MONDRAGON UNIBERTSITATEA
EMPRESA ZIENTZIEN FAKULTATEA
FACULTAD DE CIENCIAS EMPRESARIALES

SERVIDORES DHCP
DTO. INFORMATICA
Profesor: Ignacio Arenaza Nuño, José Luis Flores

Servidores DHCP

Curso 2003-2004

CAPÍTULO I

Introducción a DHCP

En los años 80, era habitual utilizar un protocolo muy sencillo llamado BOOTP que permitía que algunos sistemas (normalmente máquinas Unix corriendo /etc/bootpd) asignarán direcciones IP a sistemas tales como impresoras o servidores de terminales. El servidor utilizaba un sencillo fichero de texto para buscar la dirección MAC del "cliente" y le asignaba la dirección IP (y algún otro parámetro) según constará en dicho fichero. Actualmente este sistema se usa, por ejemplo, para algunos elementos de electrónica de red como *switches* o *hubs* y en ciertos modelos de impresoras con interfaz de red local.

El protocolo BOOTP utizaba una estructura de tramas muy sencilla y el tráfico generado era mínimo. Desgraciadamente, no es suficiente para la mayoría de los casos y en redes de tamaño medio, su eficacia es muy baja.

A principios de la década de los noventa la IETF (Internet Engineering Task Force) desarrolló el protocolo DHCP. Su objetivo principal era superar las limitaciones de BOOTP, ampliándolo y permitiendo que los administradores de redes se olvidaran, casi por completo, de la asignación de direcciones IP a las decenas y centenares de PC's y otras máquinas de su organización.

Las siglas DHCP significan Dynamic Host Configuration Protocol, en castellano Protocolo de Configuración Dinámica de Hosts. Este protocolo es usado para grandes redes, y este servicio actúa dandole informacion de la red a las estaciones de trabajo, tales como la dirección IP, máscara de subred, servidores DNS, puertas de enlace (gateway), etc. El parámetro más importante de la configuración asignado por DHCP es la dirección IP.

El DHCP permite la transmisión de la configuración de hosts sobre una red TCP/IP. DHCP es un protocolo que permite asignar direcciones IP dinámicas, de forma totalmente automática. Por ello no pierde las prestaciones de BOOTP, su predecesor, sino que las amplía permitiendo nuevas formas de asignación de direcciones y nuevas opciones para poder pasar a los clientes toda la información necesaria. DHCP es un protocolo implementado en los principales sistemas operativos así como otros dispositivos.

Al igual que otros protocolos similares, utiliza el paradigma cliente-servidor, para que los nodos clientes obtengan su configuración del nodo servidor.

1 DEFINICIONES

A continuación definiremos algunos términos usados:

Servidor DHCP: será la máquina que ejecutará el servicio DHCP.

Cliente DHCP: la máquina que requerirá del servicio ofertado por los servidores.

Ambito o subred: es el intervalo de direcciones de red que estarán disponibles para ser asignadas por los servidores a los clientes. Suelen definir subredes físicas por lo general, por eso se agrupan en ámbitos.

Concesión: cuando a un cliente se le ha asignado una dirección, se dice que se ha realizado una concesión y se indica hasta cuando es válida.

Red: grupo de estaciones interconectadas.

Subred: la subredes son utilizadas para dividir una red en redes más pequeñas.

Máscara de subred: es un número que se utiliza para obtener la dirección de red a partir de una dirección IP. La separación se realiza por medio de una operación lógica AND entre la dirección IP y la máscara de subred, por lo que la máscara de subred deberá tener puestos a 1 aquellos bits que corresponden a la dirección de red (incluyendo la parte de la subred) y a 0 aquellos que formen la dirección de host. Para el caso en que se usen bytes completos para dirección de red y de host.

Dirección de broadcast: se usa para identificar a todas las estaciones en todas las redes o en una red. La dirección de broadcast se forma poniendo en 1 todos los bits correspondientes a la porción de host de la dirección IP.

Dirección de multicast: Sirve para comunicarse con un grupo de estaciones que forman parte de un grupo específico.

2 Características:

A continuación se muestran las principales características de DHCP, que han hecho que sea un protocolo muy extendido, implementado en los principales sistemas operativos, routers y periféricos de red. Estas características serán detalladas en los siguientes apartados:

- Se asienta sobre el protocolo UDP (puerto 67 el servidor y 68 el cliente). Utilizando Broadcast (dirección IP de difusión 255.255.255.255), para la comunicación con el servidor. La responsabilidad de la confiabilidad de la comunicación recae sobre el cliente (mediante time-out y retransmisión en caso de error).

- Diseñado para ser compatible con BOOTP. Utiliza el mismo formato de mensaje.

- Con arquitectura cliente-servidor, permitiendo el uso de agentes de relé BOOTP verifica si el servidor y el cliente están en una red diferente.

- El servidor admite tres tipos de configuraciones de direcciones IP:

    1. **Estática**: Se configura en el servidor la dirección de red que se corresponde con la dirección LAN del cliente (equivalente a BOOTP).

    2. **Dinámica, por tiempo ilimitado**: Se indica un rango de direcciones que se asignan a cada cliente de carácter permanente, hasta que el cliente la libere.

    3. **Dinámica, arrendada:** Las direcciones se otorgan por un tiempo limitado. Un cliente debe renovar su dirección con el servidor DHCP que la ha otorgado para poder seguir utilizándola.

- Cada vez que arranca un cliente, debe volver a solicitar una dirección. El servidor debe recordar si el cliente ya tenía una dirección para reasignársela. Si el servidor no responde, el cliente podrá usar la dirección, mientras no caduque su tiempo de concesión.

La asignación de direcciones IP se configurará en un modo u otro, dependiendo de cada situación. Puede interesar un direccionamiento estático para clientes sin disco o por facilidades administrativas, pero controlando la asignación de cada dirección a cada cliente (es más cómodo para el administrador configurar un servidor, que cada cliente; interesa el direccionamiento estático para evitar que se conecten clientes no identificados o por otras razones, como la configuración de DNS).

El direccionamiento dinámico por tiempo ilimitado se utiliza cuando el número de clientes no varía demasiado, facilitando mucho la tarea del administrador.

El arrendamiento de direcciones se emplea para racionar las direcciones IP, minimizando el coste administrativo. En función de la frecuencia de inserciones o eliminaciones de clientes y de la cantidad de direcciones disponibles se concederá un mayor o menor tiempo de arrendamiento.

Hay una secuencia de mensajes definida a intercambiar para que el cliente adquiera una dirección IP. Se deben seguir los siguientes pasos:

1. El cliente manda mediante broadcast un mensaje DHCPDISCOVER. Este mensaje lo recibirá un servidor DHCP, si está en la misma subred física, o un agente relé que se encargará de enviar la petición al servidor en otra subred.

    ---------->
    solicitud de concesión

2. Cada servidor que reciba la petición puede responder con un mensaje DHCPOFFER, con la dirección disponible en el campo SU DIRECCIÓN IP. Esta dirección no está reservada en este momento, pero el servidor debe evitar, en la medida de lo posible, ofrecérsela a otros clientes. El servidor sabe si el mensaje a pasado por un agente relevo, si el campo DIRECCIÓN IP DEL ROUTER es distinto de cero. En éste caso pondrá el

bit del campo de FLAGS a uno, para que el agente utilice broadcast con el cliente.

<----------

Oferta de concesión

3. El cliente puede recibir uno o más mensajes DHCPOFFER de uno o más servidores DHCP. El cliente elegirá una configuración y enviará mediante broadcast un mensaje DHCPREQUEST, con la identificación del servidor (para indicar qué servidor se ha seleccionado) y la dirección ofrecida en el campo SU DIRECCIÓN IP (para indicar al servidor qué dirección se solicita).

---------->

Selección de concesión

4. Los servidores reciben el mensaje DHCPREQUEST, determinando si su oferta ha sido aceptada o rechazada. El servidor seleccionado envía un mensaje DHCPACK. Los parámetros de configuración enviados, no deben entrar en conflicto con los parámetros del mensaje DHCPOFFER.

<----------

Confirmación de selección

5. El cliente recibe el mensaje DHCPACK, con los parámetros de configuración. Puede comprobar si la dirección IP no está repetida, mediante ARP. Si está repetida, el cliente envía un mensaje DHCPDECLINE y espera 10 segundos antes de volver a iniciar el proceso. Del mismo modo, el cliente puede recibir un mensaje DHCPNAK, volviendo al principio.

6. El cliente puede decidir dejar la dirección asignada, enviando un mensaje DHCPRELEASE. Debe incluir la misma identificación, que la utilizada en la petición DHCPREQUEST.

CAPÍTULO II

Instalación del servidor DHCP

1 Procedimientos de instalación

Para instalar el servidor DHCP, tenemos como siempre dos procedimientos posibles:

- A partir del código fuente.
- Instalando el paquete ya compilado de la distribución (la opción más aconsejable).

1.1 Instalación a partir del código fuente

El código fuente de este servidor se obtiene a partir de la dirección oficial `http://www.isc.org/` y dentro de este sitio web llegaremos a la dirección `http://www.isc.org/products/DHCP/` o bien utilizando alguno de sus mirrors de la siguiente dirección `http://www.isc.org/ISC/MIRRORS.html`

Una vez obtenido el código fuente, la secuencia de pasos a seguir seria la siguiente:

```
# gzip -dc dhcp-3.0pl1.tar.gz | tar xvf -
dhcp-3.0pl2
dhcp-3.0pl2/doc
dhcp-3.0pl2/doc/IANA-arp-parameters
...
#
```

O bien mediante la utilidad tar de GNU:

```
# tar xvzf dhcp-3.0pl1.tar.gz
dhcp-3.0pl2
dhcp-3.0pl2/doc
dhcp-3.0pl2/doc/IANA-arp-parameters
...
#
```

Una vez descomprimido este fichero nos creará un directorio y acto seguido entraremos en él para crear el fichero Makefile, ejecutando la instrucción **configure**:

```
# cd dhcp-3.0pl2
# ./configure
```

```
System Type: linux-2.2
make[1]: Entering directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2'
Making links in common
make[2]: Entering directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2/common'
...
make[1]: Leaving directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2'
yubarta:/usr/local/src/dhcp-3.0pl2#
```

Y ahora sólo nos restaría compilarlo:

```
# make
make[1]: Entering directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2'
Making all in common
make[2]: Entering directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2/common'
cc -g  -I/usr/local/src/dhcp-3.0pl2  -I/usr/local/src/dhcp-
3.0pl2/includes -DLIN
UX_MAJOR=2 -DLINUX_MINOR=4     -c -o raw.o raw.c
...
make[2]: Leaving directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2/dhcpctl'
make[1]: Leaving directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2'
#
```

Y posteriormente instalarlo:

```
# make install
make[1]: Entering directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2'
Installing in common
make[2]: Entering directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2/common'
for dir in /usr/man/man5; do \
...
install -c  omshell.man1  \
                      /usr/man/man1/omshell.1
make[2]: Leaving directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2/dhcpctl'
make[1]: Leaving directory `/usr/local/src/dhcp-
3.0pl2/work.linux-2.2'

#
```

Después de este proceso nos creara un directorio llamado work.linux-2.2 (en nuestro caso, ya que estamos compilando en un sistema con un núcleo 2.2 de linux) donde tendremos los binarios que hemos obtenido.

Si lo deseásemos tenerlo como servicio seria necesario crear el script de arranque de sistema (de tipo BSD o System V, dependiendo de nuestro sistema) adecuado y ubicarlo en el directorio /etc/init.d (o /etc/rc.d/init.d según la distribución Linux usada) como los demás servicios.

1.2 Instalación del paquete compilado

Para el caso de la distribución Debian GNU/Linux 3.0, sólo tendríamos que realizar lo siguiente:

```
# apt-get install dhcp3-server
Reading Package Lists... Done
Building Dependency Tree... Done
...
#
```

2 Requerimientos

Para que el servicio de DHCP opere de forma correcta es necesario que el kernel del servidor DHCP tenga compilado el soporte de **Packet Filter** y **Socket Filter**. Asimismo, es necesario que aparezca el flag de **MULTICAST** entre los flags de la interfaz de red en la que escuchará el servidor DHCP.

Para saber si tenemos activado dicho flag, podemos usar la orden **/sbin/ifconfig** que nos indica las características y valores configurados para una interfaz de red. Si suponemos que la interfaz de red en cuestion es **eth0**, deberemos ejecutar:

```
# /sbin/ifconfig eth0
eth0 Link encap:Ethernet  HWaddr 00:50:DA:CF:49:FF
     inet addr:192.168.1.34 Bcast:192.168.1.255
Mask:255.255.255.0
     UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
     RX packets:1825306 errors:0 dropped:0 overruns:0 frame:0
     TX packets:1136903 errors:0 dropped:0 overruns:0 carrier:0
     collisions:0 txqueuelen:100
     RX bytes:2639395 (25.17 MiB)  TX bytes:351712409 (335.4
MiB)
     Interrupt:5 Base address:0xe800
```

Como vemos en nuestro caso, aparece el flag MULTICAST, con lo cual no tendremos problemas para hacer funcionar el servidor DHCP.

3 Arranque y detención del servidor DHCP

Para iniciar el servidor DHCP ejecutaremos lo siguiente (suponiendo una distribucion Debian GNU/Linux):

```
#/etc/init.d/dhcp start
Starting DHCP server: dhcp.
#
```

Para detener el servidor DHCP ejecutaremos lo siguiente:

```
#/etc/init.d/dhcp stop
Stopping DHCP server: dhcp.
#
```

Para reiniciar el servidor DHCP ejecutaremos lo siguiente:

```
#/etc/init.d/dhcp restart
Stopping DHCP server: dhcp.
Starting DHCP server: dhcp.
#
```

Es importante recalcar el hecho de que el servidor DHCP escuchará las peticiones por defecto en la interfaz de red **eth0,** salvo que cambiemos esto en el fichero /etc/default/dhcp3-server (de nuevo en el caso de una distribución Debian GNU/Linux 3.0):

```
INTERFACES="eth0 eth1"
```

CAPÍTULO III

Configuración del servidor DHCP

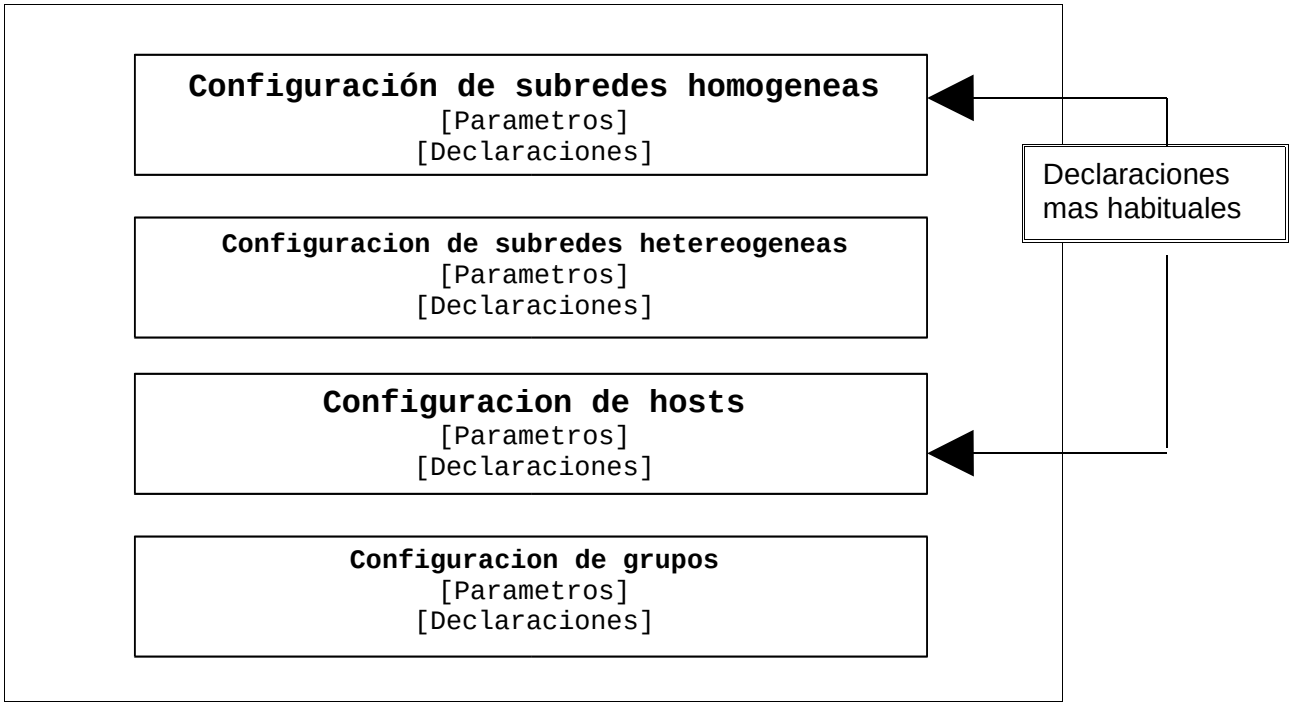
1 Configuración básica

La configuración básica del servidor está centrada en el fichero dhcpd.conf (ubicado en /etc/dhcp3/dhdpd.conf en el caso de Debian GNU/Linux 3.0) el esquema de cómo se configura es el siguiente:

```
                    Parametros COMUNES
[ Estas cabeceras se pueden aplicar de forma individual en cada
                          subred ]
```

Debemos tener en cuenta que el mismo parámetro declarado en una de las configuraciones subsiguientes sobreescribe el valor del parametro global.

Es decir, si como parametro global tenemos que el parametro que indica el nombre del dominio **`domain-name`** es **non-existent.org** y despues declaramos una subred que tiene delegado el subdominio **non-space.non-existent.org** podemos re-declarar el parametro **`domain-name`** con el nuevo dominio: **non-space.non-existent.org.**



Estas dos partes que hemos diferenciado de forma lógica, están formadas por **parámetros y declaraciones** por lo que antes de empezar a configurar subredes/grupos/hosts y definir opciones globales y locales describiremos el conjunto de parámetros más comunes utilizadas en la configuración:

Parámetro	Descripción	Tipo de datos
default-lease-time	Caducidad por defecto de la concesión (en segundos).	Numérico
domain-name	Nombre de dominio DNS para la subred, host o grupo especificado.	Texto
domain-name-servers	Lista de los servidores DNS para la subred, host o grupo especificado.	Lista de direcciones IP
fixed-address	Direcciones estáticas para asignar a un host (Soporta múltiples redes).	Lista de direcciones IP
group	Comienzo de la declaración de un grupo.	-
hardware	Tipo de hardware del interfaz de red.	Texto Alfanumérico
host	Comienzo de una declaración de host.	-
host-name	Nombre a asignar al host solicitante.	Texto
max-lease-time	Duracion máxima de la concesion (en segundos).	Numérico
netbios-name-servers	Lista de los servidores de nombres NetBIOS (Servidores WINS).	Lista de direcciones IP
range	Rango de direcciones IP a asignar en la subred especificada.	Direcciones IP inicial y final.
routers	Lista de routers (puertas de enlace) a utilizar.	Lista de direcciones IP
subnet	Comienzo de una declaracion de subred.	-
subnet-mask	Máscara de subred de una declaración de subred, host o grupo.	Máscara de red

Algunos de los parametros del servidor DHCP empiezan con la palabra reservada **options** y otros no. Esto es debido a que los primeros son específicos del protocolo DHCP y los segundos, además de este aspecto, están relacionados con el funcionamiento del servidor.

Ahora lo que nos restaría sería ver cuáles son las declaraciones posibles que podemos realizar:

- **Declaración de subredes.** Normalmente este será el tipo de declaración que realizaremos.

- **Declaración de hosts.** Este tipo de declaración nos permite asignar direcciones IP y parametro concretos a hosts individuales.

- **Declaración de grupos de hosts.** Que inicialmente no tienen porque pertenecer a la misma subred.

Una vez vistas estas opciones si deseasemos configurar el servidor para proporcionar direcciones IP en una subred concreta, el esqueleto básico a escribir en el fichero dhcpd.conf sería el siguiente:

```
subnet [Direccion Subred] netmask [Mascara de red] {

       range [Direccion IP Inicial] [Direccion IP Final];
     [ range [Direccion IP Inicial] [Direccion IP Final]; ]
     .
     .
     .
     [ range [Direccion IP Inicial] [Direccion IP Final]; ]
}
```

Podemos indicar excepciones si declaramos un rango cuya direccion inicial coincide con la final.

o bien si queremos especificar un rango y un conjunto de direcciones IP individuales, únicamente tendríamos que escribir la lista de direcciones IP separadas por “;” .

Un ejemplo de este esquema tan simple sería el siguiente:

- Subred : 192.168.1.0
- Mascara de red : 255.255.255.0

Queremos, dentro de la subred anterior, entregar direcciones IP por medio de DHCP, en el rango 192.168.1.2 a 192.168.1.20 (y reservar el resto para gestionarlas aparte). El fichero resultante seria:

```
subnet 192.168.1.0 netmask 255.255.255.0 {
     range 192.168.1.2 192.168.1.20;
}
```

Pero lo normal es especificar más parámetros necesarios en la respuesta del servidor DHCP como pueden ser:

- El nombre del dominio.
- Los servidores de DNS para ese dominio.
- La puerta de enlace por defecto
- Los servidores WINS (en un entorno Windows)
- etc.

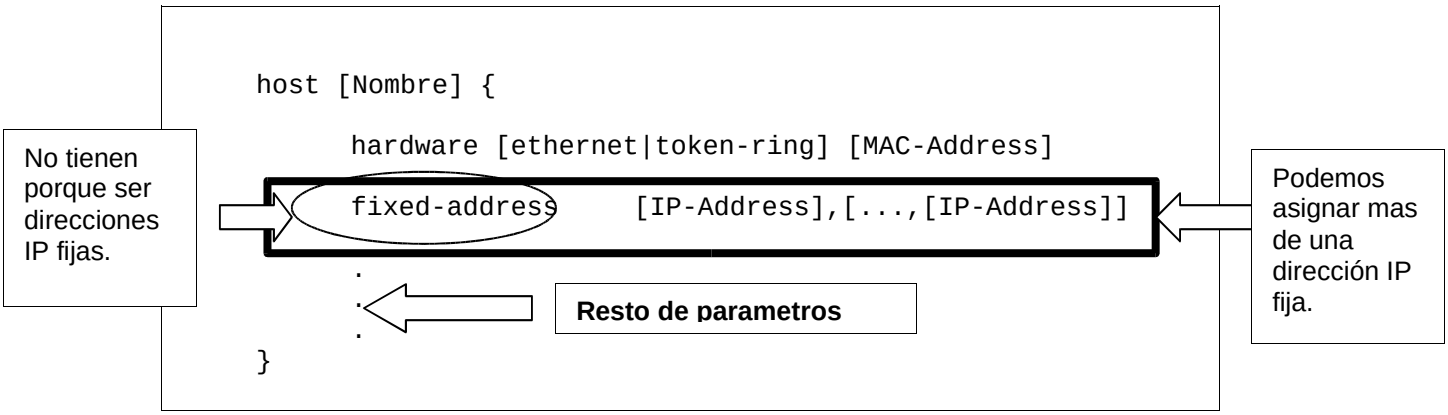
En este caso tenemos que hacer uso de los parametros que nos permiten asociar opciones a una declaracion de subred, como veremos más adelante

2 Configuración avanzada

2.1 Declaración de host.

La declaracion de host se realiza cuando deseamos asignar a un host concreto. Por host concreto entendemos que posee un nombre particular o una dirección MAC (de la tarjeta de red) particular. Recordemos que no lo podemos identificar por su dirección IP ya que es precisamente lo que estamos tratando de asignar.

Entonces la sintaxis general mínima de configuración de un host individual es la siguiente:



Ademas debemos tener en cuenta que el servidor DHCP también puede actuar como servidor BOOTP auxiliar, indicando el nombre del fichero a cargar mediante el parametro **filename** y el nombre del servidor BOOTP mediante el parametro **next-server** .

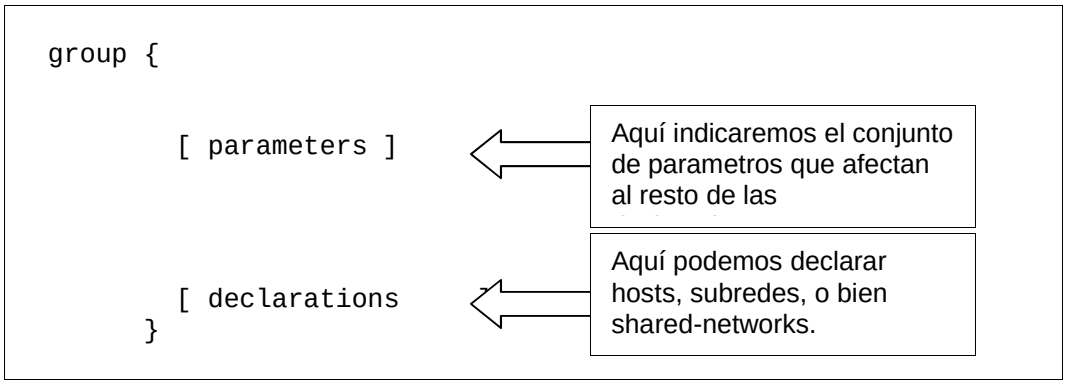
Un ejemplo de configuracion de host puede ser el siguiente:

```
host Master {
    hardware ethernet 00:00:44:34:34:54;
    fixed-address 192.168.1.2, 192.168.1.3, 192.168.1.4;
}
```

2.2 Declaracion de un grupo.

Utilizaremos una declaración de grupo cuando nos interese aplicar un mismo conjunto de parámetros a un conjunto de declaraciones. Habitualmente se utiliza para un conjunto de subredes o bien un conjunto de hosts que requieren de arranque el protocolo BOOTP.

La sintaxis general mínima de configuración de este tipo de declaración es excesivamente genérica como se puede comprobar:



Para poder entender mejor esto realizaremos un ejemplo con las siguientes caracteristicas:

- Poseemos cuatro subredes: 172.17.10.0/24, 172.17.20.0/24, 172.17.30.0/24 y 172.17.40.0/24 .
- Y por otra parte disponemos de cuatro hosts que deseamos que tengan direcciones IP fijas.

```
group {

   domain-name "eteo.mondragon.edu";
   domain-name-servers 172.17.50.1, 172.17.50.2;

   subnet 172.17.10.0 netmask 255.255.255.0 {
         range 172.17.10.1 172.17.10.254;
   }
   subnet 172.17.20.0 netmask 255.255.255.0 {
         range 172.17.20.1 172.17.20.254;
   }
   subnet 172.17.30.0 netmask 255.255.255.0 {
         range 172.17.30.1 172.17.30.254;
   }
   subnet 172.17.40.0 netmask 255.255.255.0 {
         range 172.17.40.1 172.17.40.254;
   }

   host smtp {
         hardware ethernet 00:12:23:34:34:45;
         fixed-address     172.17.50.10;
   }
   host pop3 {
         hardware ethernet 00:12:23:44:44:45;
         fixed-address     172.17.50.20;
   }
   host www {
         hardware ethernet 33:12:23:34:34:45;
         fixed-address     172.17.50.30;
   }
   host ftp {
         hardware ethernet 00:12:23:34:34:45;
         fixed-address     172.17.50.40;
```

```
        }
}
```

2.3 Declaracion de subred

Como hemos mencionado anteriormente el esqueleto presentado era muy simple, quizás demasiado simple para nuestros intereses.

Nos interesa trabajar con un esqueleto un poco más potente que maneje información en dos niveles. Aunque como se puede observar esto siempre dependerá de la instalación de red del sitio.

Vamos a indicar por tanto un esqueleto más real y más complejo de un servidor DHCP:

Parámetros Generales

Opciones especificas del protocolo DHCP

```
default-lease-time    [Tiempo en segundos];
max-lease-time        [Tiempo en segundos];

option domain-name            [Nombre del dominio];
option domain-name-servers    [IPs de los servidores de DNS];
option netbios-name-servers   [IPs de los servidores NBT];

subnet [Direccion Subred] netmask [Máscara de red] {
 option domain-name           [Nombre del dominio ];
 option domain-name-servers   [IPs de los servidores de DNS];
 option netbios-name-servers  [IPs de los servidores NetBIOS];

 range [Direccion IP Inicial] [Direccion IP Final];
 [ range [Direccion IP Inicial] [Direccion IP Final]; ]
      .
      .
      .
 [ range [Direccion IP Inicial] [Direccion IP Final]; ]

}
```

Opciones especificas del servidor y el protocolo DHCP

Parámetros especificos para la subred

Podemos indicar excepciones si declaramos un rango cuya direccion inicial coincide con la final.

3 Integración de DHCP con el DNS.

3.1 Configuración básica del servidor DHCP.

El servidor DHCP de ISC tiene la capacidad de integrarse con aquellos servidores que soportan el protocolo DNS con la extensión de DNS Dinámico (DDNS - RFC 2136), y añadir o borrar los registros de DNS en función de la creación o expiración de las concesiones DHCP.

Existen al menos dos métodos diferentes de actualización de DNS dinámico, siendo ambos incompatibles entre sí, y hay un tercero en desarrollo que será el futuro estándar oficial.

Los dos métodos de actualización disponibles en este momento se llaman:

- ad-hoc DNS update mode
- interim DNS update mode

Existe un tercer modo de actualización (llamado **none**) que indica que no queremos realizar ningún tipo de actualización en absoluto, y será el que debamos especificar **obligatoriamente** en caso de querer usar el servidor DHCP de forma independiente al servidor DNS.

Para indicar el método de actualización deseado tenemos que usar el parámetros global **ddns-update-style**:

```
ddns-update-style none;
ddns-update-style ad-hoc;
ddns-update-style interim;
```

Se recomienda no usar el método de actualización **ad-hoc** ya que está obsoleto a partir de las versiones 3.0 del servidor de DCHP de ISC **y no funciona**. Por tanto, el método recomendado es el método **interim**.

Además de indicar el método de actualización del DNS, debemos indicar en que dominio queremos que se integren dichas actualizaciones. Para ello debemos usar la opción **ddns-domainname**:

```
ddns-domainname "dhcp.midominio.com"
```

Lo anterior presupone que existe un dominio llamado "dhcp.midominio.com" configurado en el servidor DNS y que dicho dominio acepta actualizaciones dinámicas.

3.2 Configuración del Servidor DNS.

Para que el servidor DNS acepte las actualizaciones que el servidor DHCP le propone, hace falta configurar éste adecuadamente. Si vamos a realizar las actualizaciones en el dominio "dhcp.midominio.com" lo más adecuado (de hecho, es **altamente recomendable**) es crear una zona especialmente para la ocasión y configurar dicha zona para actualizaciones dinámicas:

```
zone "dhcp.midominio.com" {
        type master;
        file "db.dhcp.midominio.com";
        allow-update { 192.168.1.200; };
};
```

Siendo 192.168.1.200 la dirección IP del servidor DHCP, que es a quien permitimos que realice actualizaciones al DNS de la zona.

De igual forma, el servidor DHCP llevará a cabo actualizaciones dinámicas de las direcciones concedidas en la zona de resolución inversa (registros de tipo PTR), con lo cual deberemos configurar una zona de resolución inversa adecuada para esos cambios:

```
zone "1.168.192.in-addr.arpa" {
       type master;
       file "db.1.168.192.in-addr.arpa";
       allow-update { 192.168.1.200 };
};
```

3.3 Seguridad en las actualizaciones

La configuración anterior tiene un problema: es muy fácil suplantar la dirección IP del servidor DHCP y por tanto cambiar los datos del DNS sin tener autorización para ello.

La solución a este problema es usar actualizaciones seguras, que utilizan criptografía para firmar digitalmente las transacción de actualización del DNS, haciendo así imposibles en la práctica las suplantaciones. El tipo de actualizaciones seguras que vamos a ver aquí se llaman firmas TSIG (Trusted SIGnatures).

Para ello lo primero que tenemos que hacer es generar una clave criptográfica que nos permita llevar a cabo la firma digital. Es necesario guardar esta clave de forma que nadie la pueda leer, ya que sino podrían falsificar las firmas digitales y realizar actualizaciones del DNS no autorizadas. En este caso, la misma clave se usa tanto en el servidor DNS como en el servidor DHCP. Es lo que se llama criptografía de clave secreta, de clave compartida o de cifrado simétrico.

Para generar dicha clave tenemos que usar la orden **dnssec-keygen**[1] que viene con el servidor DNS de ISC. La forma de usarla es:

```
dnssec-keygen -a HMAC-MD5 -b 128 -n USER
CLAVE_ACTUALIZACION_DHCP
```

donde:

- la opción **-a** indica el tipo de algoritmo criptográfico usado para crear la clave. En nuestro caso debemos usar HMAC-MD5 obligatoriamente para crear firmas TSIG.

[1] Esta orden es válida para la versión 9.x del servidor de DNS de ISC; en caso de usar la versión 8.x del servidor de DNS de ISC, debemos utilizar la orden **dnskeygen**. La sintaxis de esta orden es diferente a la de **dnssec-keygen**, por lo que en este caso debemos usar:

```
# dnskeygen -H 128 -u -c -n CLAVE_ACTUALIZACION_DHCP
```

- -b 128 indica el tamaño de la clave en bits. En nuestro caso hemos indicado un tamaño de 128 bits, que es más que suficiente al tratarse de cifrado simétrico y además es el valor recomendado en el RFC 2136.
- -n USER indica que se trata de una clave de tipo USUARIO (en oposición a las claves de tipo ZONE o HOST, usadas para firmar registros de zona o de host respectivamente).
- CLAVE_ACTUALIZACION_DHCP es el nombre de la clave que vamos a generar. Podemos utilizar el nombre que más nos convenga.

La orden anterior crea dos ficheros:

```
KCLAVE_ACTUALIZACION_DHCP.+157+00000.key
KCLAVE_ACTUALIZACION_DHCP.+157+00000.private
```

que contienen la clave pública y privada generadas respectivamente (los numeros que aparecen en el nombre del fichero hacen referencia a las características intrinsecas de la clave y no son de interés en nuestro caso). En nuestro caso, al tratarse de una clave HMAC-MD5 ambos ficheros contienen el mismo valor.

Es importante proteger adecuadamente dichos ficheros para que nadie los pueda leer salvo root. De lo contario cualquiera puede realizar actualizaciones de DNS no autorizadas.

Así que lo que vamos a hacer es mover dichas claves al directorio **/etc/bind**, donde están todos los ficheros de configuración del DNS, y hacer que sólo root pueda leer su contenido:

```
# mv KCLAVE_ACTUALIZACION_DHCP* /etc/bind
# chown root.root KCLAVE_ACTUALIZACION_DHCP*
# chmod 600 KCLAVE_ACTUALIZACION_DHCP*
```

3.4 Configuración del servidor DNS

Una vez creadas las claves, debemos configurar BIND para que sólo acepte actualizaciones en las zonas anteriores si vienen firmadas con la clave que acabamos de crear. Para ello es necesario añadir las siguientes líneas al fichero de configuración de BIND **/etc/bind/named.conf**:

```
key CLAVE_ACTUALIZACION_DHCP {
         algorithm HMAC-MD5.SIG-ALG.REG.INT;
         secret pRP5FapFoJ95JEL06sv4PQ==;
};

zone "dhcp.midominio.com" {
        type master;
        file "db.dhcp.midominio.com";
        allow-update { KEY CLAVE_ACTUALIZACION_DHCP; };
```

```
};

zone "1.168.192.in-addr.arpa" {
       type master;
       file "db.1.168.192.in-addr.arpa";
       allow-update { KEY CLAVE_ACTUALIZACION_DHCP; };
};
```

El valor del campo **secret** de la opción **key** tiene que ser el que aparece en la línea **Key:** del fichero KCLAVE_ACTUALIZACION_DHCP.+157+00000.private:

```
# cat /etc/bind/ KCLAVE_ACTUALIZACION_DHCP.+157+00000.private
Private-key-format: v1.2
Algorithm: 157 (HMAC)
Key: pRP5FapFoJ95JEL06sv4PQ==
```

Por tanto, es necesario también proteger el fichero **/etc/bind/named.conf** para que sólo root pueda consultar su contenido, ya que hemos incluido una copia de la clave secreta en él.

3.5 Configuración del servidor DHCP

De igual forma, debemos configurar el servidor DHCP para que las actualizaciones que lleve a cabo en el servidor DNS vayan firmadas. Aprovecharemos además para indicarle cual es el servidor primario en el que debemos llevar a cabo las actualizaciones. Añadiremos para ellos los siguientes valores en la zona de configuración global del servidor DHCP:

```
...
key CLAVE_ACTUALIZACION_DHCP {
         algorithm HMAC-MD5.SIG-ALG.REG.INT;
         secret pRP5FapFoJ95JEL06sv4PQ==;
};

zone dhcp.midominio.com. {
         primary 192.168.1.254;
         key CLAVE_ACTUALIZACION_DHCP;
}

zone 1.168.192.in-addr.arpa. {
         primary 192.168.1.254;
         key CLAVE_ACTUALIZACION_DHCP;
}
...
```

Veamos en detalle qué son cada uno de los elementos que hemos añadido. En primer lugar hemos creado un bloque idéntico al del fichero named.conf donde declaramos la clave que vamos a usar para las actualizaciones del DNS.

```
key CLAVE_ACTUALIZACION_DHCP {
```

```
        algorithm HMAC-MD5.SIG-ALG.REG.INT;
        secret pRP5FapFoJ95JEL06sv4PQ==;
};
```

Los valores son idénticos al caso anterior: indicamos el tipo de clave con la palabra reservada **algorithm** (que en nuestro caso siempre sera HMAC-MD5.SIG-ALG.REG.INT) e indicamos el valor de la clave con la palabra reservada **key** (de nuevo el valor a indicar aquí es el que aparece en la línea Key: del fichero KCLAVE_ACTUALIZACION_DHCP.+157+00000.private).

Por tanto, es necesario también proteger el fichero **/etc/dhcp/dhcpd.conf** para que sólo root pueda consultar su contenido, ya que hemos incluido una copia de la clave secreta en él.

A continuación le indicamos cuales son las zonas del DNS donde se deben realizar los cambios dinámicos cada vez que el servidor DHCP otorge una concesión nueva o una concesión expire (y tenga que borrar los datos asociados a esta concesión en el DNS).

Para ello le indicamos cuales son las dos zonas a usar (una para la resolución directa y otra para la resolución inversa), así como los datos necesarios para contactar con el servidor DNS adecuado y la clave a usar para la firma digital de las actualizaciones.

```
zone dhcp.midominio.com. {
        primary 192.168.1.254;
        key CLAVE_ACTUALIZACION_DHCP;
}
```

- En el caso de la zona de resolución directa hemos dicho que vamos a usar el dominio **dhcp.midominio.com**, y por tanto es el valor que especificamos tras la palabra reservada **zone**.
- Con el parámetro **primary** le indicamos cual es el servidor primario autoritativo para dicha zona (su dirección IP). Es importante usar la dirección del servidor de DNS primario puesto que es él quien mantiene la copia maestra original de los datos de la zona que sirve para hacer las réplicas al resto de servidores secundarios de dicha zona.
- Por último con el parámetro **key** le indicamos el nombre de la clave que vamos a usar para firmar digitalmente las actualizaciones, que deberá coincidir con el nombre definido más arriba.

Exactamente igual para la zona de resolución inversa:

```
zone 1.168.192.in-addr.arpa. {
        primary 192.168.1.254;
        key CLAVE_ACTUALIZACION_DHCP;
}
```

4 Resolución de problemas.

En el fichero **/var/lib/dhcp/dhcpd.leases** (ruta válida en el caso de Debian GNU/Linux 3.0) del servidor DHCP podemos ver la base de datos de concesiones DHCP. La información sobre cada una de las concesiones DHCP de cada dirección IP asignada recientemente se almacena de modo automático en esta base de datos. La información incluye la duración de la concesión, a que cliente se ha asignado la dirección IP, las fechas iniciales y finales de la concesión, y la dirección MAC de la tarjeta de interfaz de red utilizada para obtener dicha concesión así como, opcionalmente, algunos valores adicionales asociados a la concesión.

Es importante tener en cuenta que las fechas de inicio y fin de la concesión aparecen en formato horario UTC (hora GMT) por razones técnicas.

Este es el aspecto que tiene una entrada del fichero **dhcpd.leases**:

```
lease 192.168.1.10 {
  starts 1 2003/05/05 14:40:26;
  ends 2 2003/05/06 14:40:26;
  binding state active;
  next binding state free;
  hardware ethernet 00:50:da:e7:ae:d8;
  client-hostname "barad-dur";
}
```

(1) (2) (3) (6) (4) (5)

Podemos ver que esta concesión está asociada a la dirección IP 192.168.1.10 (1), que la concesión ha empezado el 5 de Mayo de 2003 a las 14:40:26 UTC (2), que termina el 6 de Mayo a las 14:40:26 UTC (3), que dicha concesión se ha otorgado al nodo cuya dirección MAC es 00:50:da:e7:ae:d8 (4) y que el cliente DHCP ha indicado que su nombre de host es "barad-dur" (5).

Además podemos ver que la concesión está activa (6), es decir, que aún es válida. Es importante resaltar este punto puesto que los datos de las concesiones que ya han expirado se mantienen en este fichero durante algun tiempo (hasta que las direcciones IP de dichas concesiones inactivas se reutilizan para otros clientes). Por tanto conviene mirar el campo **binding state** para saber cual es el estado de la concesión (**active** o **free**).

En caso de que tuviésemos algún error en el servidor, podríamos revisar el fichero **/var/log/syslog** (o /var/log/messages, dependiendo de la distribución Linux) viendo todos los errores detallados que se han producido.

CAPÍTULO IV

Bibliografía

- The DHCP Handbook (2nd Edition). Ralph Droms, Ted Lemon, Ralph E. Droms. Sams; 2nd edition (October 28, 2002). ISBN: 0672323273.
- DHCP for Windows 2000. Neall Alcott. O'Reilly & Associates; 1st edition (January 2001). ISBN: 1565928385.
- The DHCP Handbook: Understanding, Deploying, and Managing Automated Configuration Services. Ted Lemon, Ralph E. Droms. Pearson Higher Education; 1st edition (October 1, 1999). ISBN: 1578701376.