BootP and DHCP

Flexible and Scalable Host Configuration

 2005/03/11

Shortcomings of RARP

- *Reverse Address Resolution Protocol*
- Only IP Address distribution
- No subnet mask
- Using hardware address for identification
- New methods needed: BOOTP, DHCP

RARP was one of the first protocols which offers automatically an IP Address to a new connected client. But RARP is an old protocol with many disadvantages. It can only distribute an IP Address without a subnet mask. RARP uses the hardware address for identification. This make it impossible to connect new clients to the network without some administrative work.

Bootstrap Protocol (BOOTP)

A static solution with many parameters

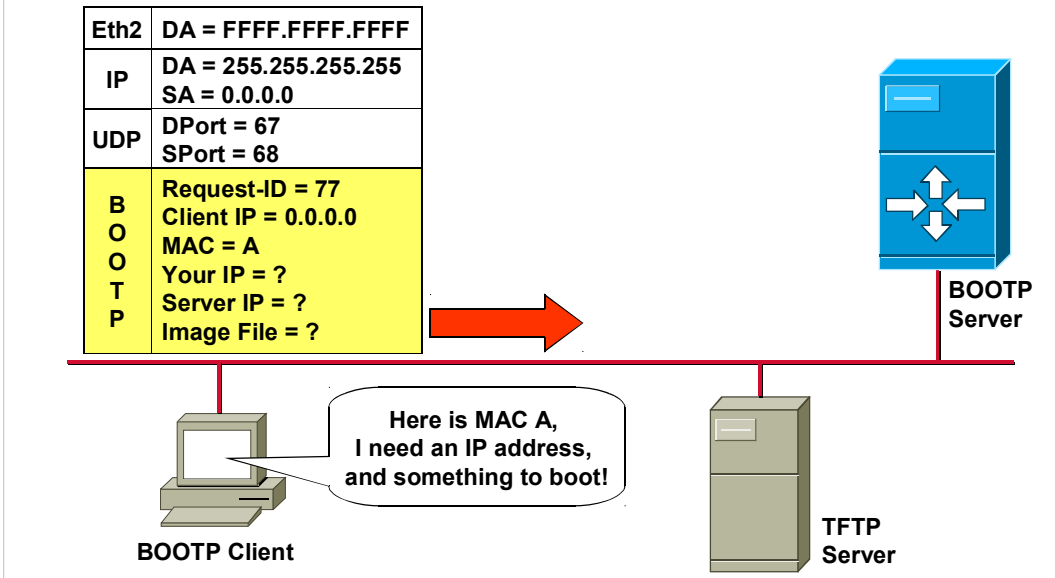
Goal

- Clients request IP address and other parameters from server
  - Subnet mask, configuration filename, ...
- IP addresses are predefined in a list
  - Fixed mapping MAC address ➔ IP address
- Defined in RFC 951 and RFC 1048

The Bootstrap Protocol can offer many important parameters to the client. The most important parameters are the subnet mask and the configuration filename. With the configuration filename it is possible to connect a no-disk client.

Also BOOTP uses a fixed mapping via hardware address (Ethernet Mac Address).

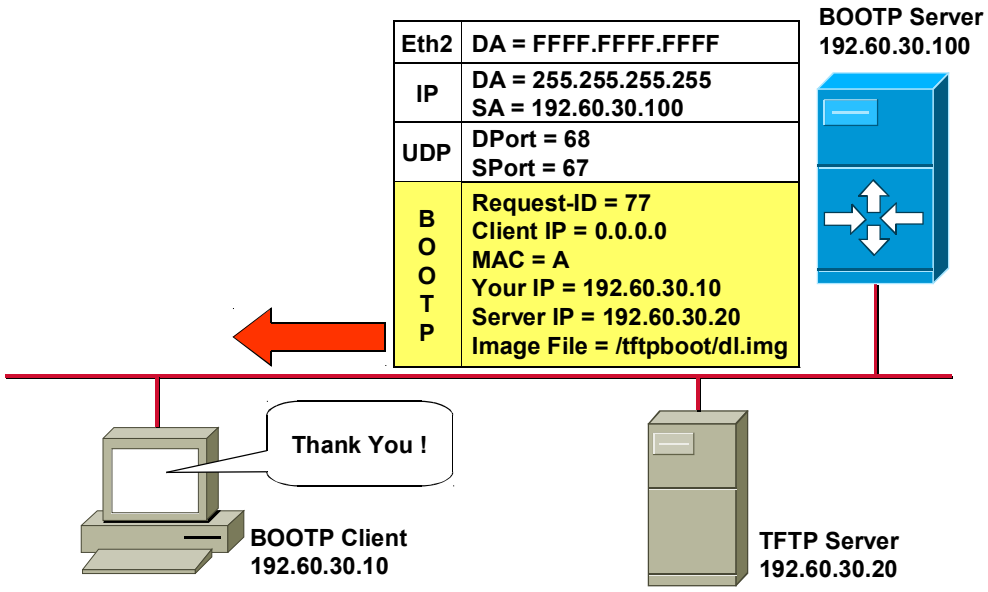
Bootstrap



(C) Herbert Haas 2005/03/11 5

In the picture above you see the classic bootstrap principle. The are 2 important servers. The TFTP server with the configuration file and the BOOTP server. After a new client connect to a network he needs an IP address and something to boot. Via an IP broadcast (BOOTP works with UPD, Ports 67 and 68) he sends out a request.

Bootstrap



(C) Herbert Haas 2005/03/11 6

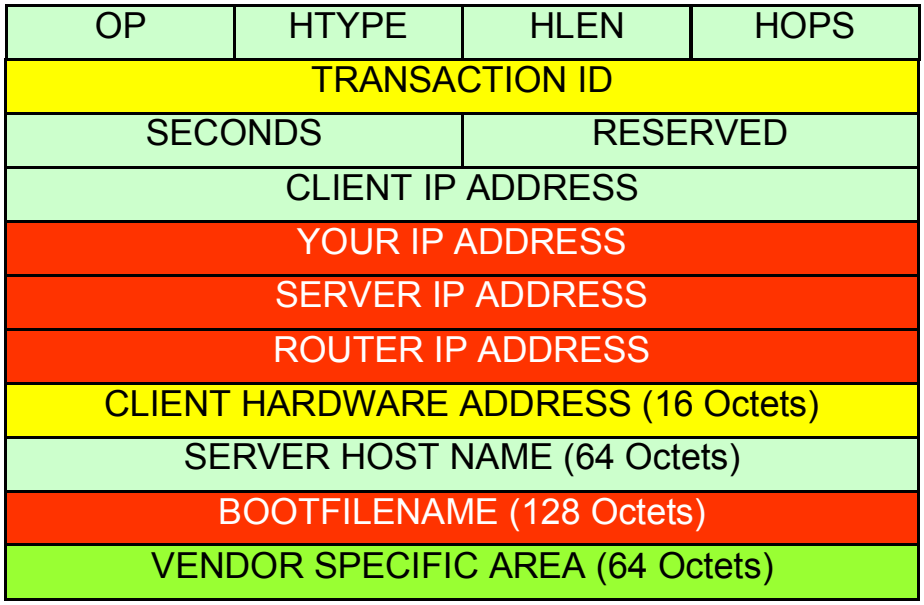
After the BOOTP server receipt the request from the BOOTP client, he uses his fixed mapping method (MAC A address = ….. IP address) to offer the client an IP address. The BOOTP server also sends the client information about the TFTP server and the name of the configuration file.

Principles

- Separation of the boot task into a BOOTP-part and a TFTP-part
- BOOTP server only needs to maintain a small database !
- Image- and configuration-files can be stored on another machine
- BOOTP client is responsible for error detection

After an error detection (timeout) there will be a retransmission. The timeout is selected randomly from a special interval, which is increased as error last on -> avoiding network overload. For the error detection the UDP and a checksum is used. Also the IP datagram has the “Do Not Fragment Bit” set to 1.

BOOTP - Message Format



In the picture above you see the BOOTP message format. One line is 4 bytes long. Note the 64-octet vendor specific area at the bottom of the frame. This space can be used for various additional messages and will be extended by DHCP.

In the middle part (red) the most important information is carried, which is the assigned IP address, the IP address of a server from which this client can boot, and an optional router IP address if this server is located on another subnet.

The detailed meaning of each field will be explained in the following slides.

BootP - Message Fields

- Operation Code (OP)
  - Message Type
- Hardware Address Type (HTYPE)
- Hardware Address Length (HLEN)
- Hops
  - Broadcast loop/storm avoidance
  - Increased/checked by routers

The Hops field is important to avoid broadcast loops in a network. Every time a BOOTP packet is checked by a router, the router increase the hops field per 1.

Operation Code (OP)

1… Boot request

2… Boot reply

Hardware Address Type (HTYPE)

Network Type (1… Ethernet 10MBit)

Hardware Address Length (HLEN)

6… Ethernet

BootP - Message Fields

- Transaction ID
  - Used for identification (random number)
- Seconds
  - Seconds elapsed since client started trying to boot
- Client IP-address
  - Filled in by client in boot request if known
- Your IP-address
  - Filled by server if client doesn't know its own address

The Transaction ID consists of a random number and ensures that a client identifies the correct reply packet among others, associated to its request. That is, both the request and the assoicuated reply have the same Transaction ID.

Seconds is set to the number of seconds that have elapsed since the client has started booting. According to RFC 951: "This will let the servers know how long a client has been trying. As the number gets larger, certain servers may feel more 'sympathetic' towards a client they don't normally service. If a client lacks a suitable clock, it could construct a rough estimate using a loop timer. Or it could choose to simply send this field as always a fixed value, say 100 seconds."

If a router is configured to forward BOOTP requests (broadcasts) then it might also wait until a certain value for "Seconds" has been exceeded. This measure would mitigate broadcast storms.

The Client might fill in its own IP address if already known and other parameters are requested.

Mostly "Your IP address" is used, which contains the IP address assigned to the client.

BootP - Message Fields

- **Server IP-address**
  - Returned in boot reply by server
- **Router IP-address**
  - Server is part of another Subnet
  - IP-address of the BootP relay
- **Client Hardware-address**
  - MAC-address of client

The "Server-IP-address" contains the IP address of an optional boot server.

If a gateway does decide to forward the request, it should look at the 'giaddr' (gateway IP address) field. If zero, it should plug its own IP address (on the receiving cable) into this field. It may also use the 'hops' field to optionally control how far the packet is reforwarded. Hops should be incremented on each forwarding. For example, if hops passes '3', the packet should probably be discarded.

The Client's HW address is needed to find an entry in the address-table at the BOOTP server.

BootP - Message Fields

- **Server Host Name**
  - Optional server host name
- **Bootfilename**
  - Contains directory path and filename of the bootfile
- **Vendor Specific Area**
  - Optionally contain vendor information of the BootP server
  - RFC 1048: also possible to mention the subnet mask, hostname, domain name, DNS, etc

Optionally, the servers domain name can be specified. This field is limited to 64 bytes.

The "Bootfilename" contains the directory path and filename of the bootfile, which is located at the server specified above.

Dynamic Host Configuration Protocol (DHCP)

A dynamic solution with even more parameters

Principles

- Nearly identical to BOOTP
  - Slightly extended messages only
  - More parameters
- Uses UDP communication
  - Client-Side: Port 67
  - Server-Side: Port 68
- Based on a leasing idea!
  - Dynamic configuration
- RFC 2131 and RFC 2132

The Dynamic Host Configuration Protocol works nearly identical to BOOTP. DHCP uses the same message format with only slightly chances.

DCHP based on a leasing idea. The IP address will be leased from the server to the client for a special time, after this time expired the client need to send his request again.

Flexible Configurations

- **Automatic:** Host gets permanent address
- **Dynamic:** Address has expiration date/time (leasing) !
- **Manual:** Fixed mapping MAC → IP

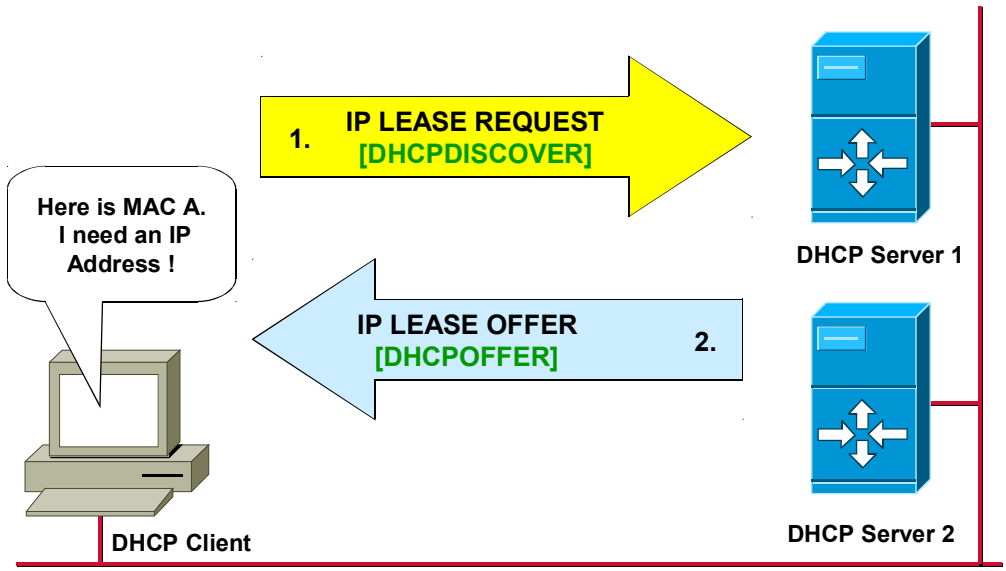
In the slide above you see the three different kind of configuration methods. BOOTP uses a manual configuration, a fixed mapping (MAC -> IP). DHCP has a dynamic configuration. The offered IP address from the server will be expire after a special time (leasing idea).

Parameters

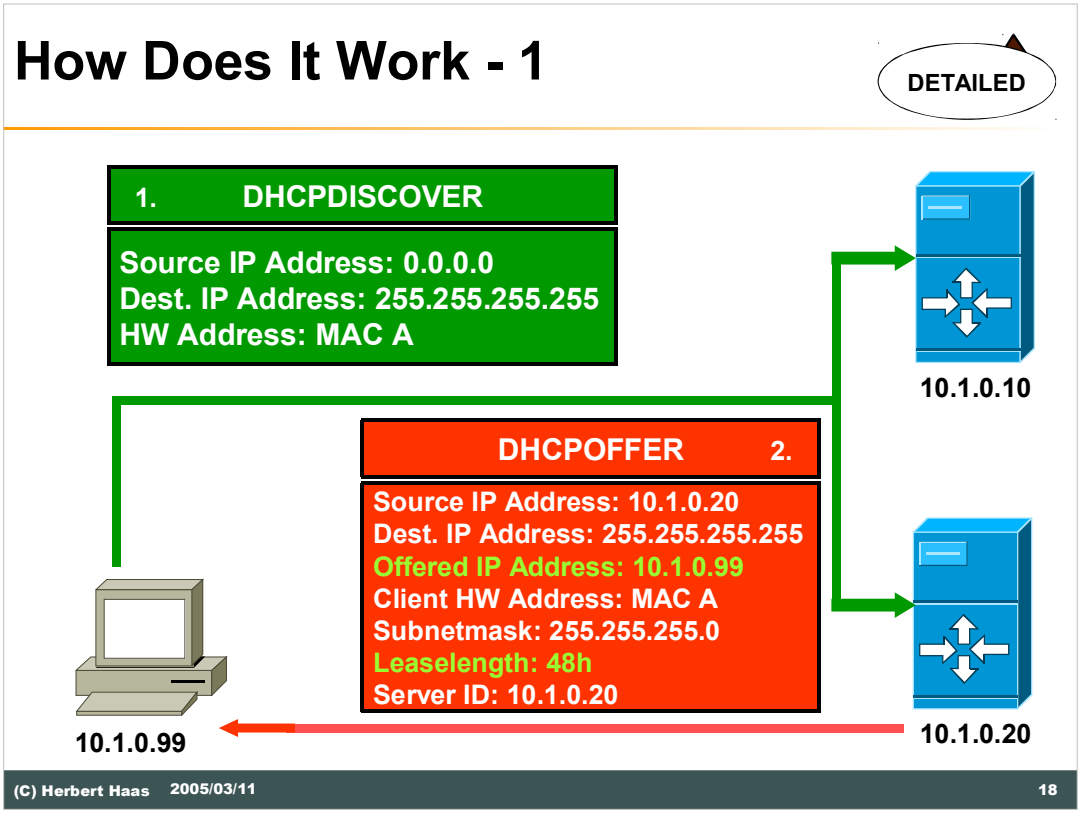
- IP address
- Subnet mask
- DNS Server
- NetBIOS Name Server
- List of default gateways
- Ethernet Encapsulation
- Router Discovery (RFC 1256)
- Path MTU Discovery (RFC 1191)
- etc...

In this slide you see some configuration parameters which can send with DCHP. It is also possible to transfer info about the maximal fragment size, ARP cache timeout, TCP keepalive, default TTL, source routing options and MTU.

How Does It Work - 1

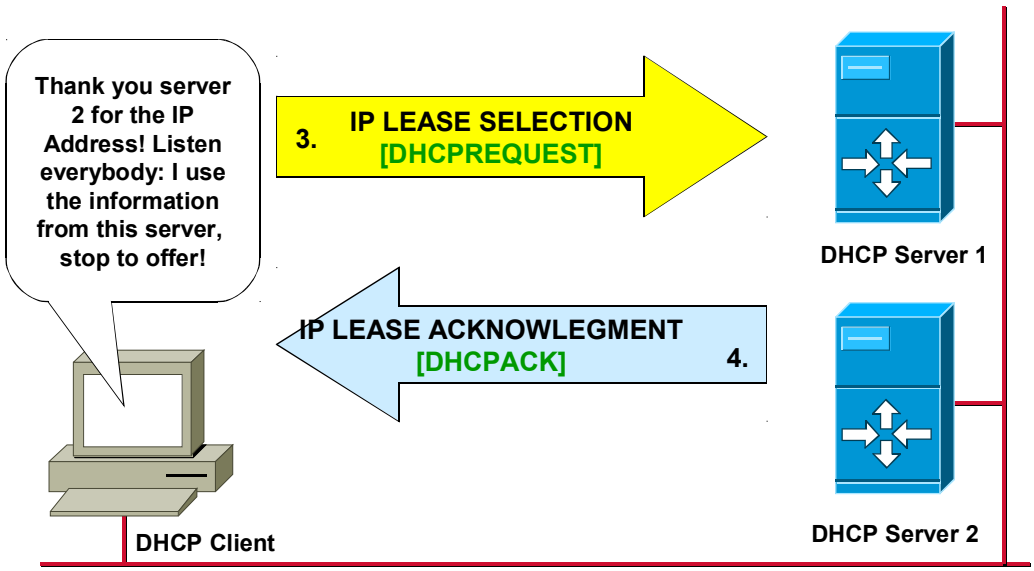


In the slide above you see the basic principle of DHCP. It is possible in a bigger network that there are not only one DHCP server. The DHCP client connect to the network at starts sending out a IP LEASE REQUEST [DHCPDISCOVER] (via broadcast, like BOOTP). Every DHCP server in the network receives this message. Every DHCP server has a own address pool. If one server has addresses left in this pool, he sends back an IP LEASE OFFER [DHCPOFFER] (in this offer there is the IP address for the client) to the client.



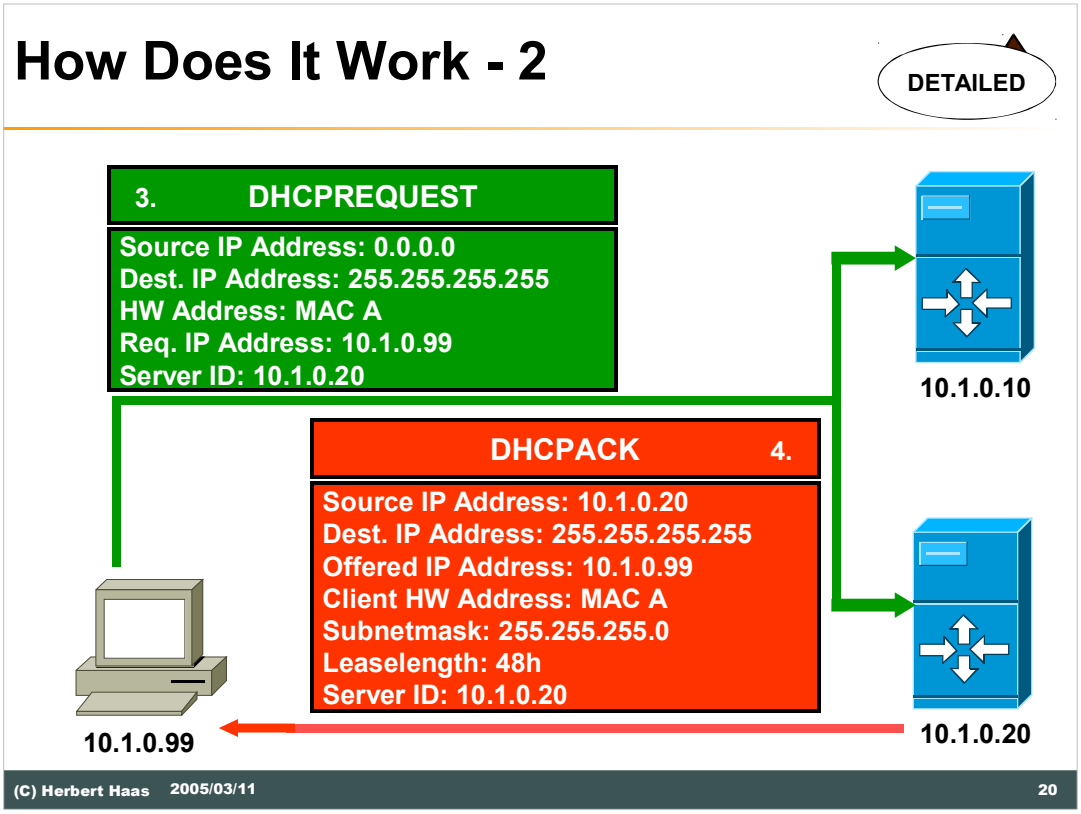
This picture shows the same as the last one, but more detailed. The client sends out his DHCPDISCOVER message and both servers receive it. Then server 10.1.0.20 sends back his DHCPOFFER. In this offer there are the IP address for the client (Offered IP Address), subnet mask, server ID and also the lease length.

How Does It Work - 2



(C) Herbert Haas 2005/03/11 19

After the client gets an offer from one server, he sends out an IP LEASE SELECTION [DHCPREQUEST] to tell the other server that he will accept the offer from server 2 and that the other servers can stop sending him offers. The DHCPREQUEST is also a broadcast.



One important thing is the server ID in the DHCPREQUEST. This server ID tells the server from which the client gets his IP address that the client will take this offered address. After server 2 receipt the DHCPREQUEST he sends back the DHCPACK to acknowledgment this lease.

Bound

- DHCPACK (success) is send by the server who's offer was accepted
- Client receives the DHCPACK
- Client enters the BOUND state
- TCP/IP is completely initialized

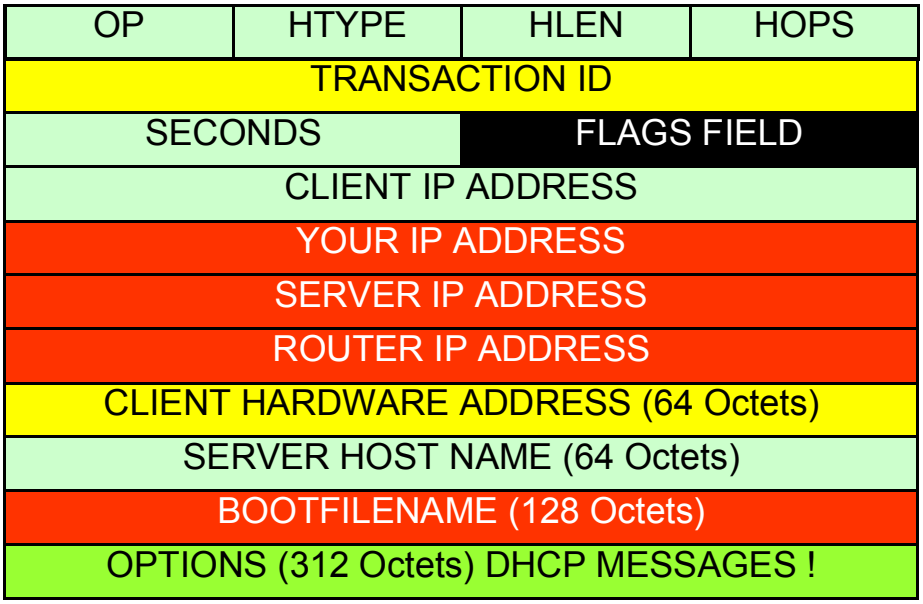
After the client receipt the DHCPACK (if all was successful) the client enters the BOUND state. After the client is BOUND TCP/IP complete initialized and the client is ready for data transfer.

DHCPNACK

- DHCPNACK (no success) will be send if
  - Client tries to lease the previous IP address, but this address is no longer available
  - Client's IP address is invalid
  - Client may have been moved to an other subnet

If the client receipt a DHCPNACK message from the server something went wrong. Connection failure, IP address invalid, client move to an other subnet, etc can all lead to a negative acknowledgment. If the client receipt this kind of message, he need to start again from the beginning (sending out a DHCPDISCOVER).

DHCP - Message Format



This picture shows the DHCP message format. It is nearly completely the same like the BOOTP message format. The only different is the OPTION Field (DHCP MESSAGES) which contains the DHCPREQUEST, DHCPOFFER, DHCPDISCOVER, etc.

DHCP-specific Message Fields

- DHCPDICOVER
  - Client broadcast to find DHCP server
- DHCPOFFER
  - Response to a DHCPDISCOVER
  - Offering an IP address
- DHCPREQUEST
  - Request the parameters offered by one server
- DHCPINFORM
  - Client ask for more information

The DCHPINFORM message is used from the client, if this client needs more information then normal.

DHCP-specific Message Fields

- **DHCPACK**
  - Acknowledgement from server to client
- **DHCPNACK**
  - Negative ACK from server to client
- **DHCPDECLINE**
  - Message from server to client indicating an error
- **DHCPRELEASE**
  - Message from server to client canceling a lease and relinquishing network address

Timer

- After DHCPACK → beginning of the lease period is registered
- Located in the DHCPACK message
  - Lease Time
  - T1 (renewal attempt)
  - T2 (sub renewal attempt)
- T1 and T2 are configured at the DHCP server
  - T1 = 0,5 x lease time
  - T2 = 0,875 x lease time

DHCP relies on a leasing idea. The offered IP address expired after a special time. There are 3 times. There is a „lease time“, a T1 and a T2. T1 and T2 based on the lease timer (T1 ~ 0.5 x lease time; T2 ~ 0,875 x lease time). This multiplier is configured at the DHCP server.

Timer

- T1 and T2 start when client is bound
- Client RENEW the lease when T1 expired
  - Client enters RENEWING state and sends a DHCPREQUEST to the server
  - If server accept, a DHCPACK contains a new lease time

After the client enters the BOUND state, both timers start. If the client still in the network after T1 expired, the client sends out an DHCPREQUEST message, because he wants to renew the lease.

Timer

- If the lease could not be RENEWED after T1, the client makes another try after T2
  - Client try to connect other DHCP server
- DHCP server can answer with
  - DHCPACK and RENEWING the lease
  - DHCPNACK to force the client to reinitialize

T2 is only a 2nd try. If something go wrong at the first time, the client still have the chance to renew his lease after T2 expired. In this try he also connect other DHCP servers.

If the client receipt a DHCPACK his lease is renew. If the client gets a DHCPNACK message the lease expired and the client starts from the beginning (he sends out a DHCPDISCOVER to all DHCP servers).

Subnets

- DHCP is related to BootP
- DHCP messages are broadcast based
  - Not forwarded by routers
  - Or routers are configured as BOOTP Relay Agent

DHCP and BOOTP sends out his packets via IP broadcast. But routers not forwarded broadcasts -> broadcast storm in the whole network. But there is a special function on routers called “BOOTP Relay Agent” which allows the routers to forward this special BOOTP/DHCP messages.