



2021
2022



Despliegue de Aplicaciones Web

Tema 0: Introducción a los servicios en red



UNIÓN EUROPEA
Fondo Social Europeo
El FSE invierte en tu futuro

Programa financiado por el Ministerio de Educación y Formación Profesional y
cofinanciado por el Fondo Social Europeo

Miguel Ángel García García
IES Doñana



Contenido

1	Introducción	3
2	Direccionamiento	3
2.1	Protocolo IPv4	3
2.1.1	Dirección IPv4	3
2.1.2	Máscara de red	4
2.1.3	Puerta de enlace	5
3	Servicios de red, protocolos y puertos	5
4	Servicio DHCP	7
4.1	Instalación de servidores de configuración de parámetros de red	7
4.2	Preparación del servicio para asignar configuraciones básicas de red	7
4.3	Configuración de asignaciones estáticas	9
5	Preparación del entorno de trabajo	10

1 Introducción

La informática es una ciencia en la que se producen cambios con vertiginosa rapidez. Esto se puede observar tanto en la evolución del hardware en los últimos años que, además de aumentar sus prestaciones a gran velocidad, se ha miniaturizado, como en el software que cada día nos permite realizar tareas que antes consumían una gran cantidad de recursos intelectuales, automatizando prácticamente todos y cada uno de los aspectos de la vida cotidiana.

En la actualidad, las aplicaciones web están en gran auge, tanto a nivel web como en aplicaciones móviles. Por ello, la importancia de este módulo para entender como implementar una aplicación web desde sus cimientos hasta el despliegue completo de la misma. **El objetivo de cualquier programador web es la publicación de la aplicación desarrollada.**





Para comenzar, cualquier tema en cualquier aspecto de la vida necesita de unos cimientos, y esa máxima se ha aplicado a los contenidos del presente módulo. Por ello, se comenzará por explicar los **servicios de red que intervienen en el despliegue de una aplicación web**. Posteriormente se seguirá por las **aplicaciones que son vitales para desplegar la aplicación** (por ejemplo: FTP) y continuaremos con una breve explicación de las **arquitecturas web que existen en el mercado**. Llegado al ecuador del módulo, es momento de explicar cómo **administrar un servidor web** y posteriormente, cómo **desplegar una aplicación web en un servidor de aplicaciones**. Finalmente, terminaremos con un tema no menos importante, como es la **documentación y el sistema de control de versiones**.

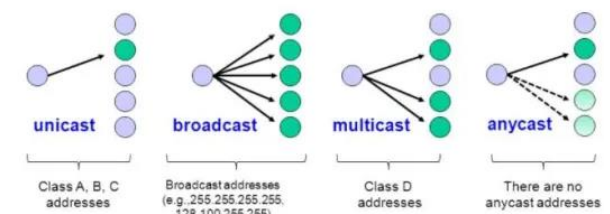
Una aplicación web necesita de servicios de red para poder funcionar de forma correcta y coherente. Estos servicios son el **servicio DHCP**, **DNS** y el **servicio de directorio LDAP**. Estos servicios los veremos más adelante, pues antes de poder usar cualquier servicio, deberemos tener configurada correctamente la red del equipo.

2 Direccionamiento

El direccionamiento es una función propia de los protocolos de la capa de red/Internet que permite la identificación y transmisión de información entre nodos, tanto si están en la misma red como si están en redes diferentes. Es importante dominar este concepto, así como los términos asociados, pues la configuración y el mantenimiento de un servidor parten de esta base.

Existen los siguientes tipos de direccionamiento:

-  **Unicast:** Identifica una interfaz de un único nodo.
-  **Multicast:** Identifica un grupo de interfaces que, generalmente, pertenecen a diferentes nodos. Cuando un paquete se envía a una dirección multicast, va dirigido a todos los nodos que pertenecen a la misma.
-  **Broadcast:** Identifica al grupo formado por todas las interfaces de los nodos conectados a la red, permitiendo envíos de información a todos ellos con un único mensaje simultáneamente y sin necesidad de emitir el mismo mensaje nodo por nodo.
-  **Anycast:** Identifica un grupo de interfaces que, generalmente, pertenecen a diferentes nodos. Cuando un paquete se envía a una dirección anycast, va dirigido al nodo miembro del grupo anycast que esté físicamente más cerca del remitente.

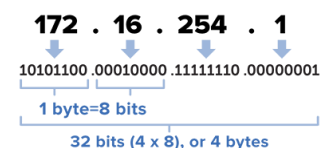


2.1 Protocolo IPv4

IPv4 (Protocolo de Internet versión 4) es la cuarta revisión del Protocolo de Internet (IP) que se usa para identificar dispositivos en una red a través de un sistema de direccionamiento. Es el protocolo más utilizado para conectar dispositivos. Todos los servicios que se describirán a lo largo de este módulo soportan IPv4.

2.1.1 Dirección IPv4

Una dirección IPv4 es un conjunto de 4 octetos (32 bits) separados por puntos (.), que ofrecen un espacio de direccionamiento de 2^{32} posibles valores. Cada octeto puede tomar valores comprendidos entre 0 y 255, aunque también tienen su correspondiente representación en formato binario:

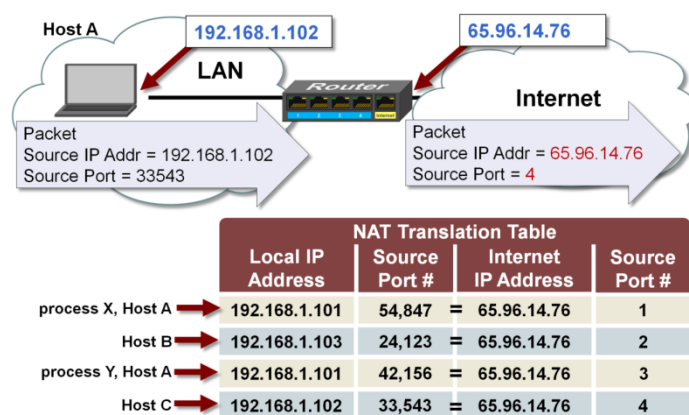


Estas direcciones se asignan a las interfaces de red de los diferentes nodos y se puede incluso asignar más de una dirección a la misma interfaz. Gracias a esta asignación, los nodos pueden identificarse y se permite la comunicación entre ellos.

Cada **dirección** se utiliza en el nivel de red del modelo TCP/IP y podrá ser **fija** (establecida manualmente) o **dinámica** (obtenida a partir de un servidor DHCP). De los tipos de direccionamiento descritos anteriormente, IPv4 únicamente puede emplear unicast, multicast o broadcast. Además, cada dirección puede clasificarse según su ámbito en:

- ✚ **Privada:** Es la que tiene un nodo dentro de una red de área local y únicamente es visible desde esa misma red. Para acceder a un servicio ofrecido por un nodo desde otro nodo que resida en la misma red local, bastará con que este último conozca su dirección IP privada. Un ejemplo típico de uso de estas direcciones para acceder a un servicio es el acceso a la página web de configuración de un router doméstico, que generalmente se realiza a través de la URL <http://192.168.1.1>. En realidad, cualquier dispositivo (smartphone, smartTV, tablet, ordenador portátil...) que se conecta a una red doméstica o corporativa (independientemente de que el tipo de conexión sea cableada o inalámbrica) dispondrá de una dirección privada.
- ✚ **Pública:** Tiene conexión directa a Internet y es la dirección que realmente es visible desde dicha red. Ejemplos de este tipo de dirección son la que toma un equipo que se conecta directamente a Internet a través de un módem, el router en la interfaz que conecta con el operador, un smartphone conectado por 4G-5G...

Para que un nodo que reside en una red local A pueda acceder a un servidor que está en otra red local B perteneciente a una organización diferente y para la que no hay forma de encaminar internamente los paquetes, será necesario conocer la dirección IP pública tras la que se encuentra dicho nodo. Generalmente, en esta comunicación se hace uso de **NAT**.



IMPORTANTE: La dirección IP pública de un dispositivo no puede conocerse desde la configuración del mismo equipo, pues no depende de ninguno de sus parámetros de conexión. Para conocer la IP pública, puedes hacer uso de páginas web como <https://www.cual-es-mi-ip.net>

2.1.2 Máscara de red

Dentro de cada dirección IPv4 hay una porción de los bits de orden superior que representa la dirección de la red en la que se encuentra (todos los nodos de la red tienen los mismos valores en esa porción de bits). El número de bits que se toman para identificar la parte de red viene determinado por la **máscara de red/subred** que, al igual que la dirección IP, es un conjunto de 4 octetos (32 bits) separados por puntos. No obstante, las máscaras no pueden tomar todo el rango de valores posible, y se permite únicamente un conjunto contiguo de bits a 1 que representa la porción de red y a continuación otro conjunto contiguo de bits a 0 que representa la porción de hosts. Las máscaras también pueden representarse en notación CIDR mediante /N, donde N es el número de bits activos (1) de la máscara.



Originalmente, las direcciones formaban parte de **redes classfull** en las que no se declaraban las máscaras de red de forma expresa. Eso era porque las direcciones IP estaban clasificadas en clases que tenían asociada una máscara de red implícita, por lo que no era necesario definirla (los dispositivos de red analizaban los primeros bits de la dirección IP para saber a la clase a la que pertenecía y, por tanto, la máscara correspondiente). Sin embargo, hoy en día las direcciones forman parte de **redes classless** y las direcciones IP deben definirse junto con su máscara de red/subred.

El proceso mediante el cual se incrementa el número de bits a 1 de la máscara por encima de los valores predeterminados se conoce como **subnetting**, y permite dividir la red en redes más pequeñas que albergan un número menor de hosts. El proceso inverso que reduce el número de bits a 1 de la máscara por debajo de los valores predeterminados se conoce como **supernetting**, y permite unificar varias redes en otras que pueden albergar más hosts.

Gracias a VLSM (Variable Length Subnet Mask), es posible dividir una red en subredes aplicando una máscara y continuar dividiendo esas subredes de forma recursiva en redes más pequeñas aumentando de nuevo el número de bits de la porción de red.

2.1.3 Puerta de enlace

Cuando un nodo quiere comunicarse con otro, comprueba si tanto él como el nodo destino están en la misma red. Esto lo hace obteniendo ambas redes mediante operaciones AND entre las IP y la máscara del nodo origen, y finalmente realizando la comparación bit a bit de ambas redes.

En caso de que ambas redes sean iguales, las comunicaciones se llevan a cabo sin problemas gracias a la existencia del protocolo de resolución de direcciones ARP. Mediante este protocolo, el nodo origen se ocupa de obtener la dirección MAC asociada a la dirección IP destino mediante el envío de paquetes ARP request a la dirección broadcast y a la posterior recepción del paquete ARP reply.

Sin embargo, si el recurso al que se quiere acceder no está en la misma red, se hace necesario usar algún elemento que, siendo accesible, reenvíe los paquetes IP que no coinciden con ninguna ruta de la tabla de enrutamiento al destino. Generalmente, ese elemento es un router que dispone de una dirección IP privada en el rango válido de nuestra red local y que se considera como puerta de enlace para el equipo por ser una dirección alcanzable (de la misma forma que se podía alcanzar un equipo en la misma red) y que permite acceder a otras redes.

A menudo, el término "**puerta de enlace**" aparece como "**puerta de enlace predeterminada**", "**puerta de acceso**" o "**Gateway**".

3 Servicios de red, protocolos y puertos

Se denomina "**servicio de red**", al conjunto de recursos y procesos que buscan satisfacer las necesidades que los clientes demandan a través de la red. Tanto si son accedidos por nodos de la misma red como si los clientes son nodos ajenos a esta.

La implantación de estos servicios reporta un elevado número de ventajas (permiten un uso más eficiente de los recursos, mejoran la productividad en los procesos, facilitan tareas de configuración...).

Generalmente, los servicios de red se implantan en **arquitecturas cliente-servidor**, lo cual favorece su consumo por parte de los clientes. Los nodos que alojan y ofrecen estos servicios se denominan **servidores**. Sin embargo, en algunos casos el servicio se encuentra distribuido entre varios nodos que cooperan entre sí (arquitecturas P2P), descentralizándose de esta manera el papel del servidor, que pasa a residir en este caso en todos y cada uno de los pares.

Los servicios de red pueden residir en equipos con sistemas operativos diseñados especialmente para comportarse como servidores (SOLARIS, Ubuntu Server, Microsoft Windows Server...), aunque también pueden alojarse sobre equipos con sistemas operativos cliente (Ubuntu Desktop, MacOS, Microsoft Windows 10,...)

Algunos de estos sistemas operativos ya tienen instalado por defecto el servicio, el cual puede estar deshabilitado, u ofrecen la posibilidad de instalarlo de forma fácil desde sus propias fuentes. En otros casos, es necesario instalar el software necesario desde fuentes externas al equipo que, ocasionalmente, es de terceros. De todos modos, para que

un equipo pueda actuar como servidor es necesario que disponga del software y el hardware necesarios para prestar el servicio concreto, pues no todos los servicios precisan de los mismos requisitos.

Todo servicio de red necesita de una serie de protocolos que rijan cómo se ofrecerá el servicio.

En la siguiente tabla se muestra un listado de algunos de los servicios más comunes y los protocolos de la capa de aplicación del modelo TCP/IP asociados:

Servicio	Protocolo
Servicios de administración y configuración de sistemas	DHCP (Dynamic Host Configuration Protocol o Protocolo de configuración dinámica de Host). Es un protocolo de red que permite que los clientes de una red obtengan los parámetros de configuración IP automáticamente
	DNS (Domain Name System o Sistemas de nombres de dominio). Es un protocolo empleado en la traducción de nombres de equipos y recursos a direcciones IP y viceversa.
Servicios de publicación de información en la web	HTTP (Hypertext Transfer Protocol o Protocolo de transferencia de hipertexto). Es un protocolo de comunicación empleado para la transmisión de documentos hipermedia, como HTML.
Servicios de acceso remoto	Telnet (Telecommunication Network). Es el protocolo empleado para hacer conexiones remotas no cifradas y cuyo programa cliente toma el mismo nombre.
	SSH (Secure Shell o Shell segura). Es el protocolo empleado para hacer conexiones remotas cifradas y cuyo programa cliente toma el mismo nombre.
Servicios de transferencia de ficheros	FTP (File Transfer Protocol o Protocolo de transferencia de ficheros). Es el protocolo empleado para transferir ficheros a través de la red.
	TFTP (Trivial File Transfer Protocol o Protocolo simple de transferencia de ficheros). Es el protocolo empleado para transferir ficheros de pequeño tamaño a través de la red sin autenticar al usuario.
Servicios de impresión, comparación de archivos y sistemas de ficheros en red	NFS (Network File System o Sistema de archivos en red). Protocolo que permite que hosts remotos monten sistemas de ficheros sobre la red e interactúen con ellos como si estuvieran montados localmente. Incluido por defecto en la mayoría de los sistemas UNIX/LINUX.
	SMB (Server Message Block). Se emplea para interconectar equipos Microsoft Windows y compartir archivos e impresoras entre ellos. SAMBA es la implementación de código abierto del protocolo SMB.

La capa de transporte del modelo TCP/IP, se ocupa de identificar el proceso o servicio del nodo destinatario que recibirá los datos lo cual permite que múltiples servicios estén ejecutándose de forma simultánea en el equipo servidor. Esto se consigue gracias al **puerto**, que es un número que permite identificar el servicio destinatario del paquete recibido.

Aunque en cada protocolo se definen los puertos en los que por defecto el servicio debe escuchar peticiones, estos normalmente se pueden modificar. Para ello, se debe tomar precaución de no elegir ningún puerto que esté siendo utilizado por otro servicio, pues de lo contrario uno de los dos no se iniciará. Por último, que no menos importante, hay que asegurarse de que los clientes conocen el nuevo puerto de escucha, pues de lo contrario no podrán acceder al servicio.

A nivel de la capa Internet, algunos servicios trabajan únicamente con IPv4, aunque la mayor parte de ellos son compatibles tanto con IPv4 como con IPv6.

4 Servicio DHCP

En este apartado se explicarán los procedimientos para la instalación y configuración del servicio de concesión de configuraciones de red (DHCP). Concretamente, se realizará una configuración de ejemplo usando el sistema operativo Linux Ubuntu 20.04 y el software servidor ISC.

Nota: No se profundizará mucho en este tipo de servicio, ya que se sale de los objetivos del módulo

4.1 Instalación de servidores de configuración de parámetros de red

El primer paso sería actualizar los repositorios de Linux localizados en el fichero `/etc/apt/sources.list`.

```
$ sudo apt update
```

Una vez recuperadas las versiones más actuales, se realiza una actualización de los paquetes con las versiones más recientes:

```
$ sudo apt upgrade
```

Seguidamente, se va a usar el software servidor DHCPD (Dynamic Host Configuratio Proccol Daemon) para proporcionar la configuración de red de aquellos equipos que lo soliciten. Para eso, se va a utilizar el paquete `isc-dhcp-server`, cuya instalación se ejecuta de la siguiente manera:

```
$ sudo apt install isc-dhcp-server
```

IMPORTANTE: Para poder instalar el servicio, el equipo servidor debe tener una configuración de red válida.

4.2 Preparación del servicio para asignar configuraciones básicas de red

Para la preparación y gestión del servicio DHCP se deben conocer, principalmente, dos ficheros:

- `/etc/default/isc-dhcp-server`: donde se guarda el valor de las interfaces físicas del servidor donde se van a recibir las peticiones de configuración de red de los clientes.
- `/etc/dhcp/dhcpd.conf`: fichero que guarda los parámetros de configuración que se van a repartir a los clientes. Contienen valores como rango de IP, máscaras y puertas de enlace, entre otros.

Los mensajes que se mandan del cliente al servidor tendrán un **puerto** de destino **67** y los del servidor al cliente tendrán un **puerto** destino **68**. Ambos casos son tipos de paquetes UDP.

En primer lugar, se va a especificar el nombre del adaptador de red que va a estar escuchando las peticiones de los clientes. Para eso se debe modificar el fichero `/etc/default/isc-dhcp-server`, especificando en la línea `INTERFACES` el nombre de la interfaz.

Véase un ejemplo: la siguiente captura muestra el contenido del fichero `/etc/default/isc-dhcp-server` de un servidor. En este caso, en la última línea, se puede comprobar que el nombre de la interfaz por donde se van a recibir las peticiones sería `enp0s3`.

```
GNU nano 4.8 /etc/default/isc-dhcp-server
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
#DHCPDv4_CONF=/etc/dhcp/dhcpd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpd6.conf

# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPDv4_PID=/var/run/dhcpd.pid
#DHCPDv6_PID=/var/run/dhcpd6.pid

# Additional options to start dhcpd with.
# Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
# Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACESv4="enp0s3"
INTERFACESv6="enp0s3"
```


Una vez configurado el fichero `/etc/default/isc-dhcp-server`, habría que modificar el fichero `/etc/dhcp/dhcpd.conf`. Este archivo está compuesto por una serie de sentencias que pueden ser de dos tipos:

1. **Parámetros.** Permiten asignar un valor a una opción del servicio. Su sintaxis admite dos maneras:
 - a. **Nombre_parámetro:** solo se escribiría el nombre del parámetro. Si se especifica, significará que está activo.
 - b. **Nombre_parámetro valor/es:** donde es necesario especificar un valor al parámetro.
2. **Declaraciones.** Conjuntos de parámetros agrupados entre {}. Dentro de una declaración también se pueden especificar nuevas declaraciones. La sintaxis quedaría de la siguiente manera:

```
Declaración {  
    Parámetros;  
    Declaración{};  
}
```

Las declaraciones pueden ser de varios tipos, pero nosotros sólo nos vamos a centrar en los siguientes:

- a) **Subnet:** esta declaración indica la red bajo la cual van a recibir las configuraciones de red los clientes. De todos los parámetros, como mínimo, se debe encontrar el **parámetro range**, donde se especifica un rango de IP asignables a los equipos que la soliciten. La sintaxis sería:

```
subnet nombre_red netmask máscara_de_subred {  
    range ip_inicio ip_fin;  
    [parámetros];  
}
```

- b) **Shared-network:** es la declaración que se utiliza cuando en un mismo segmento de red físico se tienen varias subredes. En este caso, el adaptador físico del servidor DHCP debe tener una IP para cada subred a la que va a servir la configuración. Así que, en esta declaración, debe haber varias declaraciones subnet, una por cada subred que haya en el segmento, quedando, por tanto, la sintaxis de la siguiente forma:

```
shared-network nombre_identificativo {  
    [parámetros];  
    subnet ... {  
    }  
    ...  
    subnet ... {  
    }  
}
```

- c) **Host:** sirve para realizar reservas estáticas para algunos equipos, de manera que, cuando el servidor reciba una petición de una MAC específica, siempre se le asignará la misma IP. La sintaxis quedaría así:

```
host nombre_identificativo {  
    [parámetros];  
    hardware Ethernet dirección_MAC;  
    fixed-address dirección_IP;  
}
```

MUY IMPORTANTE: Para que las modificaciones realizadas en el fichero de configuración tengan efecto, se debe reiniciar el servicio con: `# service isc-dhcp-server restart`

Si se quisiera configurar el servicio de una manera básica, los parámetros que se deberían configurar para que los clientes tengan conectividad a la red serían los siguientes:

- **subnet nombre_red netmask máscara_de_subred { . . }:** la sentencia subnet englobaría los parámetros de configuración para la red indicada en el nombre_red y definida por la máscara_de_subred. Todo lo que esté definido en ese nivel tendrá valor para esa red. Entre otros, se pueden encontrar los siguientes parámetros:
 - **range {ip_inicio} {ip_fin}:** indica el rango de direcciones que va a repartir.
 - **Option routers {Gateway}:** indica el valor de la puerta de enlace para comunicarse con otras redes.

- **Option domain-name-servers {dns1, dns2, ..., dnsN}**: se especificarían las direcciones IP de los servidores DNS que van a resolver los nombres a los clientes. Si se ponen varios, tomaría el primer valor como primario y los siguientes como secundarios.
- **Default-lease-time {tiempo}**: indica el número de segundos por defecto que durará una concesión a un equipo si el cliente no solicita un tiempo de arrendamiento específico.
- **Max-lease-time {tiempo}**: señala la duración máxima en segundos que se asignará a una concesión. Este sería útil en el caso de que el cliente quisiera una asignación que durara un tiempo determinado, entonces no superaría este valor.

NOTA: Al colocar los parámetros default-lease-time y max-lease-time fuera de la declaración subnet se consigue que tengan validez para todas las declaraciones que se hicieran en el fichero. Si estuviesen dentro de la declaración subnet, sólo afectarían a los parámetros de la red indicada en ese subnet.

Por ejemplo, en la siguiente imagen, se puede observar cómo se especifica una serie de parámetros básicos para un servicio DHCP

En este caso, se definen los valores que se darían para la red 10.5.5.0/27. El rango de IP válidas sería desde el 10.5.5.26 hasta el 10.5.5.30, con un valor de puerta de enlace de 10.5.5.1 y los DNS 8.8.8.8 y 8.8.4.4. El parámetro default-lease-time será de 600 segundos (10 minutos) y, en el caso de que el cliente solicitara un mayor tiempo de concesión, el valor max-lease-time sería de 7200 segundos (120 minutos).

```
subnet 10.5.5.0 netmask 255.255.255.224 {  
    range 10.5.5.26 10.5.5.30;  
    option domain-name-servers 8.8.8.8, 8.8.4.4;  
    # option domain-name "internal.example.org";  
    option subnet-mask 255.255.255.224;  
    option routers 10.5.5.1;  
    option broadcast-address 10.5.5.31;  
    default-lease-time 600;  
    max-lease-time 7200;  
}
```

Al especificar los valores de tiempo dentro de la sentencia subnet, sólo se aplicarían a las configuraciones de la red 10.5.5.0.

Una vez que se ha configurado la interfaz y definido los parámetros básicos para ofrecer una configuración IP válida, se debe reiniciar el servicio con la sentencia:

```
$ sudo systemctl restart isc-dhcp-server
```

También se puede realizar una parada del servicio y después un inicio:

```
$ sudo systemctl stop isc-dhcp-server  
$ sudo systemctl start isc-dhcp-server
```

Para comprobar el estado del servicio, se puede ejecutar la sentencia:

```
$ sudo systemctl status isc-dhcp-server
```

4.3 Configuración de asignaciones estáticas

Asignación estática (también denominada reserva) es aquella en la que, a una dirección MAC, se le da una dirección IP determinada por un tiempo indefinido.

La **asignación estática** se especificaría en el fichero de configuración **/etc/dhcp/dhcpd.conf**. Para ello, se debe insertar una declaración que se incluiría al final del fichero con la sentencia host, donde se especificarían los siguientes valores:

- **Hardware Ethernet {MAC}**: donde {MAC} sería la dirección física del cliente que solicitará la configuración.
- **Fixed-address {IP}**: donde {IP} sería la dirección IP que se le asignaría a la dirección MAC especificada previamente.

```
host webserver {  
    hardware ethernet 0:0:c0:5d:bd:95;  
    fixed-address 10.5.5.2;  
}
```

5 Preparación del entorno de trabajo

A lo largo de las unidades se van a describir algunos de los protocolos ya mencionados. Asimismo, se plantearán escenarios de ejemplo y se propondrán soluciones a los mismos basadas en la instalación y configuración del servicio correspondiente sobre equipos virtualizados con sistemas operativos libres y propietarios, aunque nos centraremos principalmente en la distribución de Ubuntu Server LTS más reciente.

Para los clientes usaremos una distribución ligera con interfaz gráfica tipo linux.