

Servicios Básicos

- [DHCP](#)
- [BIND9 DNS](#)
- [Transmisión de ficheros](#)
- [Tomcat para Ubuntu](#)

DHCP

Para probar estas aplicaciones siempre puedes crear dos máquinas virtuales con [virtualbox](#), yo lo hice con una máquina virtual y el pc que las aloja. Siendo cliente del servidor **dhcp** el pc que aloja las máquinas virtuales. Recuerda que para que te funcione tendrás que apagar el servidor **dhcp** del router.

Para instalar el servidor **DHCP** desde la línea de comandos en un sistema bajo **Debian**, teclea lo siguiente:

- **apt-get install dhcp3-server**
- Siempre se puede buscar otros servidores o actualizaciones con **apt-cache search dhcp**.

La ruta del fichero que necesitamos para configurar dhcp es:

- **/etc/dhcp/dhcpd.conf**

Para que la configuración de dos rangos diferentes de IP la configuración sería la siguiente:

```
subnet 192.168.5.0 netmask 255.255.255.0 {  
    range 192.168.5.100 192.168.5.120;  
    range 192.168.5.200 192.168.5.210;  
}
```

Ahora definiremos los dos rangos anteriores con un tiempo de leasing de 5 min. (300 seg.) y un tiempo máximo de 15 min. (900 seg.)

```
subnet 192.168.5.0 netmask 255.255.255.0 {  
  
    range 192.168.5.100 192.168.5.120;  
    range 192.168.5.200 192.168.5.210;  
    default-lease-time 300;  
    max-lease-time 900;  
  
}
```

Para reiniciar el servidor DHCP tenemos varias opciones:

- **/etc/init.d/isc-dhcp-server restart**
- **service isc-dhcp-server restart**

Para forzar que un cliente pida una nueva dirección IP

- **dhclient -r**
- **dhclient eth0**

Ficheros de log para encontrar información relacionada con el servidor DHCP

- **cat /var/log/syslog | grep dhcp**
- **/var/lib/dhcp/dhcp.leases**

Seguidamente veremos un fichero de configuración completo y como podemos reservar direcciones ip para clientes específicos:


[Akg-dhcp.png](#)
Akg-dhcp.png or type unknown

BIND9 DNS

Para instalar el servidor de nombres Bind9 debemos ejecutar el siguiente comando:

- **apt-get install bind9**

Cuando queramos añadir un dominio al servidor de nombres (DNS) bind9, deberemos añadir las siguientes líneas al fichero **/etc/bind/named.conf.local** para decirle que lo gestionaremos con bind.

 A continuación tendremos que crear el archivo especificado en la zona que acabamos de crear, en este archivo se incluirán las opciones del dominio.



Cosas a tener en cuenta:

- En la configuración del fichero **/etc/bind/named.conf.options** la opción *forwarders* sirve para configurar nuestro servicio como pasarela, esto quiere decir que nuestro servidor no hace la resolución de nombres, sino que se la encarga a otro DNS.
- Para configurar el cliente para que coja de manera predeterminada el servidor de nombres es bien fácil:

En el archivo **/etc/resolv.conf del cliente**

domain dominio.com/net/es/etc

search dominio.com

nameserver 192.168.0.2 #Dirección del DNS

- Para comprobar en el cliente, que el servidor funciona y nos devuelve la dirección **www.dominio.com**

host www.dominio.com

Hemos de obtener

www.domino.com has address 192.168.0.10

Transmisión de ficheros

Diferencias entre ftp y ssh

Los dos son servicios de transmisión de ficheros. El segundo hace la transmisión de forma cifrada. Además, el servicio **openssh-server** permite establecer una terminal remoto seguro con el que se pueden enviar comandos del cliente al servidor, viajando en una comunicación cifrada.

Programas ftp

Tenemos muchas alternativas cuando se trata de ftp, por ejemplo atftpd, ftpd-ssl, gftp, **proftp**, **vsftp**, ... Yo personalmente me quedo con los dos últimos. Si quieres buscar servidores ftp prueba con este comando:

- **apt-cache search "ftp server"**

Configuraciones habituales ftp

- En muchos de estos servicios el único usuario habilitado es el anónimo. Si queremos cambiar esto deberemos habilitarlo.
- También podemos encontrar negado el permiso para enviar ficheros del cliente al servidor.
- Si en el archivo de configuración nos encontramos con la siguiente línea, querrá decir que podemos encerrar a los usuarios en sus propias carpetas. **chroot_local_user=YES**

Problema del ftp

El protocolo de comunicación FTP tiene el problema de la seguridad, es decir, el password del usuario viaja por la red i puede ser capturado por un programa de captura de paquetes como [Wireshark](#).

Por eso cuando queramos seguridad en nuestros envíos nunca utilizaremos el protocolo ftp

FTP activo o FTP pasivo

- **En el modo Activo** es el servidor quién comienza la comunicación desde el puerto 20 hacia un puerto cualquiera del cliente.
- **En el modo Pasivo**, no se abre la conexión con el puerto 20 del servidor, sino que es el cliente quién inicia la conexión y lo hace directamente en el puerto 21 del servidor.
- El modo pasivo es el que se utiliza más habitualmente, ya que nos evita abrir un puerto adicional y esto lo hace más adecuado para pasar los cortafuegos, que únicamente habilitan el puerto 21.

FTP Anónimo y las cuotas.

El servidor **ftp anónimo** nos es útil cuando queremos dar un servicio de descargas con un volumen muy grande de usuarios y no queremos dar de alta a cada usuario por separado. El usuario anónimo actúa como si fuese un usuario normal pero con menos privilegios: Normalmente solo tiene acceso a la descarga y tiene una estructura visible del servidor muy reducida.

Las cuotas permiten al administrador dar un espacio máximo a el usuario. De esta manera, pueden dar servicio ftp a diversos usuarios y asegurarse que nunca un usuario individualmente puede consumir todo el espacio disponible del disco duro del servidor.

Comandos ssh y sftp

Para descargar un archivo desde el servidor:

- **scp usuario@192.168.0.10:/home/usuario/fichero.txt**

Para subir un archivo al servidor

- **scp fichero.txt usuario@192.168.0.10:/home/usuario**

El protocolo TFTP

El protocolo **tftp** (Trivial File Transfer Protocol) es un protocolo heredado del **ftp** pero mucho más simple, con funciones recortadas. La idea es tener un protocolo más sencillo para que dispositivos con poco espacio pueden tener un programa **ftp**, se pueden conectar a un servidor y enviar o recibir ficheros de forma simple.

El caso donde es más utilizado es para arrancar desde una red un pc. La BIOS tiene un espacio limitado y necesita de un protocolo sencillo con tal de descargar el S.O. del servidor y arrancar.

Tomcat para Ubuntu

Parte 1 Instalación y Usuario

Antes de comenzar ha instalar tomcat necesitamos tener instalado Java y además haber creado un tipo de usuario específico.

```
sudo useradd -m -d /opt/tomcat -U -s /bin/false tomcat
sudo apt update
sudo apt install default-jdk
java -version
```

Después iremos a la carpeta /tmp para descargar los archivos de instalación de tomcat y procederemos a su descompresión en una carpeta específica.

```
cd /tmp
wget https://dlcdn.apache.org/tomcat/tomcat-10/v10.0.20/bin/apache-tomcat-10.0.20.tar.gz
sudo tar xzvf apache-tomcat-10*tar.gz -C /opt/tomcat --strip-components=1

#Damos los permisos necesarios
sudo chown -R tomcat:tomcat /opt/tomcat/
sudo chmod -R u+x /opt/tomcat/bin
```

Parte 2 Configuración de usuarios

Vamos a configurar los usuarios Manager y HostManager.

Primero añadimos los password necesarios para cada usuario en

```
/opt/tomcat/conf/tomcat-users.xml
```

```
<role rolename="manager-gui" />
<user username="manager" password="manager_password" roles="manager-gui" />

<role rolename="admin-gui" />
<user username="admin" password="admin_password" roles="manager-gui,admin-gui" />
```

Por defecto tomcat esta configurado con restricción de acceso, para cambiar esto, primero para el usuario manager, vamos al archivo:


```
/opt/tomcat/webapps/manager/META-INF/context.xml
```

Y comentamos desde Valve hasta el final:

```
<Context antiResourceLocking="false" privileged="true" >
  <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
    sameSiteCookies="strict" />
  <!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /> -->
  <Manager
    sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|Number|string)|org\.apache\.catalin
a\.filters\.Csr>
</Context>
```

Después para el usuario host-manager:

```
/opt/tomcat/webapps/host-manager/META-INF/context.xml
```

Misma operación en ambos.

Parte 3. Crear un servicio

Localizamos java

```
sudo update-java-alternatives -l
```

Creamos el archivo tomcat.service

```
/etc/systemd/system/tomcat.service
```

```
[Unit]
Description=Tomcat
After=network.target

[Service]
Type=forking

User=tomcat
Group=tomcat

Environment="JAVA_HOME=/usr/lib/jvm/java-1.11.0-openjdk-amd64"
```

```
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom"
Environment="CATALINA_BASE=/opt/tomcat"
Environment="CATALINA_HOME=/opt/tomcat"
Environment="CATALINA_PID=/opt/tomcat/temp/tomcat.pid"
Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"

ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh

RestartSec=10
Restart=always

[Install]
WantedBy=multi-user.target
```

Reiniciamos el systema de servicios

```
systemctl daemon-reload
systemctl start tomcat
systemctl status tomcat

sudo systemctl enable tomcat
ufw allow 8080
http://your_server_ip:8080
```