT

sudo iptables - A FORWARD - i enp0s3 - o enp0s8 - m state -- state RELATED, ESTABLISHED - j ACCEPT

sudo iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE

sudo apt install iptables-persistent

sudo su —

(przejście na roota)

iptables-save > /etc/iptables/rules.v4

ping 10.0.0.3

ping 8.8.8.8

Przechodzimy na klienta. Powinniśmy teraz już móc korzystać z Internetu

Podsumowanie

Skonfigurowaliśmy solidne podłoże do kolejnych ćwiczeń i możemy w następnym ćwiczeniu skoncentrować się na serwerze http