

## Unknown Title

---



Tú dijiste:

```
explica: # DHCP Server Configuration file\n#see /usr/share/doc/dhcp*/dhcpd.conf.sample # This file was created by FOG #Definition of PXE-specific options # Code 1: Multicast IP Address of bootfile # Code 2: UDP Port that client should monitor for MTFTP Responses # Code 3: UDP Port that MTFTP servers are using to listen for MTFTP requests # Code 4: Number of seconds a client must listen for activity before trying # to start a new MTFTP transfer # Code 5: Number of seconds a client must listen before trying to restart # a MTFTP transfer option space PXE; option PXE.mtftp-ip code 1 = ip-address; option PXE.mtftp-cport code 2 = unsigned integer 16; option PXE.mtftp-sport code 3 = unsigned integer 16; option PXE.mtftp-tmout code 4 = unsigned integer 8; option PXE.mtftp-delay code 5 = unsigned integer 8; option arch code 93 = unsigned integer 16; use-host-decl-names on; ddns-update-style interim; ignore client-updates; # Specify subnet of ether device you do NOT want service. # For systems with two or more ethernet devices. # subnet 136.165.0.0 netmask 255.255.0.0 {} # --- SUBRED LOCAL (10.2.7.0) --- # Declaramos la subred para que el servicio inicie, # pero al no poner 'range', NO otorgará IPs en esta red. subnet 10.2.7.0 netmask 255.255.255.0 { # Vacío: No hace nada aquí. } # --- SUBRED 10 (172.18.10.0) --- subnet 172.18.10.0 netmask 255.255.255.0 { option subnet-mask 255.255.255.0; range dynamic-bootp 172.18.10.100 172.18.10.250; default-lease-time 21600; max-lease-time 43200; option routers 172.18.10.1; option domain-name-servers 8.8.8.8; next-server 10.2.7.5; # IP de tu FOG Server } # --- SUBRED 20 (172.18.20.0) --- subnet 172.18.20.0 netmask 255.255.255.0 { option subnet-mask 255.255.255.0; range dynamic-bootp 172.18.20.100 172.18.20.250; default-lease-time 21600; max-lease-time 43200; option routers 172.18.20.1; option domain-name-servers 8.8.8.8; next-server 10.2.7.5; } # --- SUBRED 30 (172.18.30.0) --- subnet 172.18.30.0 netmask 255.255.255.0 { option subnet-mask 255.255.255.0; range dynamic-bootp 172.18.30.100 172.18.30.250; default-lease-time 21600; max-lease-time 43200; option routers 172.18.30.1; option domain-name-servers 8.8.8.8; next-server 10.2.7.5; } class "Legacy" { match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00000"; filename "undionly.kkpxe"; } class "UEFI-32-2" { match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00002"; filename "i386-efi/snponly.efi"; } class "UEFI-32-1" { match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00006"; filename "i386-efi/snponly.efi"; } class "UEFI-64-1" { match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00007"; filename "snponly.efi"; } class "UEFI-64-2" { match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00008"; filename "snponly.efi"; } class "UEFI-64-3" { match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00009"; filename "snponly.efi"; } class "UEFI-ARM64" { match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00011"; filename "arm64-efi/snponly.efi"; } class "SURFACE-PRO-4" { match if substring(option vendor-class-identifier, 0, 32) = "PXEClient:Arch:00007:UNDI:003016"; filename "snponly.efi"; } class "Apple-Intel-Netboot" { match if substring(option vendor-class-identifier, 0, 14) = "AAPLBSDPC/i386"; option dhcp-parameter-request-list 1,3,17,43,60; if (option dhcp-message-type = 8) { option vendor-class-identifier "AAPLBSDPC"; if
```

```
(substring(option vendor-encapsulated-options, 0, 3) = 01:01:01) { # BSDP List option vendor-  
encapsulated-options  
01:01:01:04:02:80:00:07:04:81:00:05:2a:09:0D:81:00:05:2a:08:69:50:58:45:2d:46:4f:47; filename  
"snponly.elf"; } } }
```

ChatGPT Plus

Te explico el archivo por partes. Es el **dhcpd.conf** que usa FOG para dar servicio DHCP y además permitir arranque por red (PXE).

---

## ◆ Cabecera y opciones PXE

Bash

```
# DHCP Server Configuration file  
# This file was created by FOG
```

Indica que es el archivo de configuración del servidor DHCP (ISC DHCP) generado por FOG.

---

## ◆ Espacio de opciones PXE

Bash

```
option space PXE;  
option PXE.mtftp-ip code 1 = ip-address;  
option PXE.mtftp-cport code 2 = unsigned integer 16;  
option PXE.mtftp-sport code 3 = unsigned integer 16;  
option PXE.mtftp-tmout code 4 = unsigned integer 8;  
option PXE.mtftp-delay code 5 = unsigned integer 8;
```

Aquí se define un **espacio de opciones personalizado llamado PXE**.

Estas opciones se usan para:

- MTFTP (Multicast TFTP)
- Comunicación cliente-servidor durante el arranque por red

Normalmente FOG no usa multicast siempre, pero deja preparadas estas opciones.

---

## ◆ Opción arquitectura del cliente

Bash

```
option arch code 93 = unsigned integer 16;
```

Esta opción identifica la **arquitectura del cliente PXE**:

- BIOS tradicional
- UEFI 32 bits
- UEFI 64 bits
- ARM64
- etc.

Esto es clave para saber qué archivo de arranque enviar.

---

## ◆ Parámetros generales

Bash

```
use-host-decl-names on;
ddns-update-style interim;
ignore client-updates;
```

- `use-host-decl-names on`; → usa nombres definidos en declaraciones host.
  - `ddns-update-style interim`; → estilo de actualización DNS dinámica.
  - `ignore client-updates`; → ignora intentos del cliente de actualizar DNS.
- 

## ◆ Subred donde NOquieres dar servicio (10.2.7.0)

Bash

```
subnet 10.2.7.0 netmask 255.255.255.0 {
    # Vacío: No hace nada aquí.
}
```

 Esto es importante.

Aquí declaras la subred para que el servicio DHCP arranque correctamente (ISC DHCP necesita conocer todas las interfaces),  
pero **NO hay range**, por lo tanto:

- 👉 NO entrega IPs en esa red.
- 👉 No interfiere con el DHCP existente.

Perfectamente configurado para tu escenario.

---

## ◆ **Subred 172.18.10.0/24**

Bash

```
subnet 172.18.10.0 netmask 255.255.255.0 {  
    option subnet-mask 255.255.255.0;  
    range dynamic-bootp 172.18.10.100 172.18.10.250;
```

- Define la red
- Entrega IPs desde .100 hasta .250

dynamic-bootp → compatible con clientes BOOTP (PXE antiguo).

---

## ◆ **Tiempos de concesión**

Bash

```
default-lease-time 21600;      # 6 horas  
max-lease-time 43200;          # 12 horas
```

- Tiempo normal: 6h
  - Máximo: 12h
- 

## ◆ **Puerta de enlace y DNS**

Bash

```
option routers 172.18.10.1;  
option domain-name-servers 8.8.8.8;
```

- Gateway: .1
  - DNS: Google
- 

## ◆ **PXE - next-server**

Bash

```
next-server 10.2.7.5;
```

MUY IMPORTANTE.

Esto indica:

- 👉 “El servidor TFTP/PXE está en 10.2.7.5”
- 👉 Es tu servidor FOG

Aunque el cliente esté en 172.18.x.x, arrancará contra FOG en 10.2.7.5.

---

Las subredes 20 y 30 hacen exactamente lo mismo, cambiando:

- Red
- Gateway
- Rango IP

Correctamente separadas.

---

## ◆ **4 Clases PXE (parte más importante)**

Aquí decides qué archivo de arranque se envía según el tipo de máquina.

---

### ◆ **BIOS Legacy**

Bash

```
class "Legacy" {  
    match if substring(option vendor-class-identifier, 0, 20) =  
"PXEClient:Arch:00000";  
    filename "undionly.kkpxe";  
}
```

Arquitectura 00000 = BIOS tradicional.

Se envía:

Código

undionly.kkpxe

👉 Para equipos antiguos sin UEFI.

---

### ◆ **UEFI 32 bits**

Bash

```
class "UEFI-32-2" {
    match if substring(option vendor-class-identifier, 0, 20) =
"PXEClient:Arch:00002";
    filename "i386-efi/snponly.efd";
}
```

Envía:

Código

i386-efi/snponly.efd

---

## ◆ UEFI 64 bits

Bash

```
class "UEFI-64-1" {
    match if substring(option vendor-class-identifier, 0, 20) =
"PXEClient:Arch:00007";
    filename "snponly.efd";
}
```

Arquitecturas:

- 00007
- 00008
- 00009

Todos usan:

Código

snponly.efd

👉 Es el más común hoy en día.

---

## ◆ ARM64

Bash

```
class "UEFI-ARM64" {
    match if substring(option vendor-class-identifier, 0, 20) =
"PXEClient:Arch:00011";
    filename "arm64-efi/snponly.efd";
}
```

Para equipos ARM (menos común en entornos educativos).

---

## ◆ Caso especial Surface Pro 4

Bash

```
class "SURFACE-PRO-4"
```

Algunas Surface envían un identificador diferente.

Aquí lo detectas y fuerzas snponly.efi.

---

## ◆ Apple Netboot (BSDP)

Bash

```
class "Apple-Intel-Netboot"
```

Soporte para:

- Mac antiguos con Intel
- Protocolo BSDP (Apple Boot Service Discovery Protocol)

Aquí se personaliza la respuesta DHCP para que macOS pueda arrancar por red.

---

## ◆ 🔥 ¿Qué hace todo junto?

1. NO interfiere con DHCP en 10.2.7.0
2. Da IPs en:
  - 172.18.10.0
  - 172.18.20.0
  - 172.18.30.0
3. Indica que el servidor PXE está en 10.2.7.5
4. Detecta automáticamente si el equipo es:

- BIOS
- UEFI 32
- UEFI 64
- ARM

- Apple

## 5. Envía el archivo correcto para arrancar FOG

---

# ◆ Resumen técnico

Tu configuración está:

- Bien segmentada
  - No genera conflicto en 10.2.7.0
  - Multiarquitectura
  - Lista para entornos mixtos BIOS/UEFI
- 

Siquieres, puedo explicarte también:

- Cómo funcionaría esto si FOG NO fuera DHCP
- Cómo usar IP Helper en el router en vez de DHCP en FOG
- Cómo simplificarlo si solo tienes UEFI 64

Tú dime 

**Tú dijiste:**

mi pregunta ahora es, cuando yo estudie sobre DHCP Server, hacia falta descomentar un authoritative. No recuerdo por qué o si debo hacerlo en esta situación: Estos son mis apuntes: Plantillas Examen DHCP Router OpenWRT. Lo único que debes hacer es poner el cable de Internet en el puerto Ethernet1. Y el que va al resto de equipos en el Ethernet0. El resto de puertos no los uses, pon un switch en el Ethernet0. Servidores DHCP Linux. Usamos los US24.04LTS. Contraseña: ubuntu Password: ubuntu Instalamos paquetería: apt update && apt install isc-dhcp-server -y Ponemos la IP Estática: nano /etc/netplan/ARCHIVO.yaml IP Estática. netplan apply Una vez instalada la paquetería, debemos ir al archivo: /etc/default/isc-dhcp-server Debemos poner el nombre de la interfaz que vamos a usar para dar el servicio de DHCP. Podemos saber cual es cuando hacemos "ip a". Ahora debemos modificar el archivo: /etc/dhcp/dhcpd.conf Recuerda que en esta ocasión hay que tener "descomentadas" las líneas: authoritative; ddns-update-style none; Ahora cada uno de los dos servidores tendrá una configuración distinta. Explicación FAILOVER. Servidor Primario. Servidor Secundario. A continuación, debemos establecer las pools y sus configuraciones. Esto debe estar en COMÚN en ambos servidores. Recuerda: El tiempo de las concesiones se pone en segundos. Parte en común. Guardamos y hacemos: systemctl restart isc-dhcp-server systemctl status isc-dhcp-server Ahora configuraremos IP estática de router MIKROTIK para luego acceder desde GUI: ((MIKROTIK)) Antes de encender la máquina o de conectar ningún cable: Ir a la sección de "Configure: Network" y meter más interfaces de red. Lo encendemos: La contraseña es "admin" y no tiene "pass". Simplemente dale a Intro. Te pedirá que le pongas una contraseña nueva: pon "a". Una vez que estemos dentro, debes ponerle una IP Ver IPs: ip

address/print Cambiar IP: ip address/add address=192.168.1.2/24 interface=ether1 Quitar IP: ip address/remove 4 (o el número que sea) Ahora usamos la siguiente guia para el resto de la configuración: Guia Mikrotik MUY IMPORTANTE: Cuando el Router Mikrotik actua de Relay (puede que en otras ocasiones también), debemos activar el NAT. Esto permitirá que cuando los Servidores DHCP hagan sus "leases", se configuren bien el tiempo de las concesiones. En caso contrario, solo se les asignará "30 minutos" (1800 segundos). Anexos Poner alias "busybox": alias b=/gns3/bin/busybox Comprobar "leases" de los servidores DHCP Linux: /var/lib/dhcp/dhcpd.leases sudo grep -B 12 'binding state active' /var/lib/dhcp/dhcpd.leases Mejor aún: dhcp-lease-list dhcp-lease-list --parsable Recuerda poner la "b" del alias de "busybox" delante de los comandos de prueba. Mostrar ip: ip a Mostrar gateway: ip route Mostrar configuración DNS: cat /etc/resolv.conf Estatus de los servidores DHCP: journalctl -u isc-dhcp-server systemctl status isc-dhcp-server grep failover /var/log/syslog grep -i state /var/log/syslog tail -f /var/log/syslog | grep dhcp check syntax dhcpcd.conf dhcpcd -t -cf /etc/dhcp/dhcpcd.conf Configuración OpenWRT GUI. Guia OpenWRT GUI. Configuración del Tunel En un contenedor del Proxmox o una VM. Configura el SSH para que permita conectarse por ROOT si hace falta: nano /etc/ssh/sshd\_config: busca "PermitRootLogin" y dale valor de "yes" En el mismo archivo: "GatewayPorts yes". systemctl restart sshd ssh -p 22 root@10.2.7.62 -R 10.2.7.50:1080:192.168.1.1:80 -fN ssh -p 22 "usuario"@"pc\_con\_acceso" -R "pc\_con\_acceso":1080:"pc\_del\_diagrama":80 -fN Comprueba en "pc\_con\_acceso" que está escuchando por el puerto: ss -ltnp | grep 1080 Tiene que salir algo así: Si ves 0.0.0.0:1080 o 10.2.7.62:1080 = está escuchando en la interfaz pública y podrás conectarte desde otra máquina a 10.2.7.62:1080. Si aparece 127.0.0.1:1080 solo está ligado a localhost — entonces solo desde el propio 10.2.7.62 podrás usar curl http://127.0.0.1:1080. Ahora puedes poner la ip del pc\_con\_acceso en el navegador así: http://10.2.7.62:1080 Calcular RANGO CIDR: ipaddressguide Otra de forma de calcularlo, es usar el "DHCP Setup" de Mikrotik: Pon las IPs estáticas en el Mikrotik. Ahora usa el DHCP Setup para "configurar" el servicio DHCP, pero no lo hagas del todo. Solo hasta la parte donde te salen las IPs disponibles para el POOL.

ChatGPT Plus

Muy buena pregunta 🔥 porque aquí hay una diferencia importante entre:

- Un DHCP “normal” de laboratorio/examen
- Un DHCP como el de FOG en un entorno ya existente

Vamos por partes.

## 🔥 ¿Qué hace authoritative;?

Cuando pones:

Bash

authoritative;

Estás diciendo:

👉 “Este servidor DHCP es el servidor oficial de esta red.”

¿Qué implica eso?

- Si un cliente intenta renovar una IP que **no pertenece a este servidor**
- O viene de otra red
- O tiene una concesión antigua inválida

El servidor responde con un **DHCPNAK inmediato**, obligando al cliente a pedir una IP nueva.

---

## ¿Por qué en tus apuntes del examen había que descomentarlo?

Porque en ese escenario:

- Tú eras el único servidor DHCP
- La red estaba bajo tu control
- No había otro DHCP activo

Entonces:

- Es correcto poner **authoritative**;
  - Evita problemas con leases antiguas
  - Hace que los clientes renueven correctamente
- 

## Ahora vamos a tu caso con FOG

Tu situación actual:

- En la red **10.2.7.0/24** ya existe un DHCP
- FOG NO debe dar servicio ahí
- FOG solo dará servicio en:
  - 172.18.10.0
  - 172.18.20.0
  - 172.18.30.0

Y además, seguramente esas redes dependen de routers o VLANs.

---

## ¿Debes poner authoritative; en este escenario?

 Sí, pero solo si FOG es el único DHCP en esas subredes.

Es decir:

Si en 172.18.10.0/24

NO existe otro servidor DHCP

Entonces puedes poner:

Bash

authoritative;

Y es incluso recomendable.

---

## NO deberías usarlo si:

- Hay otro DHCP activo en esas redes
  - Estás haciendo pruebas en una red compartida
  - Hay riesgo de conflicto
- 

## ¿Afecta a la red 10.2.7.0?

No.

Porque en esa red:

Bash

```
subnet 10.2.7.0 netmask 255.255.255.0 { }
```

No hay range, por lo tanto:

- 👉 No entrega IPs
- 👉 No responde a DISCOVER
- 👉 No interfiere

Aunque pongas authoritative;, no dará servicio en esa red.

---



# Diferencia clave con FAILOVER

En tus apuntes hablás de:

Explicación FAILOVER. Servidor Primario / Secundario.

En configuraciones FAILOVER:

- Es obligatorio authoritative;
- Ambos servidores deben estar sincronizados
- Es crítico para evitar inconsistencias

Pero FOG normalmente no se usa en FAILOVER DHCP.

---



## Entonces... ¿qué haría yo en tu caso?

Si:

- FOG es el único DHCP en 172.18.x.x
- No hay otro servidor en esas VLANs

Yo añadiría arriba del archivo:

Bash

```
authoritative;  
ddns-update-style none;
```

Porque:

- Hace el comportamiento más limpio
  - Evita leases antiguas
  - Es la configuración típica en entornos controlados
- 



## Pero ojo con esto (muy importante)

Si esas redes están usando **DHCP Relay en el router**:

- El router reenvía las peticiones a FOG
- FOG sigue siendo el único DHCP
- Puedes usar authoritative; sin problema

No tiene nada que ver con el NAT del Mikrotik que mencionas en tus apuntes (eso era para otro escenario específico).

---

## Resumen claro para tu caso

Escenario	¿authoritative?
Único DHCP en esas subredes	Sí
Hay otro DHCP activo	No
Red de laboratorio aislada	Sí
FOG solo como ProxyDHCP	No necesario

---

## Pregunta clave ahora

¿Tu FOG está funcionando como:

1. • DHCP completo (como el archivo que me has pasado)?
2. • ProxyDHCP (sin dar IP, solo dando PXE)?

Porque si fuera ProxyDHCP, la respuesta cambia completamente.

Si me confirmas eso, te lo ajusto exactamente a tu topología.

Tú dijiste:

Los apuntes que te pasé eran de la configuración de un FAIL-OVER (dos DHCP server) que daban IPs en una subnet que era diferente a la suya.

ChatGPT Plus

Perfecto ahora todo encaja.

Tus apuntes eran de un escenario **FAILOVER con dos servidores DHCP** que daban servicio a una red distinta a la suya (mediante relay). Eso cambia bastante el contexto respecto a tu FOG actual.

---

## En un escenario FAILOVER

Cuando tienes:

- Servidor DHCP Primario

-  Servidor DHCP Secundario
-  Ambos comparten la misma pool
-  Atienden peticiones vía DHCP Relay

Entonces:

Bash

```
authoritative;  
ddns-update-style none;
```

Es prácticamente obligatorio.

## ¿Por qué?

Porque:

1. Ambos servidores deben considerarse **autoridad oficial de esa red**.
2. Si un cliente aparece con una lease antigua inválida:
  - El servidor debe enviar DHCPNAK inmediatamente.
3. Sin authoritative, el servidor puede ignorar ciertos estados y provocar comportamientos raros.

En failover, si no lo pones, pueden aparecer:

- Clientes que no renuevan correctamente
- Leases inconsistentes
- Estados “stale” entre primario y secundario

---

## Ahora comparémoslo con tu FOG

Tu caso actual NO es failover.

Tienes:

- Un único servidor DHCP (FOG)
- Varias subredes 172.18.x.x
- Una red 10.2.7.0 donde no debe actuar

Eso es un escenario mucho más simple.

---

# Entonces, ¿debes usar authoritative en FOG?

Sí, puedes usarlo sin problema.

Porque:

- FOG es el único DHCP en 172.18.10/20/30
- No hay configuración failover
- No hay otro servidor compitiendo

Y además:

La red 10.2.7.0 no se ve afectada porque no tiene range.

---

## Diferencia clave

En FAILOVER:

- authoritative es necesario para coherencia entre servidores.

En FOG único:

- authoritative es recomendable pero no crítico.

En ProxyDHCP:

- No se usa (porque no asigna IPs).
- 

## ¿Qué pasaría si NO lo pones?

Nada grave.

Simplemente:

- Si un cliente llega con una IP antigua incorrecta
- El servidor puede tardar más en forzarle a pedir una nueva

Pero funcionará igualmente.

---

## Conclusión clara para tu situación

Tu configuración actual de FOG:

- ✓ Puede funcionar perfectamente sin authoritative;
  - ✓ Es totalmente válido añadirlo
  - ✓ No afecta a la red 10.2.7.0
  - ✓ No genera conflicto si es el único DHCP en esas VLANs
- 

Si quieres, te explico también:

- Qué diferencia hay exactamente entre DHCP normal, ProxyDHCP y Relay en cuanto a authoritative.
- Por qué en algunos entornos empresariales NO se pone.

Eso ya es nivel bastante fino de redes 😊

ChatGPT puede cometer errores. Considera verificar la información importante. Ver [preferencias de cookies](#).