SEGUNDA PARTE

Realice un túnel hacia el servidor web implementado en clase, de manera que los recursos de su servidor web puedan ser visualizados desde cualquier lugar por fuera de su red local. Para efectos de prueba, agregar una página personalizada a su sitio web.

Se sugiere usar:

- Port forwarding en Vagrant.
- Vagrant share.
- ngrok (agregarlo al path –variables de entorno-- una vez instalado).

Forwarded Ports

Vagrant forwarded ports permiten acceder a un puerto en su máquina host y hacer que todos los datos se reenvíen a un puerto en la máquina invitada, ya sea a través de TCP o UDP. Por ejemplo: si la máquina invitada está ejecutando un servidor web que escucha en el puerto 80, puede realizar una asignación de puerto reenviado al puerto 8080 (o cualquier cosa) en su máquina host. A continuación, puede abrir su navegador en localhost: 8080 y navegar por el sitio web, mientras todos los datos de red reales se envían al invitado.

La configuración del *forwarded port* configuration espera dos parámetros, el puerto en el invitado y el puerto en el host. Ejemplo:

```
Vagrant.configure("2") do |config|
  config.vm.network "forwarded_port", guest: 80, host: 8080
end
```

ngrok

ngrok allows you to expose a web server running on your local machine to the internet. Para instalarlo y configurarlo se debe hacer lo siguiente:

- Descomprimir para instalar: En Windows, simplemente haz doble clic en ngrok.zip.
- Conecte su cuenta: Ejecutar este comando agregará su authtoken a su archivo ngrok.yml. Conectar una cuenta mostrará una lista de sus túneles abiertos en el tablero, le dará tiempos de espera de túnel más largos y más. Visite el panel para obtener su token de autenticación.
- Agregue la ruta donde descomprimió el archivo ejecutable a la variable de entorno PATH de su cuenta.
- Para comprobar que la instalación fue exitosa, abra una terminal y use el comando: *ngrok version*.

Vagrant Share

Vagrant Share le permite compartir su entorno Vagrant con cualquier persona del mundo, lo que permite la colaboración directamente en su entorno Vagrant en casi cualquier entorno de red con un solo comando: vagrant share. Vagrant share tiene tres modos o funciones principales. Estas características no son mutuamente excluyentes, lo que significa que cualquier combinación de ellas puede estar activa en cualquier momento:

Vagrant Share es un complemento de Vagrant que **debe instalarse**. No se incluye con los paquetes del sistema Vagrant. Vagrant Share requiere el uso de ngrok. Para instalar el complemento Vagrant Share, ejecute el siguiente comando:

```
# vagrant plugin install vagrant-share
```

HTTP

El modo de función que se necesita es HTTP sharing, el cual creará una URL que puede proporcionarle a cualquier persona. Esta URL se enrutará directamente a su entorno de Vagrant. La persona que usa esta URL no necesita tener Vagrant instalado, por lo que se puede compartir con cualquier persona. Esto es útil para probar webhooks o mostrar su trabajo a clientes, compañeros de equipo, gerentes, etc. Para usar el uso compartido de HTTP, simplemente ejecute vagrant share:

```
# vagrant share
==> default: Detecting network information for machine...
default: Local machine address: 192.168.84.130
default: Local HTTP port: 9999
default: Local HTTPS port: disabled
==> default: Creating Vagrant Share session...
==> default: HTTP URL: http://b1fb1f3f.ngrok.io
```

Vagrant detecta dónde se está ejecutando su servidor HTTP en su entorno de Vagrant y genera el punto final que se puede usar para acceder a este recurso compartido. Simplemente proporcione esta URL a cualquier persona con la que desee compartirla.

Si Vagrant tiene problemas para detectar el puerto de sus servidores en su entorno, use las marcas --http y / o --https para ser más explícito.

El recurso compartido será accesible durante el tiempo que se ejecute el recurso compartido vagrant. Presione Ctrl-C para salir de la sesión para compartir.

SSH

Vagrant share hace que sea trivialmente fácil permitir el acceso SSH remoto a su entorno de Vagrant proporcionando el indicador --ssh a vagrant share.

El uso compartido sencillo de SSH es increíblemente útil si desea dar acceso a un colega para solucionar problemas de operaciones. Además, permite la programación en pareja con un entorno Vagrant, si lo desea.

```
# vagrant share --ssh
==> default: Detecting network information for machine...
default: Local machine address: 192.168.84.130
==> default: Generating new SSH key...
default: Please enter a password to encrypt the key:
default: Repeat the password to confirm:
default: Inserting generated SSH key into machine...
default: Local HTTP port: disabled
default: Local HTTPS port: disabled
default: SSH Port: 2200
==> default: Creating Vagrant Share session...
share: Cloning VMware VM: 'hashicorp/vagrant-share'. This can take some time...
share: Verifying vmnet devices are healthy...
share: Preparing network adapters...
share: Starting the VMware VM...
share: Waiting for machine to boot. This may take a few minutes...
share: SSH address: 192.168.84.134:22
share: SSH username: tc
share: SSH auth method: password
```

```
share:
share: Inserting generated public key within guest...
share: Removing insecure key from the guest if it's present...
share: Key inserted! Disconnecting and reconnecting using new SSH key...
share: Machine booted and ready!
share: Forwarding ports...
share: -- 31338 => 65534
share: -- 22 => 2202
share: SSH address: 192.168.84.134:22
share: SSH username: tc
share: SSH auth method: password
share: Configuring network adapters within the VM...
==> share:
==> share: Your Vagrant Share is running! Name: bazaar_wolf:sultan_oasis
==> share:
==> share: You're sharing with SSH access. This means that another can SSH to
==> share: your Vagrant machine by running:
==> share:
==> share:   vagrant connect --ssh bazaar_wolf:sultan_oasis
==> share:
```

Cualquiera puede luego SSH directamente a su entorno Vagrant ejecutando vagrant connect --ssh NAME donde NAME es el nombre del recurso compartido generado anteriormente.

```
# vagrant connect --ssh bazaar_wolf:sultan_oasis
Loading share 'bazaar_wolf:sultan_oasis'...
The SSH key to connect to this share is encrypted. You will
require the password entered when creating the share to
decrypt it. Verify you have access to this password before
continuing.

Press enter to continue, or Ctrl-C to exit now.
Password for the private key:
Executing SSH...
Welcome to Ubuntu 12.04.3 LTS (GNU/Linux 3.8.0-29-generic x86_64)
```

Connect

Vagrant puede compartir cualquiera o todos los puertos a su entorno Vagrant, no solo SSH y HTTP. El comando vagrant connect le da a la persona que se conecta una IP estática que puede usar para comunicarse con el entorno compartido de Vagrant. Cualquier tráfico TCP enviado a esta IP se envía al entorno compartido de Vagrant. Solo se llama “vagrant share --full” y se procede como en el caso del ssh.

Security

Es comprensible que compartir su entorno de Vagrant plantee una serie de problemas de seguridad.

El mecanismo de seguridad principal para Vagrant Share es la seguridad a través de la oscuridad junto con una clave de cifrado para SSH. Además, hay varias opciones de configuración disponibles para ayudar a controlar el acceso y administrar la seguridad:

- --disable-http no creará una URL HTTP de acceso público. Cuando está configurado, la única forma de acceder al recurso compartido es con vagrant connect.

TERCERA PARTE

Paso 1: Instale los paquetes necesarios para la configuración de PXE

Para instalar y configurar el servidor PXE en centos se necesitan los siguientes paquetes “dhcp, tftp-server, ftp server (vsftpd), xinted”.

```
# yum install dhcp tftp tftp-server syslinux vsftpd xinetd
```

Paso 2: Configurar el servidor DHCP para PXE

El protocolo de configuración dinámica de host (en inglés: Dynamic Host Configuration Protocol, también conocido por sus siglas de DHCP) es un protocolo de red de tipo cliente/servidor mediante el cual un servidor DHCP asigna dinámicamente una dirección IP y otros parámetros de configuración de red a cada dispositivo en una red para que puedan comunicarse con otras redes IP. Este servidor posee una lista de direcciones IP dinámicas y las va asignando a los clientes conforme éstas van quedando libres, sabiendo en todo momento quién ha estado en posesión de esa IP, cuánto tiempo la ha tenido y a quién se la ha asignado después. Así los clientes de una red IP pueden conseguir sus parámetros de configuración automáticamente.

Cuando instalamos el paquete dhcp, se crea un archivo de configuración de muestra del servidor dhcp en “/usr/share/doc/dhcp*/dhcpd.conf.example“, aunque el archivo de configuración de dhcp está en '/etc/dhcp/dhcpd.conf'.

Copie las siguientes líneas en el archivo "/etc/dhcp/dhcpd.conf", reemplace la subred ip y otros detalles según su entorno.

```
# vim /etc/dhcp/dhcpd.conf

# DHCP Server Configuration file.
ddns-update-style interim;
ignore client-updates;
authoritative;
allow booting;
allow bootp;
allow unknown-clients;

# Internal subnet for DHCP Server.
subnet 192.168.50.0 netmask 255.255.255.0 {
range 192.168.50.5 192.168.50.250;
option domain-name-servers 192.168.50.4;
option domain-name "www.serverpxe.com";
option routers 192.168.50.4;
option broadcast-address 192.168.50.255;
default-lease-time 600;
max-lease-time 7200;

# IP of PXE Server
next-server 192.168.50.4;
filename "pxelinux.0";
```

Paso 3: Edite y configure el servidor tftp (/etc/xinetd.d/tftp)

TFTP son las siglas de Trivial file transfer Protocol (Protocolo de transferencia de archivos trivial).

Es un protocolo de transferencia muy simple semejante a una versión básica de FTP. TFTP a menudo se utiliza para transferir pequeños archivos entre computadoras en una red, como cuando un cliente ligero arranca desde un servidor de red.

- Utiliza UDP (en el puerto 69) como protocolo de transporte (a diferencia de FTP que utiliza los puertos 20 y 21 TCP).
- No puede listar el contenido de los directorios.
- No existen mecanismos de autenticación o cifrado.
- Se utiliza para leer o escribir archivos de un servidor remoto.

En el caso de la configuración del servidor PXE, tftp se utiliza para la carga de arranque. Para configurar el servidor tftp, edite su archivo de configuración "/etc/xinetd.d/tftp", cambie el parámetro "disable = yes" a "disable = no" y deje los otros parámetros como están.

```
# vim /etc/xinetd.d/tftp

service tftp
{
 socket_type = dgram
 protocol    = udp
 wait        = yes
 user        = root
 server      = /usr/sbin/in.tftpd
 server_args = -s /var/lib/tftpboot
 disable     = no
 per_source  = 11
 cps         = 100 2
 flags       = IPv4
}
```

Todos los archivos relacionados con el arranque de red deben colocarse en el directorio raíz de tftp "/var/lib/tftpboot/". Ejecute los siguientes comandos para copiar los archivos de arranque de red en dicho directorio.

```
# cp -v /usr/share/syslinux/pxelinux.0 /var/lib/tftpboot
# cp -v /usr/share/syslinux/menu.c32 /var/lib/tftpboot
# cp -v /usr/share/syslinux/memdisk /var/lib/tftpboot
# cp -v /usr/share/syslinux/mboot.c32 /var/lib/tftpboot
# cp -v /usr/share/syslinux/chain.c32 /var/lib/tftpboot

# mkdir /var/lib/tftpboot/pxelinux.cfg
# mkdir /var/lib/tftpboot/networkboot
```

Paso 4: Monte el archivo ISO de CentOS 7.x y copie su contenido en el servidor ftp local

Puede obtener la imagen ISO a partir de este enlace:

http://ftp.iij.ad.jp/pub/linux/centos-vault/7.2.1511/isos/x86_64/

Ejecute los siguientes comandos para montar el archivo ISO y luego copie su contenido en el directorio del servidor ftp "/var/ftp/pub".

```
# mount -o loop CentOS-7-x86_64-DVD-1511.iso /mnt/
mount: /dev/loop0 is write-protected, mounting read-only

# cd /mnt/
# cp -av * /var/ftp/pub/
```

Copie el archivo Kernel (vmlimz) y el archivo initrd del archivo ISO montado a "/var/lib/tftpboot/networkboot/".

```
# cp /mnt/images/pxeboot/vmlinuz /var/lib/tftpboot/networkboot/
# cp /mnt/images/pxeboot/initrd.img /var/lib/tftpboot/networkboot/
```

Puede desmontar el archivo ISO usando el comando "umount".

```
# umount /mnt/
```

Paso 5: Crear archivo de menú kickStart y archivo de menú PXE

Antes de crear un archivo kickstart, primero se crea la contraseña de root en una cadena encriptada que se usará en el archivo kickstart.

```
[root@pxe ~]# openssl passwd -1 nvargas
$1$2mgCwtcO$aEIlJAFYsvVaspPJWNsTU0
```

El archivo kickstart predeterminado del sistema se coloca en /root con el nombre "anaconda-ks.cfg". Crearemos un nuevo kickstart en la carpeta /var/ftp/pub con el nombre "centos7.cfg".

Copie el siguiente contenido en el nuevo archivo kickstart. Modifique el archivo kickstart según sus necesidades. En este archivo van todas las opciones que se eligen automáticamente durante la instalación.

```
# vim /var/ftp/pub/centos7.cfg

# Firewall configuration
firewall --disabled

# Install OS instead of upgrade
install

# Use FTP installation media
url --url="ftp://192.168.50.4/pub/"

# Root password
rootpw --iscrypted $1$2mgCwtcO$aEIlJAFYsvVaspPJWNsTU0

# System authorization information
auth useshadow passalgo=sha512

# Use graphical install
graphical
firstboot disable

# System keyboard
keyboard us
```

```
# System language
lang es_ES

# SELinux configuration
selinux disabled

# Installation logging level
logging level=info

# System timezone
timezone America/Bogota

# System bootloader configuration
bootloader location=mbr
clearpart --all --initlabel
part swap --asprimary --fstype="swap" --size=1024
part /boot --fstype xfs --size=300
part pv.01 --size=1 --grow
volgroup root_vg01 pv.01
logvol / --fstype xfs --name=lv_01 --vgname=root_vg01 --size=1 --grow
%packages
@^minimal
@core
%end
%addon com_redhat_kdump --disable --reserve-mb='auto'
%end
```

Cree un archivo de menú PXE (/var/lib/tftpboot/pxelinux.cfg/default), copie el siguiente contenido en el archivo de menú PXE. Es el menú que aparece durante el arranque.

```
# vim /var/lib/tftpboot/pxelinux.cfg/default

default menu.c32
prompt 0
timeout 60
MENU TITLE Menu PXE para instalar CentOS 7
LABEL centos7_x64
MENU LABEL CentOS 7_X64
KERNEL /networkboot/vmlinuz
APPEND initrd=/networkboot/initrd.img inst.repo=ftp://192.168.50.4/pub
ks=ftp://192.168.50.4/pub/centos7.cfg
```

Paso 6: Inicie y habilite el servicio xinetd, dhcp y vsftpd

Utilice los siguientes comandos para iniciar y habilitar xinetd, dhcp y vsftpd.

```
# systemctl start xinetd
# systemctl enable xinetd

# systemctl start dhcpd.service
# systemctl enable dhcpd.service
Created symlink from /etc/systemd/system/multi-user.target.wants/dhcpd.service to
/usr/lib/systemd/system/dhcpd.service.

# systemctl start vsftpd
# systemctl enable vsftpd
Created symlink from /etc/systemd/system/multi-user.target.wants/vsftpd.service to
```

/usr/lib/systemd/system/vsftpd.service.

// In Case SELinux is enabled, then set the following selinux rule for ftp server.
setsebool -P allow_ftpd_full_access 1

// Open the ports in the OS firewall using the following firewall-cmd commands.
firewall-cmd --add-service=ftp --permanent
firewall-cmd --add-service=dhcp --permanent
firewall-cmd --add-port=69/tcp --permanent
firewall-cmd --add-port=69/udp --permanent
firewall-cmd --add-port=4011/udp --permanent
firewall-cmd --reload

INTEGRANTES

- Natalia Rodriguez Mesa
- Natalia Vargas Insuasti