Práctica 1 – Configuración de servicios de red

(0.75 puntos)

## Realización práctica por los alumnos del grupo FR2(turno 1):

**David Suarez González**

**Daniel Monjas Miguélez**

1. Compruebe las direcciones IP que tienen asignadas las diferentes interfaces de red de su equipo mediante el comando *ifconfig*, ¿cómo se llaman dichas interfaces? ¿qué direcciones de red tienen definidas?

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | **Nombre Interfaz** | **PC\_1** | **PC\_2** | **PC\_3** | | **enp0s3** | 10.0.2.15 | 10.0.2.15 | 10.0.2.15 | | **enp0s9** | 33.1.1.2 | 33.1.1.3 | 33.1.2.2 | | **enp0s10** | 192.168.1.1 | 192.168.1.2 | 192.168.1.3 |   (En la terminal de PC\_1-Ubuntu20.04)  administrador@pc1:~$ ifconfig  enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255  inet6 fe80::3d00:5458:c3ab:e588 prefixlen 64 scopeid 0x20<link>  ether 08:00:27:fd:98:cc txqueuelen 1000 (Ethernet)  RX packets 31451 bytes 44402901 (44.4 MB)  RX errors 0 dropped 0 overruns 0 frame 0  TX packets 2613 bytes 210564 (210.5 KB)  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  enp0s9: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  inet 33.1.1.2 netmask 255.255.255.0 broadcast 33.1.1.255  inet6 fe80::a00:27ff:fe4d:8a84 prefixlen 64 scopeid 0x20<link>  ether 08:00:27:4d:8a:84 txqueuelen 1000 (Ethernet)  RX packets 0 bytes 0 (0.0 B)  RX errors 0 dropped 0 overruns 0 frame 0  TX packets 76 bytes 8101 (8.1 KB)  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  enp0s10: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  inet 192.168.1.1 netmask 255.255.0.0 broadcast 192.168.255.255  inet6 fe80::a00:27ff:fe21:6685 prefixlen 64 scopeid 0x20<link>  ether 08:00:27:21:66:85 txqueuelen 1000 (Ethernet)  RX packets 0 bytes 0 (0.0 B)  RX errors 0 dropped 0 overruns 0 frame 0  TX packets 77 bytes 8182 (8.1 KB)  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  inet 127.0.0.1 netmask 255.0.0.0  inet6 ::1 prefixlen 128 scopeid 0x10<host>  loop txqueuelen 1000 (Bucle local)  RX packets 268 bytes 23507 (23.5 KB)  RX errors 0 dropped 0 overruns 0 frame 0  TX packets 268 bytes 23507 (23.5 KB)  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  Al usar el comando ifconfig accedemos a la configuración de interfaces de red del equipo. Cuando lo ejecutamos sin argumentos simplemente muestra las interfaces activas, que son las siguientes:   * enp0s3: Tiene dirección de red IP 10.0.2.15 y es la interfaz de red NAT (Acceso a internet). * enp0s9: Tiene dirección de red IP 33.1.1.2 y es la interfaz de red Datos. * enp0s10: Tiene dirección de red IP 192.168.1.1 y es la interfaz de red de Gestión. * lo: Tiene dirección de red IP 127.0.0.1 y es la interfaz de red lo (Loop Back) |

1. Compruebe que existe conectividad con otro equipo del laboratorio, mediante la utilidad *ping*. ¿Es posible hacer ping desde el PC\_1 al PC\_3 por la red 33.1.1.0/24? ¿Y por la red 192.168.1.0/16? Justifique su respuesta. A partir de ahora la primera de las redes la llamaremos de *datos* mientras que la segunda será la de *gestión*.

|  |
| --- |
| administrador@pc1:~$ traceroute 33.1.2.2  traceroute to 33.1.2.2 (33.1.2.2), 30 hops max, 60 byte packets  1 \_gateway (10.0.2.2) 0.214 ms 0.167 ms 0.147 ms  2 192.168.0.1 (192.168.0.1) 3.375 ms 3.346 ms 3.150 ms  3 10.195.52.1 (10.195.52.1) 24.178 ms 24.168 ms 23.934 ms  4 10.80.4.125 (10.80.4.125) 19.884 ms 19.835 ms 19.742 ms  5 \* \* \*  6 \* \* \*  7 \* \* \*  ----------------  30 \* \* \*  Como se puede ver no hay conexión entre PC1 y PC3 por la red 33.1.1.0/24. Todos los paquetes que se transmiten se pierden. No podemos alcanzar dicha IP, pues en el encaminamiento que siguen los paquetes a través de los routers, llega un momento en que se detienen en un router específico, que podemos averiguar con “traceroute”, (en este caso, el router con dirección IP 10.80.4.125 ):    Los paquetes transmitidos por ping son recibidos como muestra la primera captura. Además la segunda captura muestra con wireshark como los recibe PC3. |

1. Cree una cuenta de usuario en su equipo, habilite el servicio *telnet* y compruebe con algún compañero que dicho servicio es accesible.

|  |
| --- |
| Usaré la cuenta administrador que ya existe en los entornos virtuales, e intentaré conectar dos de estos entornos mediante telnet.    La opción disable = no implica que el servicio está habilitado y funciona. Ahora para comprobar que el servicio funciona se accede desde PC3 a PC1. Para que el cambio en el archivo de configuración sea efectivo se ha de realizar la orden “service xinetd restart”. Comprobamos que ahora el servicio telnet está activo y en escucha en el PC\_1 (como servidor):  root@pc1:/home/administrador# service xinetd restart  root@pc1:/home/administrador# netstat -ltp  Conexiones activas de Internet (solo servidores)  Proto Recib Enviad Dirección local Dirección remota Estado PID/Program name  tcp 0 0 localhost:domain 0.0.0.0:\* ESCUCHAR 557/systemd-resolve  tcp 0 0 0.0.0.0:ssh 0.0.0.0:\* ESCUCHAR 732/sshd: /usr/sbin  tcp 0 0 pc1:ipp 0.0.0.0:\* ESCUCHAR 589/cupsd  tcp6 0 0 [::]:http [::]:\* ESCUCHAR 774/apache2  tcp6 0 0 [::]:ftp [::]:\* ESCUCHAR 715/vsftpd  tcp6 0 0 [::]:ssh [::]:\* ESCUCHAR 732/sshd: /usr/sbin  tcp6 0 0 [::]:telnet [::]:\* ESCUCHAR 17158/xinetd  tcp6 0 0 ip6-localhost:ipp [::]:\* ESCUCHAR 589/cupsd    Como se puede ver el servicio funciona correctamente (además se observa que se hizo un login incorrecto, ya que me equivoqué al poner la contraseña). También cree un directorio (mkdir prueba) y un archivo (archivo\_prueba.txt).  administrador@pc1:~$ mkdir prueba  administrador@pc1:~$ cd prueba/  administrador@pc1:~/prueba$ touch archivo\_prueba.txt  administrador@pc1:~/prueba$ ls  archivo\_prueba.txt  Y al comprobarlo en PC\_1 vemos que si que está el archivo y el directorios creados. |

1. Configure el servicio telnet para que:
   1. Sólo sea accesible desde la dirección IP de su compañero.

|  |
| --- |
| Este es el archivo de configuración de telnet, donde la opción only\_from = <dir\_ip> indica la dirección ip de los host remotos para los que el servicio estará disponible.        Como se puede observar en la primera captura PC3 si puede acceder a PC1 por medio del servicio telnet. Sin embargo, PC2 no puede acceder a PC1 por medio de telnet pues su dirección IP no concuerda con la del archivo de configuración del servicio telnet de PC1. |

* 1. cSe registren en el fichero /var/log/telnet.loglos intentos de acceso con y sin éxito al servicio telnet, indicando la dirección IP del equipo que intenta el acceso.

|  |
| --- |
| En primer lugar añadimos la opción log\_on\_success += HOST, para que en caso de un login exitoso se guarde la dirección IP de quien la ha realizado. La opción log\_on\_failure += HOST no es necesaria añadirla pues ya estaba en el archivo de configuración. Finalmente la opción log\_type = FILE *“/var/log/telnet.log”* hace que la información obtenida de un login se mande al archivo /var/log/telnet y en caso de que este archivo no exista se crea.    Ahora mostramos la información que se almacena en el archivo /var/log/telnet tras un login existoso y uno rechazado, pues la dirección IP no coincide con la que el archivo de configuración de telnet permite.    Como se puede ver en el archivo se ha almacenado un intento fallido de conectar por medio del servicio telnet a PC1 desde PC2, y un intento exitoso de hacer login por medio de telnet desde PC3 a PC1. |

1. Habilite el servicio *ftp* en su equipo (de la “*a”* a la “*c”*).

|  |
| --- |
| Para habilitar el servicio ftp utilizo el comando “service vsftpd start”.  a) Para ello en el archivo de configuración del servicio vsftpd, es decir, en el archivo vsftpd.conf incluimos la siguiente línea.  listen = no  Esta opción se encarg de que vsftpd **no** funcione en modo standalone. Si la opción estuviese a yes si lo haría.    b) Para impedir el acceso de la cuenta anonymous lo único que tenemos que hacer es añadir al archivo de configuración vsftpd.conf la siguiente línea,  anonymous\_enable=NO  Al poner esta opción a no se prohiben los login anónimos.    c) Para permitir el acceso al servicio a las cuentas locales se incluye en el archivo de configuración vsftpd.conf la siguiente línea,  local\_enable=YES  Al poner esta opción a YES se permiten el acceso a las cuentas locales al servicio.    En los tres apartados no ha sido necesario realizar cambios pues el archivo vsftpd.conf estaba correctamente configurado de forma predeterminada. Aún así se adjuntan capturas demostrando las opciones a poner. |

1. Pida a un compañero que pruebe el servicio ftp. ¿Qué comandos utilizó para ello?

|  |
| --- |
| En este ejercicio uso ftp para conectar PC1 y PC3, para ello en primer lugar uso el comando *service vsftpd start* en ambos PC para iniciar el servicio. En este caso la conexión será desde PC3 a PC1.    Desde el PC3 una vez hecho el login se utilizan los comandos,  cd Descargas, para acceder al directorio Descargas de PC1.  ls para ver que archivos/directorios se encuentran en el momento en el directorio Descargas.  get <file> para descargar en PC3 un archivo de PC1, en este caso se hace get prueba\_2.  exit para finalizar la conexión entre PC1 y PC3.  Finalmente al hacer un cat sobre el archivo descargado se ve su contenido, donde pone que pertenece al PC1. El texto lo puso yo como prueba. |

1. Configure el servicio ftp para que:
   1. Únicamente pueda ser utilizando a través de la cuenta de usuario que hemos creado en nuestro equipo.

|  |
| --- |
| En el archivo vsftpd.conf incluimos las siguientes líneas,  userlist\_deny=NO, esta opción se examino sólo si userlist\_enables está activado. Si esta opción se configura a NO, entonces se denegará el login a los usuarios que no estén especificados en el archivo userlist\_file.  userlist\_enable=YES, si esta opción está activada, vsftpd cargará una lista de nombres de usuarios desde el archivo dado por userlist\_file.  userlist\_file=/etc/vsftpd.user\_list, este será el archivo que contendrá el nombre de los usuarios a los que se podrán acceder remotamente (obviamente dicho usuario deberá acceder en el sistema). Este archivo contendrá una sola línea en la que estará escrito “daniel” que es un usuario creado previamente en PC1. Ahora veamos que funciona accediendo desde PC2.      Vemos que en el segundo login pone servicio denegado ya que el nombre de usuario que se ha introducido no está en el archivo vsftpd.user\_list. |

* 1. Acepte la subida de ficheros al servidor ftp.

|  |
| --- |
| Para esto utilizamos la opción write\_enable=YES, la cual controla si cualquier comando de FTP tiene permitido hacer cambios en el sistema de archivos o no.    Ahora comprobaremos desde PC1 que el archivo que hemos enviado está,  administrador@pc1:~$ ls Descargas Documentos enviado\_a\_pc1 Escritorio Imágenes Música Plantillas prueba Público Vídeos  administrador@pc1:~$ cat enviado\_a\_pc1  Mandando mensaje desde pc\_2 a pc\_1  administrador@pc1:~$  Luego como se puede ver el mensaje se ha enviado desde PC2 a PC1. |

1. Habilite el servicio *http* en su equipo. Abra un navegador web y pruebe a visitar la página de inicio desde su equipo (http://localhost o http://127.0.0.1). Además, realice los siguentes cambios:
   1. Modifique el contenido de la página de inicio, y compruebe con la ayuda de su compañero que la dirección de su servidor es accesible.

|  |
| --- |
| La siguiente captura sería la página [http://localhost](http://localhost/) de PC1.    Modifico el archivo *“/var/www/html/index.html”* para que en lugar de It works! Ponga funciona correctamente y la fecha y hora en las que modifico el archivo. (Como no se html simplemente cambié una línea sobre lo que ya estaba programado).    Aquí sería la página de inicio modificada. Y ahora se adjunta captura accediendo a esta página desde PC2. |

* 1. Modifique el puerto de escucha del servidor de modo que el acceso a la página de inicio se haga mediante la dirección: *http://localhost:8080*.

|  |
| --- |
| Para ello modifico el archivo ports.conf de forma que el puerto de escucha sea 8080 (para ello la primera línea será Listen 8080). Además modifico el archivo “/etc/apache2/sites-enabled/000-default.conf” y cambio el puerto a 8080.  ports.conf  000-default.conf  Ahora se adjuntan capturas demostrando que se tiene que acceder a la página de inicio del local host por medio de [http://localhost:8080](http://localhost:8080/).  (la s en https se añade sola) |

* 1. Cree una página de acceso restringido (es decir, que requiera usuario y contraseña antes de mostrarla) en [*http://localhost/restringida/*](http://localhost/restringida/). Utilice como credenciales de acceso el usuario *admin* y la contraseña *1234*.

|  |
| --- |
| En primer lugar elimino los cambios hechos anteriormente para que el puerto de escucha por defecto vuelva a ser 80.  Ahora creo la capeta restringida dentro del directorio “/var/www/html/”. Este será el directorio cuyo acceso sea restringido dentro de nuestro localhost. A continuación en el directorio *“/usr/local/”* uso el comando,  sudo htpasswd -c passwords admin  Que me creará el archvio que guardará encriptada la identificación para el usuario admin. Inmediatamente se pide que se introduzca contraseña y que se vuelva a introducir para confirmar, en este caso se introduce como contraseña 1234.  Ahora me voy al directorio “/var/www/html/restringida/”, que es el directorio al cual le queremos imponer usuario y contraseña, y en el creo un archivo .htaccess. Este archivo tendrá la siguiente forma,    La opción AuthName “Name” establece el nombre del área de autorización de un directorio. Este nombre se da al usuario para que sepa que contraseña y usuario mandar.  La opción AuthType Basic selecciona el tipo de identificación de usuario para un directorio. El modo Basic establece autorización básica.  La opción AuthUserFile “file” establece el nombre del archivo textual que contiene la lista de usuarios y contraseñas para la identificación de accesos. En “file” se escribe no solo el nombre sino también la localización de este archivo.  La opción require valid-user prueba cuando un usuario es autorizado de acuerdo con unas ciertas restricciones. Valid-user permite que todos los usuarios válidos pueden acceder al recurso, en este caso, todos los usuarios que realizen un login válido de acuerdo con el archivo de autorización de la opción anterior.  Por último en el archivo de configuración de apache, apache2.conf añadimos las siguientes líneas,  </Directory /var/www/html/restringida/>  AllowOverride AuthConfig  </Directory>  Al incluir estas líneas se hace que sobre el directorio /var/www/html/restringida se aplique la opción AllowOverride AuthConfig, es decir, en ese directorio se permite las directivas de autorización de acceso sobreponiendose al archivo de configuración.  Finalmente se adjunta captura de que funciona correctamente. |