1 Servidor DHCP de ISC

O servidor DHCP máis característico en todas as versións de Linux/Unix é o servidor DHCP de ISC (Internet Systems Consortium) [1]. A implementación do servidor DHCP de ISC, prové as tres vertentes: Servidor, Cliente e Axente Relay

O servizo DHCP basicamente permite configurar un servidor para que os clientes poidan obter automaticamente:

- Enderezo IP
- Máscara de subrede
- Porta/s de enlace (Opcional)
- IP/s do/s servidor/es DNS (Opcional)
- Dominios de busca (Opcional)
- Nome de equipo (Opcional e precísase coñecer a súa MAC)

1.1 Sumario

1.2 Instalación do servizo DHCP en Debian/Ubuntu

O único necesario é escribir nunha liña de comandos:

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

En versións anteriores de Ubuntu e Debian, o nome do paquete era *dhcp3-server*.

Sempre que fagamos cambios, hai que lembrarse de reiniciar o servidor DHCP.

```
sudo service isc-dhcp-server restart
```

Tamén podemos botar unha ollada ao log do sistema, por se algo vai mal. É útil ter aberto noutro terminal o ficheiro de log do sistema:

```
tail -f /var/log/syslog
```

É obrigatorio, que o servidor DHCP teña configurado o enderezo IP de xeito estático nos interfaces que van atender peticións

Para iniciar, reiniciar e deter o servizo DHCP, teclearemos respectivamente

```
sudo service isc-dhcp-server start
sudo service isc-dhcp-server restart
sudo service isc-dhcp-server stop
```

1.3 Configuración básica

O primeiro paso é escoller en que interface de rede vai antender peticións o servidor dhcp. Todo iso establécese no ficheiro **`/etc/default/isc-dhcp-server`**. Por defecto ese interface estableceremos que é o *eth0*.

```
# Defaults for isc-dhcp-server initscript
# sourced by /etc/init.d/isc-dhcp-server
# installed at /etc/default/isc-dhcp-server by the maintainer scripts

#
# This is a POSIX shell fragment
#

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

# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPD_PID=/var/run/dhcpd.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".
INTERFACES="eth0"
```

Toda a configuración realizarase no ficheiro **`/etc/dhcp/dhcpd.conf`**[2]

O ficheiro de configuración está cheo de comentarios aclaratorios e varios exemplos de configuracións típicas.

1.3.1 Comprobación de erros

Para comprobar os erros de sintaxe, executamos o seguinte comando:

```
dhcpd -t
```

1.3.2 Opcións básicas

No principio do ficheiro, temos varias opcións de configuración:

```
#
# Sample configuration file for ISC dhcpd for Debian
#
# Attention: If /etc/ltsp/dhcpd.conf exists, that will be used as
# configuration file instead of this file.
#
#

# The ddns-updates-style parameter controls whether or not the server will
# attempt to do a DNS update when a lease is confirmed. We default to the
# behavior of the version 2 packages ('none', since DHCP v2 didn't
# have support for DDNS.)
ddns-update-style none;

# option definitions common to all supported networks...
option domain-name "exemplo.com";
option domain-name-servers 8.8.8.8, 8.8.4.4;

default-lease-time 600;
max-lease-time 7200;
```

- **option domain-name**: para indicar o nome de dominio que ofrecerá ao cliente
- **option domain-name-servers**: para indicar os enderezos IP dos servidores DNS
- **option routers**: para indicar que porta de enlace deben usar os clientes DHCP.
- **default-lease-time**: tempo mínimo que se da a unha concesión IP (en segundos).

Se o servidor é o único servidor DHCP na rede, indicamos que ese servidor é autoritativo. Deste xeito se o servidor atopa un cliente con configuración DHCP incorrecta enviaralle unha configuración correcta para esta LAN. Deberemos quitar o comentario á directiva **authoritative** no ficheiro de configuración.

```
# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
authoritative;
```

1.3.3 Asignacións de enderezos

Ao final do ficheiro de configuración engadimos o rango de IPs que se comentaba na introdución. Onde se indica a rede IP, a súa máscara é o rango de IPs a asignar aos clientes. Definímolos co bloque **subnet**

```
subnet 192.168.0.0 netmask 255.255.255.0 {
   range 192.168.0.100 192.168.0.200;
}
```

Tamén podemos indicar en vez de un rango de enderezos, máis dun

```
subnet 192.168.0.0 netmask 255.255.255.0 {
   range 192.168.0.100 192.168.0.200;
   range 192.168.0.210 192.168.0.230;
}
```

Podemos incluír tantas declaracións *subnet* como enderezos teñamos configurados de forma estática.

1.3.4 Asignacións estáticas

Co bloque host, podemos establecer reservas de enderezos para un determinado equipo. Indicar que o nome para a reserva non ten porque coincidir co nome de equipo. Indicaremos cal é o enderezo MAC do cliente e o enderezo IP que se desexa asignar *fóra* do rango de IPs que se asigna para os equipos que non teñen reserva.

Tamén podemos non especificar o enderezo IP asisgnado. Así so estaremos identificando cun nome, un enderezo MAC e/ou algunhas opcións. Neste caso, o enderezo IP tomarase da rede á que esté conectado.

```
host equipo1 {
   hardware ethernet 08:00:07:26:c0:a5;
   fixed-address 192.168.1.20;
}
```

1.3.5 Grupos

Podemos agrupar varios bloques *host* dentro dun bloque **group**

```
group {
  host equipo1 {
     hardware ethernet 08:00:07:26:c0:a5;
     fixed-address 192.168.1.20;
  }
  host equipo2 {
     hardware ethernet 08:00:07:26:c0:b5;
     fixed-address 192.168.1.21;
  }
}
```

1.3.6 *pool* de enderezos IP

O bloque **pool** pode ser usado para definir conxuntos de enderezos que deben ser tratados de forma diferenciada doutro conxunto de enderezos, na mesma subrede ou rede compartia.

Por exemplo, podemos facer que un grupo de equipos que definimos coa directiva host, tomen enderezo e/ou opcións dun determinado rango, e todos os demáis doutro.

```
subnet 192.168.0.0 netmask 255.255.255.0 {
  option routers 192.168.0.254;

 # Unknown clients get this pool.
  pool {
    option domain-name-servers dns1.exemplo.com;
    max-lease-time 300;
    range 192.168.0.200 192.168.0.253;
    allow unknown-clients;
  }

 # Known clients get this pool.
  pool {
    option domain-name-servers ns1.exemplo.com, ns2.exemplo.com;
    max-lease-time 28800;
    range 192.168.0.40 192.168.0.199;
    deny unknown-clients;
  }
}
```

unknown-clients fai referencia aos equipos definidos coa directiva *host*. As clausulas *allow* e *deny* indicarán se un cliente ten acceso ao pool ou non.

Tamén se poden definir outras clases, aparte de *unknown-clients*. Faise coa directiva *class* e é tipico definir clases a partires de parte do enderezo MAC, típicamente os 6 primeiros bytes

```
class "vms" {
    match if (substring(hardware, 1, 3) = 52:54:00);
}
```

e tamén a partires do identificador do vendedor (Vendor class). Por exemplo os dispositivos android, teñen un identificador do vendedor que comeza por "dhcpcd" e os equipos que executan un S.O. de Microsoft un que comeza por "MSFT"

```
class "Android" {
  match if substring(option vendor-class-identifier,0,5) = "dhcpcd";
}
class "Microsoft" {
  match if substring(option vendor-class-identifier,0,3) = "MSFT
```

Tamén podemos indicar unha lista de hosts

```
class "MyHosts" {
    match hardware;
}
subclass "MyHosts" 1:10:bf:48:xx:xx:xx;  # host2
subclass "MyHosts" 1:10:bf:48:xx:xx:xx;  # host3
```

Tamén podemos facer que o cliente envíe opcións propias ao servidor DHCP. Será necesario asignar a esas opción un código entre 0 e 255 que esté libre. Están reservados para o uso privado os códigos de opción entre o 224 e o 255.

Necesitariamos configurar no cliente:

```
#/etc/dhcp/dhclient.conf
option departamento code 224 = text;
send departamento "vendas";
```

E no servidor, podemos crear unha clase a partires da opción departamento que envíen os clientes:

```
option departamento code 224 = text;
class "vendas" {
    match if option departamento = "vendas";
}
class "contabilidade" {
    match if option departamento = "contabilidade";
}
```

Desta maneira, podemos asignar enderezos en diferentes pools dependendo da opción que envían os clientes:

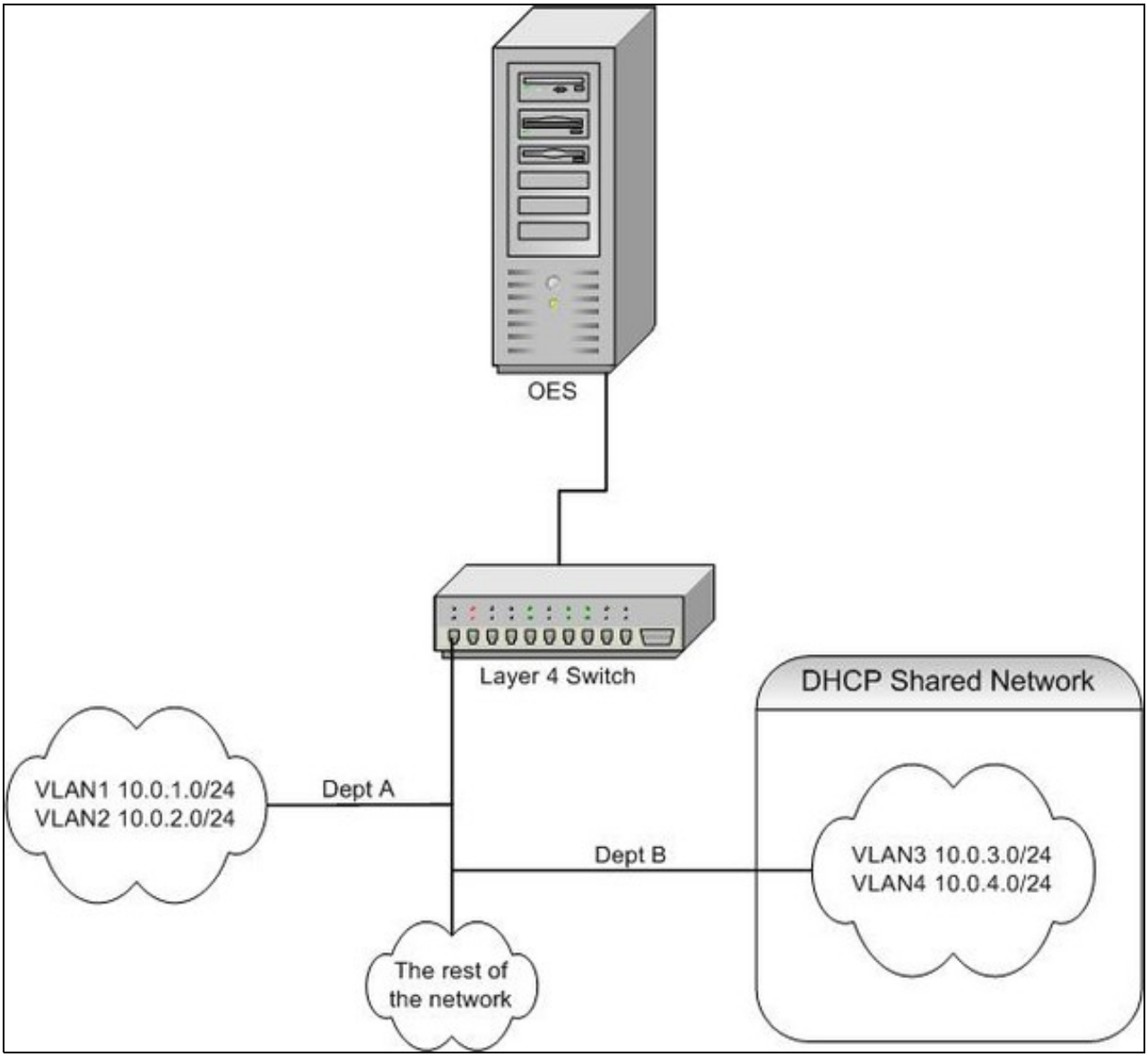
```
subnet 192.168.10.0 netmask 255.255.255.0 {
```

```
  option routers 192.168.10.254;
  pool {
    range 192.168.10.200 192.168.10.253;
    allow members of "vendas";
  }
  pool {
    range 192.168.10.40 192.168.10.199;
    allow members of "contabilidade";
  }
}
```

1.3.7 Redes compartidas

O bloque **shared-network**[3] úsase para informar ao servidor DHCP que algunhas subredes, comparten a mesma rede física. Calquera subrede nunha rede compartida debe ser declara dentro dun bloque *shared-network*.

Por exemplo, podíamolo empregar nunha rede coma a seguinte, na que conviven equipos con varios enderezos IP:




```
subnet 10.0.1.0 netmask 255.255.252.0 {
    #parameters for subnet
    range 10.0.1.1 10.0.1.254;
}
subnet 10.0.2.0 netmask 255.255.252.0 {
    #parameters for subnet
    range 10.0.2.1 10.0.2.254;
```

```
}
shared-network Vlan3_4 {
    option domain-search              "exemplo.com";
    option domain-name-servers      ns1.exempki.com, ns2.exemplo.com;
    option routers                  10.0.0.254;
    #more parameters for EXAMPLE shared-network
    subnet 10.0.3.0 netmask 255.255.252.0 {
        #parameters for subnet
        range 10.0.3.1 10.0.3.254;
    }
    subnet 10.0.4.0 netmask 255.255.252.0 {
        #parameters for subnet
        range 10.0.4.1 10.0.4.254;
    }
}
```

Neste exemplo, os equipos da VLAN1 recibirán enderezos asignados por *subnet 10.0.1.0*, os da VLAN2 por *subnet 10.0.2.0* e os das VLAN3 e VLAN4, van a recibir enderezos de *shared-network Vlan3_4* co cal, súmanse as concesións e cando se rematen as concesións de *subnet 10.0.3.0* seguirá coas de *subnet 10.0.4.0*.

1.3.8 Duración da concesión

A duración da concesión especifícase coas opcións **`default-lease-time`**, **`min-lease-time`** e **`max-lease-time`**. A opción default é a que se aplica no caso de que o cliente non pida un tempo determinado. A duración da concesión se o cliente solicita un periodo menor que o mínimo, será o que indique a opción *min-lease-time* e se é maior que o máximo será *max-lease-time*

Todos os tempos están expresados en segundos.

1.3.9 Opcións de configuración

Mediante a directiva **option** poden definirse diversas opcións que son entregadas aos clientes nunha concesión. Estas opcións, poden ser definidas de forma global, nas subredes, nas redes compartidas, nos pool, nos grupos e a nivel de host. A lista global, sempre se constrúe partindo do máis global e chegando ao máis particular. No caso de conflictos prevalece sempre a opción máis particular.

Tamén hai opcións de configuración que non levan consigo a directiva *option*.

As opcións máis comúns son as seguintes:

- **option domain-name**: para indicar o nome de dominio que ofrecerá ao cliente
- **option domain-name-servers**: para indicar os enderezos IP dos servidores DNS por orde de preferencia.
- **option routers**: para indicar que porta de enlace deben usar os clientes DHCP por orde de preferencia.
- **default-lease-time**: tempo mínimo que se da a unha concesión IP (en segundos).
- **option broadcast-address**: enderezo de broadcast.
- **option ntp-servers**: Servidores NTP (hora).
- **next-server**: Servidor TFTP que vai a ofrecer o kernel para arrancar por rede. Soe indicarse sempre e cando o servidor TFTP e o servidor DHCP son o mesmo.
- **filename**: Nome do ficheiro (dentro do servidor TFTP) que conten o kernel para arrancar por rede.
- **option host-name**: Nome do host que se vai a servir por DHCP.

```
subnet 192.168.0.0 netmask 255.255.255.0 {
  range 192.168.0.20 192.168.0.50;

  ## Debian 6 pxe boot file ##
  filename "debian6/pxelinux.0";
  next-server 192.168.0.5;
  domain
  option subnet-mask 255.255.255.0;
  option broadcast-address 192.168.0.255;
  option routers 192.168.0.5;
  option domain-name "exemplo.com";
  option domain-name-servers 8.8.8.8 8.8.4.4;
  option ntp-servers 192.168.0.6;
}
```

1.3.10 Logging

Por defecto, todos os logs do servizo DHCP van a /var/log/syslog. Isto pode resultar engorroso xa que se mesturan con todos os logs do sistema.

Para conseguir que os logs do DHCP se escriban a outro ficheiro, por exemplo `/var/log/syslog` indicamos no ficheiro de configuración a directiva **`log-facility`** e poderá tomar un dos seguintes valores: *auth, authpriv, cron, daemon, ftp, kern, lpr, mail, mark, news, ntp, security, syslog, user, uucp* e *local0* ata *local7*

```
log-facility local7;
```

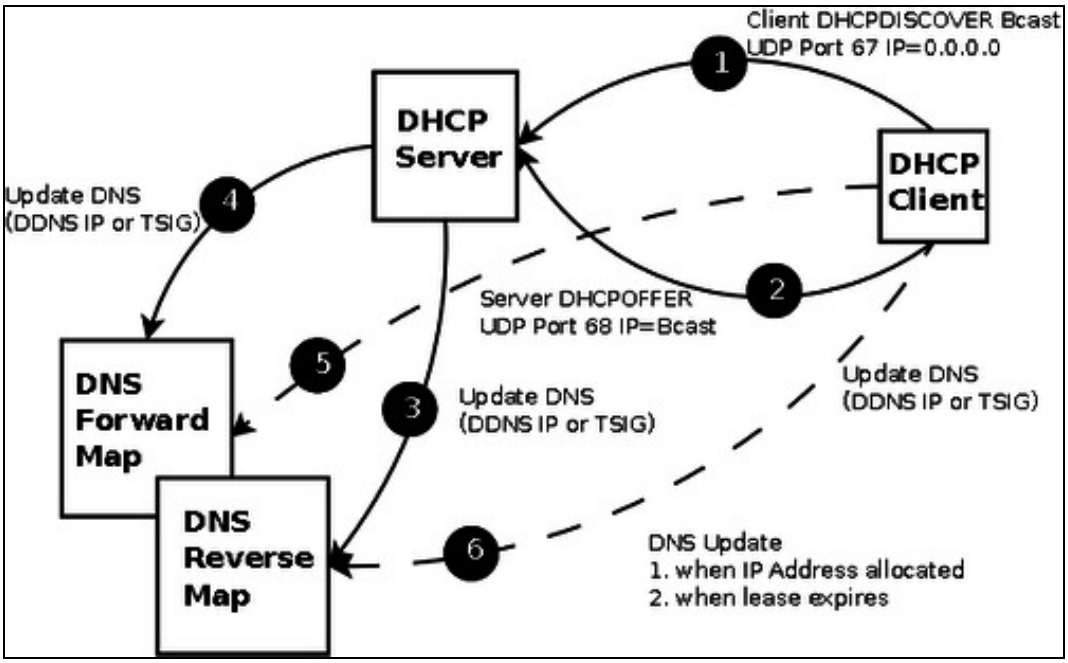
Os valores desde *local0* ata *local7* indican que é un log personal, e teremos que definilo na configuración do servizo **rsyslog**. Para facer isto, introducimos no ficheiro `/etc/rsyslog.conf` ou nun ficheiro rematado en *.conf* dentro do directorio `/etc/rsyslog.d` como por exemplo `/etc/rsyslog.d/dhcpd.conf` a seguinte liña

```
local7.* /var/log/dhcpd.log
```

O ficheiro `/var/log/dhcpd.log` debe existir, e o usuario e grupos propietario deberán ser root e adm respectivamente.

1.4 Integración con servidores DNS

Unha opción moi interesante é que os servidores DHCP actualicen os enderezos asignados aos equipos directamente no servidor DNS. O servidor DHCP de ISC, tamén pode facelo[4].



Haberá que previamente permitir as actualizacións dinámicas nas zonas que se necesite (directas e inversas). Previamente, no ficheiro de configuración `/etc/dhcp/dhcpd.conf`, temos que engadir o seguinte:

- Activar actualizacións dinámicas con *ddns-updates*. Por defecto xa están activadas.
- Nome do dominio que queremos actualizar (pode ser diferente, dependendo de se a configuración é global, para unha subrede, para un host, grupo, pool, ...) Cóllese da opción *option domain-name*, anque tamén pode especificarse con *ddns-domainname* e o dominio inverso con *ddns-rev-domainname*
- Estilo de actualización. Sempre escollemos `interim`

```
ddns-updates on;

ddns-update-style interim;
```

Se queremos que actualice tamén as reservas estáticas, temos que engadir

```
update-static-leases    on;
```

Por cada unha das zonas engadimos unha bloque coma o seguinte indicando o enderezo IP do servidor DNS primario, que é autoridade na zona. Fixarse que os nomes da zona son FQDN e rematan en "."

```
zone exemplo.com. {
  primary 127.0.0.1;
}

zone 0.168.192.in-addr.arpa. {
  primary 127.0.0.1;
}
```

1.4.1 Actulización de zonas empregando chaves

Se o servidor DNS é o servidor de BIND, e empregamos chaves, para permitir as actualizacións, necesitaremos incluír a mesma chave (as que se empreguen no servidor DNS) ben explicitamente, ou a través dunha instrucción *include*

```
key "update-key" {
        algorithm hmac-md5;
        secret "HnYDB82NMCr4NBjv+YTsyw==";
};
zone exemplo.com. {
  primary 127.0.0.1;
  key update-key;
}
zone 0.168.192.in-addr.arpa. {
  primary 127.0.0.1;
  key update-key;
}
```

Comprobamos no ficheiro de log, que se engadiron or rexistros nas zonas indicadas

```
Nov 13 14:22:50 dhcpserver named[3031]: client 127.0.0.1#24526/key update-key: signer "update-key" approved
Nov 13 14:22:50 dhcpserver named[3031]: client 127.0.0.1#24526/key update-key: updating zone 'exemplo.com/IN': adding an RR at 'base
Nov 13 14:22:50 dhcpserver named[3031]: client 127.0.0.1#24526/key update-key: updating zone 'exemplo.com/IN': adding an RR at 'base
...
Nov 13 14:22:50 dhcpserver dhcpd: Added new forward map from base.exemplo.com to 192.168.0.100
...
Nov 13 14:22:50 dhcpserver named[3031]: client 127.0.0.1#24526/key update-key: signer "update-key" approved
Nov 13 14:22:50 dhcpserver named[3031]: client 127.0.0.1#24526/key update-key: updating zone '0.168.192.in-addr.arpa/IN': deleting r
Nov 13 14:22:50 dhcpserver named[3031]: client 127.0.0.1#24526/key update-key: updating zone '0.168.192.in-addr.arpa/IN': adding an
Nov 13 14:22:50 dhcpserver dhcpd: Added reverse map from 100.0.168.192.in-addr.arpa. to base.exemplo.com
```

1.5 DHCP failover

O ter dispoñible os nosos servizos os 365 días do ano (366 en anos bisisestos) as 24 horas do día, é algo que se está convertindo en una misión imprescindible. Entre eles, está o servizo DHCP.

Necesitamos ter 2 servidores accesibles entre si, e engadimos a seguinte configuración no que vai actuar como primario:

```
failover peer "dhcp-failover" {
  primary;
  address 192.168.0.10;
  port 647;
  peer address 192.168.0.20;
  peer port 647;
  max-response-delay 30;
  max-unacked-updates 10;
  load balance max seconds 3;
  mclt 180;
  split 12;
}
```

Con estas directivas configuramos o seguinte:

- *primary*: Fai que o servidor sexa primario.
- *address*: Enderezo IP do servidor primario.

- *port*: Puerto no que escoitará o servidor primario.
- *peer address*: Enderezo IP do servidor secundario.
- *peer port*: Puerto de escoita do servidor secundario

As demais directivas son para o tempo de comunicación.

Ademais temos que incluír a configuración do *failover peer* na declaración do *pool*. Non se pode incluír directamente nun bloque *subnet*

```
subnet 192.168.0.0 netmask 255.255.255.0{
        pool{
             failover peer "dhcp-failover";
             range 192.168.0.20 192.168.0.50;
        }
        option routers 192.168.0.1;
        option domain-name-servers 4.4.4.4;
}
```

No nodo secundario, tamén incluímos algo similar. Primeiro a declaración do *failover peer*

```
failover peer "dhcp-failover" {
  secondary;
  address 192.168.0.20;
  port 647;
  peer address 192.168.0.10;
  peer port 647;
  max-response-delay 30;
  max-unacked-updates 10;
  load balance max seconds 3;
}
```

Con estas directivas configuramos o seguinte:

- *secondary*: Fai que o servidor sexa primario.
- *address*: Enderezo IP do servidor secundario.
- *port*: Puerto no que escoitará o servidor secundario.
- *peer address*: Enderezo IP do servidor primario.
- *peer port*: Puerto de escoita do servidor primario

As demais directivas son para o tempo de comunicación. Observar que se omiten as directivas *mclt* e *split* e que os enderezos IP están cambiados con respecto ao servidor primario.

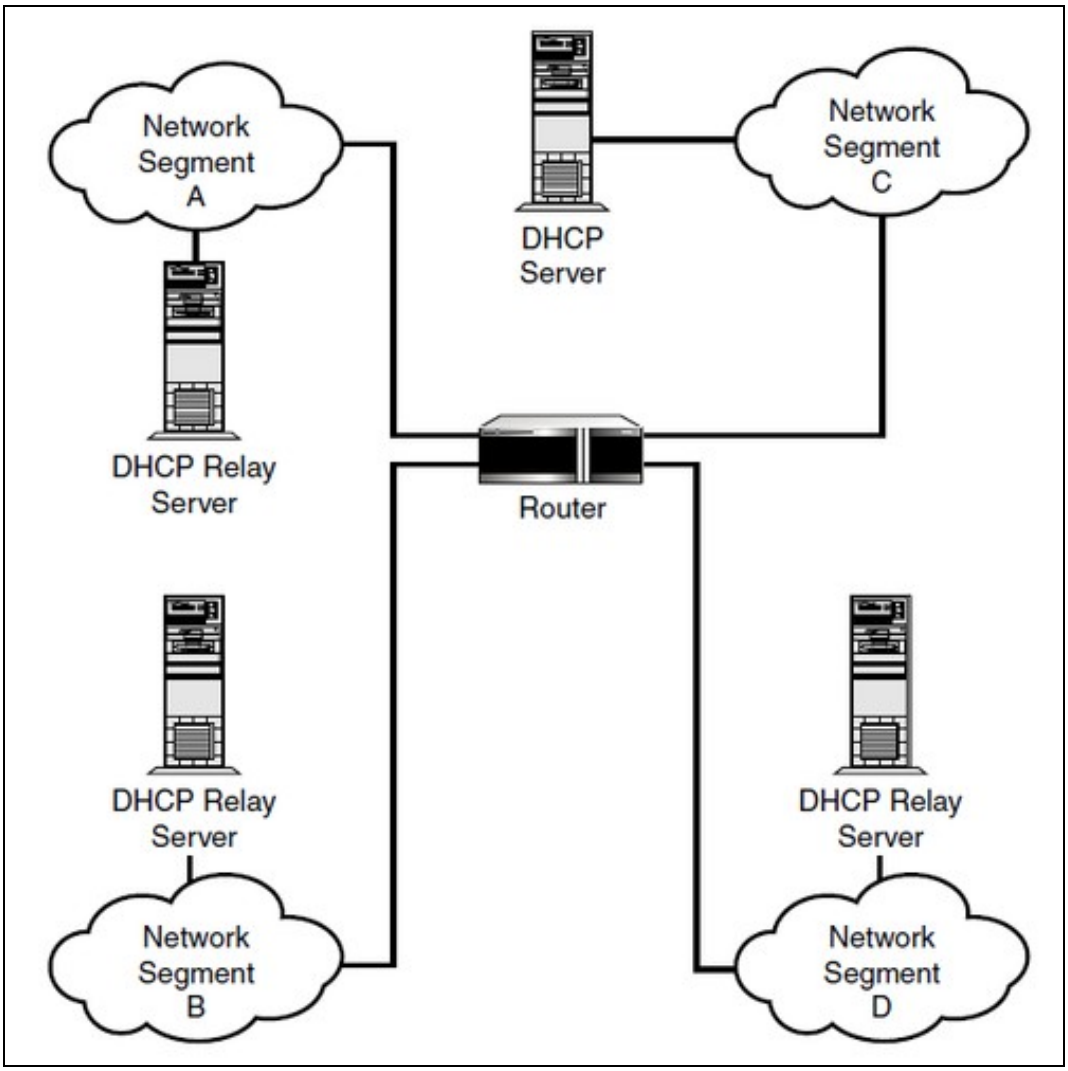
Ademais tamén temos que incluír a configuración do *failover peer* na declaración do *pool*. Loxicamente a configuración da subrede debe ser idéntica á do servidor primario.

```
subnet 192.168.0.0 netmask 255.255.255.0{
        pool{
             failover peer "dhcp-failover";
             range 192.168.0.20 192.168.0.50;
        }
        option routers 192.168.0.1;
        option domain-name-servers 4.4.4.4;
}
```

Podemos observar, que se os dous equipos funcionan correctamente, establecese un diálogo sobre as concesións asignadas, e tamén que basta que un dos dous equipos estea operativo para asignar enderezos IP aos clientes.

1.6 Axentes Relay

Nunha organización que ten a súa rede segmentada en varias subredes, non é necesario ter instalado un servidor DHCP en cada subrede. Aboda con ter un axente DHCP Relay en cada unha delas, para que reenvíe os paquetes que teñan que ver con DHCP ao servidor DHCP. Estes axentes Relay, ben poden ser equipos independentes, ou poden estar integrados nos propios routers.



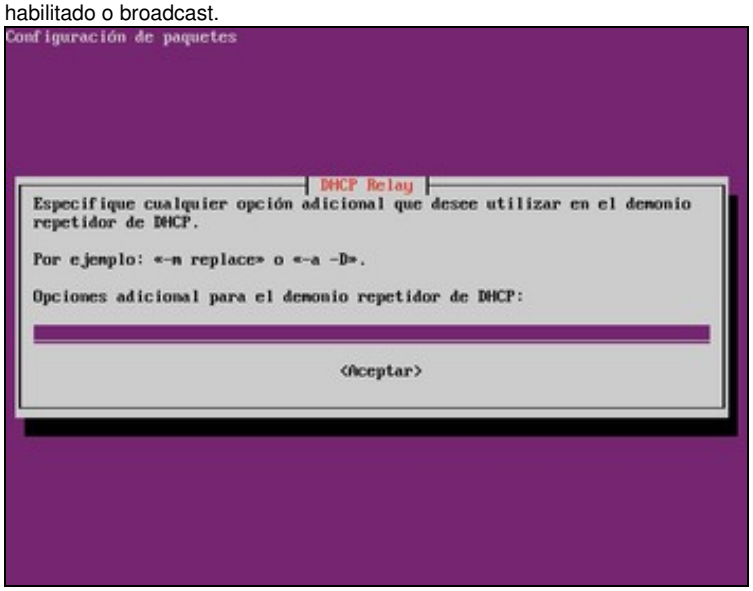
Para instalar o axente Relay nunha máquina Ubuntu, procedemos da seguinte maneira:

```
sudo apt-get install isc-dhcp-relay
```

- Instalación do axente DHCP Relay
- 

A continuación o configurador do paquete preguntanos cal é o enderezo do servidor DHCP ao que se van retransmitir os paquetes que teñan que ver con DHCP.

- 

  Indicamos en cales dos interfaces de rede, vai a escoitar por peticións DHCP. Se non poñemos ningunha, escoitará en todas as que teñan habilitado o broadcast.
- 

  Indicamos tamén as opcións que lle engadimos aos paquetes DHCP retransmitidos. Por defecto non indicamos ningunha.

Se nos equivocamos, podemos lanzar de novo o configurador executando o comando

```
sudo dpkg-reconfigure isc-dhcp-relay
```

ou editando o ficheiro */etc/default/isc-dhcp-relay*

```
# Defaults for isc-dhcp-relay initscript
# sourced by /etc/init.d/isc-dhcp-relay
# installed at /etc/default/isc-dhcp-relay by the maintainer scripts

#
# This is a POSIX shell fragment
#

# What servers should the DHCP relay forward requests to?
SERVERS="192.168.56.2"

# On what interfaces should the DHCP relay (dhrelay) serve DHCP requests?
INTERFACES=""

# Additional options that are passed to the DHCP relay daemon?
OPTIONS=""
```

Neste caso, habería que matar o proceso *dhcrelay*, que xa se inicia el so de novo coa nova configuración.

1.7 Referencias externas

1. ? ISC DHCP
2. ? .Páxina de manual de *dhcpd.conf*
3. ? Understand DHCP's "Shared Network" Option
4. ? DDNS with DHCP

- Bruno Vila Vilariño (nov 2020)