DHCP

- **Configuration guides | IP Addressing | DHCP**

1. Overview

- RFC 2131
    - Dynamic Host Configuration Protocol
    - Based on BOOTP
    - client/agent relay/server model
    - UDP port 67

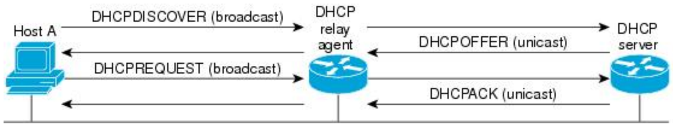


Figure 1. DHCP request for an IP address from a DHCP Server

2. Protocol

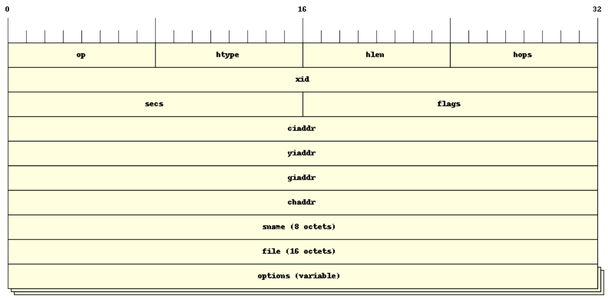


Figure 2. DHCP message

FIELD	OCTETS	DESCRIPTION
op	1	Message op code / message type. 1 = BOOTREQUEST, 2 = BOOTREPLY
htype	1	Hardware address type
hlen	1	Hardware address length
hops	1	Client sets to zero, optionally used by relay agents when booting via a relay agent.
xid	4	Transaction ID, a random number chosen by the client, used by the client and server to associate messages and responses between a client and a server.
secs	2	Filled in by client, seconds elapsed since client began address acquisition or renewal process.

FIELD	OCTETS	DESCRIPTION
flags	2	Flags
ciaddr	4	Client IP address; only filled in if client is in BOUND, RENEW or REBINDING state and can respond to ARP requests.
yiaddr	4	'your' (client) IP address.
siaddr	4	IP address of next server to use in bootstrap returned in DHCPOFFER, DHCPACK by server.
giaddr	4	Relay agent IP address, used in booting via a relay agent.
chaddr	16	Client hardware address.
sname	64	Optional server host name, null terminated string.
file	128	Boot file name, null terminated string; "generic" name or null in DHCPDISCOVER, fully qualified directory-path name in DHCPOFFER.
options	var	Optional parameters field.

3. DHCP Server

- Accepts address assignment requests and renewals from clients
- assign address, name server, gateways, …
- accepts broadcasts from local clients or relay agents
- database as a tree used for attribute inheritance
    - root: address pool for natural networks
    - branches: subnetwork address pools
    - leaves: manual bindings

Task: clear DHCP server variables

```
clear ip dhcp binding { <address> | * }
clear ip dhcp conflict { <address> | * }
clear ip dhcp server statistics
```

3.1. Database agent

- host (ftp, tftp, rcp) or storage that stores the DHCP bindings database.

Task:

```
ip dhcp database <url> [timeout <seconds> | write-dely <seconds>]
```

Task: Run DHCP server without database agent

```
(config)# no ip dhcp conflict logging
```

- not recommended
- TODO: add the reason

3.2. Address Pool

- specify which DHCP options to use for the client
  - If the client is not directly connected to the DHCP server (the giaddr field of the DHCPDISCOVER broadcast message is nonzero), the server matches the DHCPDISCOVER with the DHCP pool that has the subnet that contains the IP address in the giaddr field.
  - If the client is directly connected to the DHCP server (the giaddr field is zero), the DHCP server matches the DHCPDISCOVER with DHCP pools that contain the subnets configured on the receiving interface. If the interface has secondary IP addresses, subnets associated with the secondary IP addresses are examined for possible allocation only after the subnet associated with the primary IP address (on the interface) is exhausted.

Task: create a pool

```
(config)# ip dhcp pool <name>
```

Task: specify the subnet network number and mask of the address pool

```
(dhcp-config)# network <network-number> [mask | prefix-length]
```

Task: specify the secondary subnets

```
(dhcp-config)# network <network-number> [mask | prefix-length] secondary
```

Exclude IP address

```
(config)# ip dhcp excluded-address <low-address> [<high-address>]
```

Task: specify the domain name

```
(dhcp-config)# domain-name <example.com>
```

Task: specify the name server per order of preference

```
(dhcp-config)# dns-server <address> [<address2> ... <address8>]
```

Task: specify the default boot image for a client

```
(dhcp-config)# bootfile <filename>
```

Task: specify the netbios server

```
(dhcp-config)# netbios-name-server <address> [<address2> ... <address8>]
(dhcp-config)# netbios-node-type <type>
```

Task: specify the gateway

```
(dhcp-config)# default-router  <address> [<address2> ... <address8>]
```

Task: specify a custom DHCP code

```
(dhcp-config)# option <code> [instance <number>] {ascii <string> | hex <string> | <ip-address>}
```

Task: configure the duration of the lease

```
(dhcp-config)# lease <days> [<hours> [<minutes>] ]
```

Task: specify the lease for ever

```
(dhcp-config)# lease infinite
```

Task: configure the utilization mark of the current address pool size

```
(dhcp-config)# utilization mark high <percentage-number> [log]
(dhcp-config)# utilization mark low <percentage-number> [log]
```

Task: configure a DHCP address pool with secondary subnets

```
(dhcp-config)# override default-router ??
(dhcp-config)# override utilization high <percentage>
(dhcp-config)# override utilization low <percentage>
```

TODO: add explanation

Task: verify the DHCP address pool configuration

```
# show ip dhcp pool [name]
# show ip dhcp binding [address]
# show ip dhcp conflict [name]
# show ip dhcp database [url]
# show ip dhcp server statistics [type-number]
```

3.3. Address bindings

- Mapping between the IP address and MAC address of a client

Task: display the current mapping

```
# show ip dhcp binding
```

3.3.1. automatic bindings

- dynamically maps hardware address to an IP address from a pool.

- stored in volatile RAM and periodically copied to database agent

3.3.2. manual binding

- MAC address of hosts are found in the DHCP database

- stored in NVRAM

- can be configured

  - individually and stored in NVRAM

  - in batch from text files

Task: specify the IP address and subnet mask of the client

```
(dhcp-config)# host <address> [<mask>| </prefix-length]
```

Task:

```
(dhcp-config)# client-identifier <unique-identifier>
```

- Send with DHCP option 61
- Unique identifier
  - 7-byte: 1byte for the media , 6 byte for the MAC address
  - 27-byte: vendor, MAC address, source interface of the client

Task: determine the client identifier

```
# debug ip dhcp server packet

DHCPD:DHCPDISCOVER received from client 0b07.1134.a029 through relay 10.1.0.253.
DHCPD:assigned IP address 10.1.0.3 to client 0b07.1134.a029.
```

Task:

```
(dhcp-config)# hardware-address <hw-address> [<protocol-type> | <hw-number>]
```

- For client who can not send a client identifier in the packet

Task:

```
(dhcp-config)# client-name <name>
```

- Do not include the domain name

3.4. Static mapping

- from customer-created text file that DHCP server reads at boot
  - short configuration: no need for several numerous host pools with manual bindings
  - reduce space required in NVRAM to maintain address pools
- The file format has the following elements:
  - Database version number
  - End-of-file designator
  - Hardware type

  - Hardware address

  - IP address

  - Lease expiration

  - Time the file was created

Example

```
*time* Jan 21 2005 03:52 PM
*version* 2
!IP address    Type    Hardware address     Lease expiration
10.0.0.4 /24   1       0090.bff6.081e       Infinite
10.0.0.5 /28   id      00b7.0813.88f1.66    Infinite
10.0.0.2 /21   1       0090.bff6.081d       Infinite
*end*
```

Task: configure the DHCP server to read a static mapping text file

```
(dhcp-config)# origin file <url>
```

3.5. Pings

- DHCP server pings an IP address twice before assigning it to a client.

- If the ping is unanswered after waiting for 2 seconds, the server assumes that the address is not in use.

Task:

```
(config)# ip dhcp ping packets <number>
```

Task:

```
(config)# ip dhcp ping timeout <milliseconds>
```

3.6. BOOTP interoperability

Task: configure the DHCP server to not reply to any BOOTP requests.

```
(config)# ip dhcp boot ignore
```

Task: forward ignored BOOTP request packets to another DHCP server

```
(config)# ip helper-address <a.b.c.d>
```

3.7. Central DHCP server

- Updates specific DHCP options for remote DHCP server

Task: import DHCP option parameters from central DHCP server

```
(dhcp-config)# import all
(config)# interface <type> <number>
(config-if)# ip address dhcp
```

Task: display the options that are imported from the central DHCP server

```
# sh ip dhcp import
```

3.8. Option 82

- DHCP option contains information known by the relay agent
- for dynamic IP addresses allocation
- TOBECOMPLETED
- By default, OS DHCP server uses info provided by option 82

Task: Enable DHCP address allocation with option 82

```
(config)# ip dhcp use class
```

Task: define a DHCP class and relay agent information patterns

```
(config)# ip dhcp class <name>
(dhcp-class)# relay agent information
(dhcp-class-info)# relay-information hex <pattern> [*] [bitmask <mask>]
```

Task: display DHCP class matching results

```
# debug ip dhcp server class
```

3.8.1. Static route with the next-hop dynamically obtained through DHCP

TODO: explanation/context

Task: assign a static route for the default next-hop device when the DHCP server is accessed for an IP address

```
# ip route <prefix> <mask> {<ip-address> | <interface-number> [<ip-number>]} dhcp
[<distance>]
```

- Ensure that the DHCP client and server are defined to supply a DHCP device option 3 of the DHCP packet.
- If the DHCP client is not able to obtain an IP address or the default device IP address, the static route is not installed in the routing table.
- If the lease has expired and the DHCP client cannot renew the address, the DHCP IP address assigned to the client is released and any associated static routes are removed from the routing table.

3.9. Statistics

Task: display server statistics

```
# show ip dhcp server statistics
```

Task: reset all DHCP server counters to 0

```
# clear ip dhcp server statistics
```

4. DHCP Relay Agent

- forwards requests and replies between clients and servers not on the same physical subnet
- sets the **giaddr** field and adds option 82
- DHCP server and relay agent are enabled by default

Task: specify the packet forwarding address

```
(config-if)# ip helper-address <a.b.c.d>
```

Task: reduce the frequency with which DHCP clients change their addresses and forwards client requests to the server that handle the previous request.

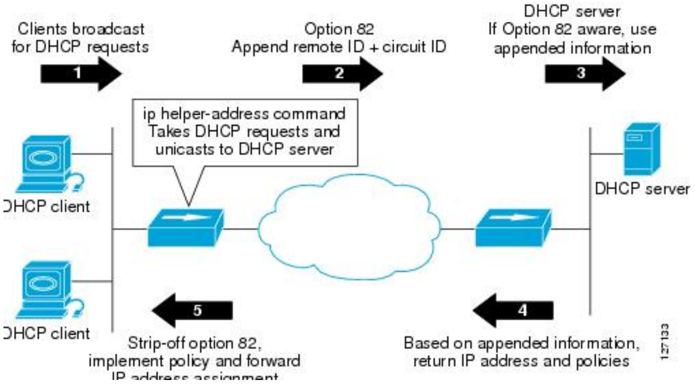
```
(config-if)# ip dhcp relay prefer known-good-server
```

- The relay agent deletes the ARP entries for addresses offered to the client on unnumbered interfaces.

Task: Disable the DHCP relay agent service

```
# no service dhcp
```

4.1. Option 82



Clients broadcast for DHCP requests
1
Option 82
Append remote ID + circuit ID
2
DHCP server
If Option 82 aware, use appended information
3
ip helper-address command
Takes DHCP requests and unicasts to DHCP server
DHCP server
DHCP client
DHCP client
5
Strip-off option 82, implement policy and forward IP address assignment
4
Based on appended information, return IP address and policies
127133

Task: insert the DHCP relay agent information option in BOOTREQUEST messages forwarded to a DHCP server

```
# ip dhcp relay information option
```

- This function is disabled by default

Task: Check whethers the relay agent information option forwarded BOOTREPLY message is valid

```
# ip dhcp relay information check
```

Task: Configure the reforwarding policy

```
# ip dhcp relay information policy {drop | keep | replace }
```

Task: Configure all interfaces as trusted sources of the DHCP relay information option.

```
# ip dhcp relay information trust-all
```

Task: Configure an interface as trusted sources of the DHCP relay information option.

```
(config-if)# ip dhcp relay information trusted
```

Task: Display all interfaces that are configure to be a trusted source for the DHCP relay information option.

```
# show ip dhcp relay information trusted-sources
```

Task: Configure per-interface support for the relay agent information option

```
(config-if)# ip dhcp relay information option-insert [none]
(config-if)# ip dhcp relay information check-reply [none]
(config-if)# ip dhcp relay information policy-action {drop | keep | replace}
```

See more optional tasks here

5. DHCP Client

Task: Acquire an IP address on an interface from DHCP

```
(config-if)# ip address dhcp
```

Task: Display the DHCP packets sent and received during troubleshooting on the client side

```
# debug dhcp detail
```

Task: Force a release of a DHCP lease

```
# release dhcp
```

> The **release dhcp** command
>
> - starts the process to immediately release a DHCP lease for the specified interface.
> - does not deconfigure the **ip address dhcp** command specified in the configuration file for the interface.

Task: Force a renewal of a DHCP lease

```
# renew dhcp
```

- The **renew dhcp** command advances the DHCP lease timer to the next stage, at which point one of the following occurs:
    - If the lease is currently in a BOUND state, the lease is advanced to the RENEW state and a DHCP RENEW request is sent.
    - If the lease is currently in a RENEW state, the timer is advanced to the REBIND state and a DHCP REBIND request is sent.
- If there is no response to the RENEW request, the interface remains in the RENEW state. In this case, the lease timer will advance to the REBIND state and subsequently send a REBIND request.
- If a NAK response is sent in response to the RENEW request, the interface is deconfigured.

5.1. Configurable DHCP client feature

- allows a client to use a user-specified client identifier, class identifier or suggested lease time when requesting an address from a DHCP server.
- options available:
    - Option 33: configure a list of static routes in the client.
    - Option 51: request a lease time for the IP address.
    - Option 55: request certain options from the DHCP server
    - Option 60: configure the vendor class identifier string to use in the DHCP interaction.
    - Option 61: specify their unique identifier

5.2. FORCERENEW Message Handling

TODO: Explain the feature

Task: configure FORCERENEW message handling

```
! Specify the key chain to be used in authenticating a request
(config)# key chain <name>
(config-keychain)# key <id>
(config-keychain-key)# key-string <text>
!
! Specify the type of authentication
(config)# interface <type number>
(config-if)# ip dhcp client authentication key-chain <name>
(config-if)# ip dhcp client authentication mode <type>
!
# ip dhcp-client forcerenew
```

6. Accounting and Security

TODO:

6.1. DHCP per interface lease limit and statistics

6.2. ARP auto-logoff

6.3. DHCP authorized ARP

6.4. DHCP accounting

6.5. DHCP secured IP address assignment