UNIVERSIDAD CENTRAL DEL ECUADOR
PROTOCOLOS DE COMUNICACIÓN DE DATOS

NOMBRE: ALCÍVAR RODRIGUEZ GUSTAVO ALFONSO
FECHA: QUITO, 07 DICIEMBRE 2014
SEMESTRE: 2014-2015 **PRIMER HEMISEMESTRE**

IP (INTERNET PROTOCOL)

IP, como el protocolo principal de la capa de Internet del conjunto de protocolos de Internet, tiene la tarea de entregar los paquetes de la fuente de host al host de destino basado únicamente en las direcciones IP. Se utiliza los siguientes comandos:

#ping [dirección IP]
#route –n //Muestra tabla de rutas
#ip route
#ip route add //Añadir ruta
#ip route del //Quitar una ruta
#arping //Permite detectar direcciones IP duplica

TCP (TRANSMISSION CONTROL PROTOCOL)

Es uno de los protocolos fundamentales en Internet. El protocolo garantiza que los datos serán entregados en su destino sin errores y en el mismo orden en que se transmitieron. También proporciona un mecanismo para distinguir distintas aplicaciones dentro de una misma máquina, a través del concepto de puerto. Se utiliza los siguientes comandos:

Netstat:
#netstat –platune
#netstat –atun

UDP (USER DATAGRAM PROTOCOL)

Es un protocolo del nivel de transporte basado en el intercambio de datagramas (Encapsulado de capa 4 Modelo OSI). Permite el envío de datagramas a través de la red sin que se haya establecido previamente una conexión, ya que el propio datagrama incorpora suficiente información de direccionamiento en su cabecera. Tampoco tiene confirmación ni control de flujo, por lo que los paquetes pueden adelantarse unos a otros; y tampoco se sabe si ha llegado correctamente, ya que no hay confirmación de entrega o recepción.
#netstat

TELNET

Telnet (Teletype Network1) es el nombre de un protocolo de red que nos permite viajar a otra máquina para manejarla remotamente como si estuviéramos sentados delante de ella.
Telnet sólo sirve para acceder en modo terminal, es decir, sin gráficos, pero es una herramienta muy útil para arreglar fallos a distancia, sin necesidad de estar físicamente en el mismo sitio que la máquina que los tenía. Puerto por defecto: 23 en TCP. Para comenzar con la configuración es necesario seguir los siguientes pasos:

CONFIGURACION

#yum install –y telnet telnet server

Acceder al archivo de configuración, este arhivo contiene las siguientes lineas lo que debemos cambia la parte sombreada de manera que este sea el resultado.

#vim /etc/sysconfig/selinux

```
# This file controls the state of SELinux on the system.
 # SELINUX= can take one of these three values:
 # enforcing - SELinux security policy is enforced.
 # permissive - SELinux prints warnings instead of enforcing.
 # disabled - No SELinux policy is loaded.
 SELINUX=disabled
 # SELINUXTYPE= can take one of these two values:
 # targeted - Targeted processes are protected,
 # minimum - Modification of targeted policy. Only selected processes are protected.
 # mls - Multi Level Security protection.
 SELINUXTYPE=targeted
```

Inicializamos los servicios para confirmar los cambios:

 # systemctl start telnet.socket
 # systemctl enable telnet.socket

Para probar que telnet funcione correctamente hacemos la siguiente prueba:

#telnet 10.3.0.33 luego nos pide el usuario de la maquina a la que nos estamos conectando
uce login: cliente
Password:

Y se conecta correctamente, creamos modificamos archivos para verificar que funciona correctamente y por último capturamos tramas en el Wireshark .
Con la ayuda de Wireshark vamos a poder identificar la clave y contraseña ya que nos es un protocolo de seguridad y que gracias a esta herramienta va a ser muy fácil detectarla.

SSH (SECURE SHELL)

Es similar a telnet, con la diferencia que SSH usa técnicas de cifrado que hacen que la información que viaja por el medio de comunicación vaya de manera no legible y ninguna tercera persona pueda descubrir el usuario y contraseña de la conexión ni lo que se escribe durante toda la sesión. Puerto por defecto: 22.

CONFIGURACIÓN:
Para configurar este protocolo no es necesario descargar ningún paquete.
Es necesario configurar un solo archivo .
#cd /etc/ssh/
#vim sshd_config
Identificamos en este archivo de configuración la siguiente línea , y la reemplazamos con la siguiente línea:

PasswordAuthentication yes
PasswordAuthentication no

Iniciamos servicios, para asegurar que se realicen los cambios:

 # systemctl restart sshd.service

Para realizar la prueba en el terminal y de igual manera que en telnet capturamos las tramas y verificamos que con ssh no se puede identificar ningún dato de la sesión con ssh.

#ssh 10.3.0.33 (ip del host al que vamos a conectarnos).

DHCP (DYNAMIC HOST CONFIGURATION PROTOCOL)

Protocolo de red que permite a los clientes de una red IP obtener sus parámetros de configuración automáticamente. Se trata de un protocolo de tipo cliente/servidor en el que generalmente un servidor posee una lista de direcciones IP dinámicas y las va asignando a los clientes conforme éstas van estando 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.

CONFIGURACION

#yum install –y dhcp

Nos vamos al archivo de configuración siguiente:

#gedit /etc/dhcp/dhcpd.conf

Realizamos la configuración necesaria para que pueda asignar las direcciones IP de manera que sea de una manera automática en un rango de [20; 80].

default-lease-time 6000; # tiempo en segundos del "alquiler"
max-lease-time 7200; # máximo tiempo en segundos que durara el "alquiler"
option ntp-servers 10.3.0.30;
especificación de un rango dentro de la subred a la que pertenece el servidor
subnet 10.3.0. 0 netmask 255.255.0.0 {
range 10.3.0.20 10.3.0.80; # rango que asigna a los clientes
option broadcast-address 10.3.255.255; # dirección de difusión
option routers 10.3.0.1; # puerta de enlace
}
configuración particular para un host
host pc01 {
hardware ethernet 08:00:27:DE:3A:46; # mac del host
fixed-address 10.3.0.33; # ip a asignar (siempre la misma)
} # guardar el archivo y salir**.**
Habiltamos los puertos 67 UDP en el Servidor y 68 UDP en el cliente

Con la ayuda de una pequeña demostración miramos como asigna la dirección de nuestro a nuestro cliente de una manera correcta y tambieno nos dirigimos al archivo para ver las IP que están siendo proporcionadas por nuestro servidor DHCP

cat /var/lib/dhcpd/dhcpd.leases

NTP (NETWORK TIME PROTOCOL)

Protocolo de Internet para sincronizar los relojes de los sistemas informáticos a través del enrutamiento de paquetes en redes con latencia variable. NTP utiliza UDP como su capa de transporte, usando el puerto 123. Está diseñado para resistir los efectos de la latencia variable.

CONFIGURACIÓN

#yum install –y ntp*

Habilitamos el puerto 123 en el cual trabaja NTP, accedemos al archivo y configuramos las siguientes líneas:

#gedit /etc/ntp.conf

server inocar.ntp.ec
server 127.127.1.0

En este momento iniciamos servicios:

#chkconfig ntpd on
#systemctl start ntpd.service
#systemctl restart ntpd.service

Con la ayuda de una pequeña demostración miramos como se sincronizan relojes de nuestro cliente con la de nuestro servidor, ejecutamos en el cliente el siguiente comando:

#ntpd – u 10.3.0.30 (ip de nuestro servidor)

TFTP (TRIVIAL FILE TRANSFER PROTOCOL)

Protocolo de transferencia muy simple semejante a una versión básica de FTP. TFTP a menudo se utiliza para transferir pequeños archivos entre ordenadores en una red, como cuando un terminal X Window o cualquier otro cliente ligero arranca desde un servidor de red.

CONIGURACIÓN

#yum install –y tftp*

Accedemos al archivo de configuración de tftp y editamos la siguiente línea y salimos:

#gedit /etc/xinetd.d/tftp
server_args= -s -c /var/lib/tftpboot

En este momento restauramos servicios:
#service xinetd restart

Nos dirigimos a la siguiente dirección para verificar que exista el directorio tftpboot, en caso de no existir creo un directorio:

#cd /var/lib
#mkdir tftpboot
Damos los permisos de leer, escribir y ejecutar:

#chmod 777 tftpboot

Hacemos que se configuren para el siguiente reinicio:

#chkconfig tftp on
#chkconfig xinetd on

Para poder realizar una sesión tftp os dirigimos a la siguiente dirección:

#cd /var/lib/tftpboot
#touch prueba //creamos un archive
Para hacer tftp a un cliente:
#tftp [Dirección IP del cliente]
***tftp>* put prueba.txt** //nombre del archivo que deseo enviar
***tftp>* q** //para salir.

SAMBA

Es una implementación libre del protocolo de archivos compartidos de Microsoft Windows (antiguamente llamado SMB, renombrado recientemente a CIFS) para sistemas de tipo UNIX. De esta forma, es posible que computadoras con GNU/Linux, Mac OS X o Unix en general se vean como servidores o actúen como clientes en redes de Windows.

CONFIGURACIÓN :
Para empezar identificamos el nombre del grupo de trabajo de Windows en cmd digitamos:
#net config Workstation
En nuestro caso es WORKGROUP
Luego abrimos le archivo hosts
C:Windows/system32/drivers/etc/hots

Y agregamos nuestro servidor CentOS y cerramos:

10.3.0.30 servidor.centos.com centos

Luego en el servidor CentOS 7 instalamos los paquetes Samba:
#yum install samba-common samba-cliente Samba

luego editamos el archivo de configuración y agregar lo siguiente:

#vim /etc/samba/smb.conf

[global]
workgroup = WORKGROUP

server string = Samba Server %v
netbios name = centos
security = user
map to guest = bad user
dns proxy = no

[Anonymous]
path = /samba/anonymous
browsable =yes
writable = yes
guest ok = yes
read only = no

Luego nos dirigimos a la raíz y creamos nuestra carpeta compartida:

#mkdir –p /samba/anonymous

Restauramos los servicios de samba y de nmb.service

#systemctl enable smb.service
#systemctl start nmb.service
#systemctl status smb.service

Habilitamos el servicio en el corta fuegos para samba y lo recargamos:

#firewall-cmd --permanent --zone=public --add-service=samba
#firewall-cmd –reload

luego nos vamos a Windows para verificar que se haya creado la carpeta compartida ,en inicio ponemos \\centos luego de haber verificado que este la carpeta centos en Windows nos vamos al servidor CentOS 7 y damos permisos de acceso a todos los usuarios de Windows

#cd /samba
#chmod -R 0755 anonymous/
#chown -R nobody:nobody anonymous/
#ls -l anonymous/

para verificar que se haya ejecutado correctamente los comandos el resultado debe ser este:

drwxr-xr-x. 2 nobody nobody 4 dic 14 12:04 anonymous

Además tenemos que permitir que el selinux para la configuración de samba de la siguiente manera:
#chcon -t samba_share_t anonymous /
luego comprobamos creando archivos y verificando que estén en la carpeta de Windows y también en el servidor CentOS 7.

VNC (VIRTUAL NETWORK COMPUTING)

Programa de software libre basado en una estructura cliente-servidor el cual permite tomar el control del ordenador servidor remotamente a través de un ordenador cliente. También llamado software

de escritorio remoto. VNC no impone restricciones en el sistema operativo del ordenador servidor con respecto al del cliente: es posible compartir la pantalla de una máquina con cualquier sistema operativo que soporte VNC conectándose desde otro ordenador o dispositivo que disponga de un cliente VNC portado.

CONFIGURACION:

#yum install vnc-server

Agregamos el servicio y los puertos de VNC usa el puerto 5900, el primer usuario usaría 5901

#firewall-cmd - -permanent - -zone=public - -add-service=vnc-server
#iptables -P INPUT ACCEPT
#iptables -A INPUT -i enps03 -p tcp --dport 5901 -j ACCEPT
#iptables -A INPUT -i enps03 -p udp --dport 5901 -j ACCEPT
#service iptables save
#firewall-cmd –reload

Copiamos el archivo de configuración genérico.

#cp /lib/systemd/system/vncserver@.service /etc/systemd/system/vncserver@:1.service

Y lo editamos, tenemos que colocar el nombre del usuario al cual nos vamos a conectar (en este caso del servidor) y el display que es un índice de todos los usuarios con los que nos vamos a conectar remotamente.

#nano /etc/systemd/system/vncserver@:1.service

Nos conectamos a la cuenta del usuario que colocamos en el archivo de configuración de VNC.

#su nombre_usuario
#vncserver
Y nos pedirá una contraseña para la conexión VNC.

Habilitamos el servidor vnc.

#systemctl daemon-reload
#systemctl enable vncserver@:1.service
#systemctl start vncserver@:1.service
En caso de que al iniciar el servicio dé un error, hay que eliminar la carpeta /tmp/.X11-unix

#rm -rf /tmp/.X11-unix

Instalamos el cliente VNC también.

#yum install tigervnc

Luego de la instalación aparecerá la aplicación en la sección de Internet, ahí colocamos la IP del servidor, seguido de ":" y el socket, que para el caso del primer usuario sería 5901.

10.3.0.30:5901