ALBERT-LUDWIGS-UNIVERSITÄT FREIBURG

Communication Systems

DHCP

University of Freiburg
Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

CoNe Freiburg

IIF INSTITUT FÜR INFORMATIK FREIBURG

Copyright Warning

- This lecture is already stolen
- If you copy it please ask the author
  - Prof. Dr. Gerhard Schneider
- like I did

Means of IP Assignments

- IP does not implement any automatic address assignment of itself
- Large networks with many end systems very common
  - Freiburg university has a 132.230.0.0/16 net assigned (how many addresses?)
  - 29000 Ethernet sockets, 16000 in use
  - ?? access points for Wireless LAN
  - No solution to assign a “private” address to every member of university (even if it would be possible because of sufficient IPs available), same even more true for IPv6 addresses

Automatic IP Configuration

- By now: large variety of mobile systems which could be connected to IP networks
  - Laptops, PDA, ...
  - Often there are more devices than connections (i.e. red painted twisted pair sockets for the university VPN)
  - Very few of these devices connected permanently
  - Manual assignment of static IPv4 addresses impractical (even if there's no shortage thanks to private IP ranges)
  - Devices may travel between such networks

Automatic IP Configuration

- IP addresses assigned by system/network administrators
- Machine has no clue of its own address during bootup (of course could be configured manually to an address)
  - No built-in address like MAC (layer 2 in Ethernets, WLAN, BlueTooth, TokenRing)
- IP has no mechanism for avoiding misconfiguration, like
  - Using wrong netmasks, network or broadcast addresses
  - Double assignment of one address, even assigning routers IP to an end system (just try it out in practical part)
- Other parameters like DNS should be configured too (explanation of DNS in a later lesson)

Automatic IP Configuration

- Several types of automatic IP address assignment
  - Centrally managed (client-server):
    - RARP – reverse ARP mostly deprecated (check the ARP literature)
    - PPP – first method of automatic IP assignment discussed last lecture, layer 2 protocol with specific IPv4 extension to handle the assignment (plus DNS)
    - DHCP – common system for centralized IP address management in broadcast networks
  - Autonomous
    - Client side address auto assignment in 169.254.X.Y class-B network (255.255.0.0 netmask)
    - IETF draft/RFC 3927 on that topic (implemented for the most modern OS, MacOS, Windows, Linux, ...)

Automatic IP Configuration

- Called **Zero Configuration IP Networking**
  - Goal: “making it possible to take two laptop computers, and connect them with a crossover Ethernet cable, and have them communicate usefully using IP, without needing a man in a white lab coat”
  - Automatic creation of a usable IP network - IP configuration plus additional standard services
  - Other/additional protocols discussed like
    - Apple's Multicast DNS/DNS-SD: similar APIs to the unicast DNS system, each machine has own list of DNS records (e.g. A, MX, PTR, SRV, etc) if machine needs to know a record it multicasts the request to 224.0.0.251 (see first practical exercise), special “.local” DNS namespace (more on DNS later in this lecture)
    - Simple Service Discovery Protocol (SSDP)
    - Service Location Protocol (SLP)

Automatic IP Configuration

- To achieve this small-network functionality in IP, there are four main areas that should be covered:
  - Allocate addresses without a DHCP server.
  - Translate between names and IP addresses without a DNS server
  - Find services, like printers, without a directory server
  - Allocate IP Multicast addresses without a MADCAP server
- At this point of lecture we are only interested in address assignment and naming service

Automatic IP Configuration– Address Assignment

- Address configuration
  - Configure interfaces with unique addresses
  - Determine which subnetmask to use (among this defines the network address – lowermost IP in a given subnet), which decides on delivery (local or via router)
  - Detect duplicate address assignment
  - Cope with address collisions
- Operation
  - Use address starting with 169.254 (class B)
  - Use last two bytes of local MAC address
  - If address collision occurs add 1 bit to the MAC and try again

Automatic IP Configuration– Address Assignement

- Concept is taken from IPv6 link local address assignment
  - IPv6 (some later lecture) uses special network prefix (fe80::) within a 64bit netmask and takes the whole MAC into host part of address
- For link local address-to-name resolution a link local multicast DNS is required – a concept similar to NetBIOS protocol is run to propagate names
- In large scale networks with various types of end systems auto IP may not solve the problem of proper IP address assignment and approach may not converge in the desired manner
- Solution to that problem – run a distinct service for address assignment
- Application running in upper OSI layers on top of IP network

DHCP – Dynamic Host Configuration Protocol – RFCs

- Application layer service to handle network layer configuration: DHCP
- BOOTP (DHCP predecessor; rather old, mostly deprecated)
- Several RFCs on BOOTP and DHCP
  - 1541 “Dynamic Host Configuration Protocol”, October 1993
  - 2131 “Dynamic Host Configuration Protocol”, March 1997
  - 2132 “DHCP Options and BOOTP vendor extensions”, march 1997
- Very few users of BOOTP
  - No dynamic lease management (do not hand out IP for ever but for certain amount of time)

DHCP – Protocol

- DHCP is able to cooperate dynamically with DNS
- Based on UDP
  - Simple connectionless extension to IP header on layer 4 (why it wouldn't be a good idea to use TCP here!?)
- DHCP operates in classical client server model
- Dynamic and static IP assignments possible
  - Predefined IP addresses could be handed out to specific hosts (matched against the MAC of requesting host)
  - IP addresses could be handed out dynamically from a pool

DHCP – Protocol

- As hosts join and leave, DHCP has to maintain list of available IP addresses
  - Each time a host joins the server allocates an arbitrary IP address from its pool of available addresses
  - Each time a host leaves, an address is returned to the pool
- Prerequisite: underlying broadcast layer 2 network (in other networks, PPP might be the better choice)
- Client sends request toward the server on standard port 67 (it does not know about the server -> broadcast request!)
- Server replies to the client on port 68

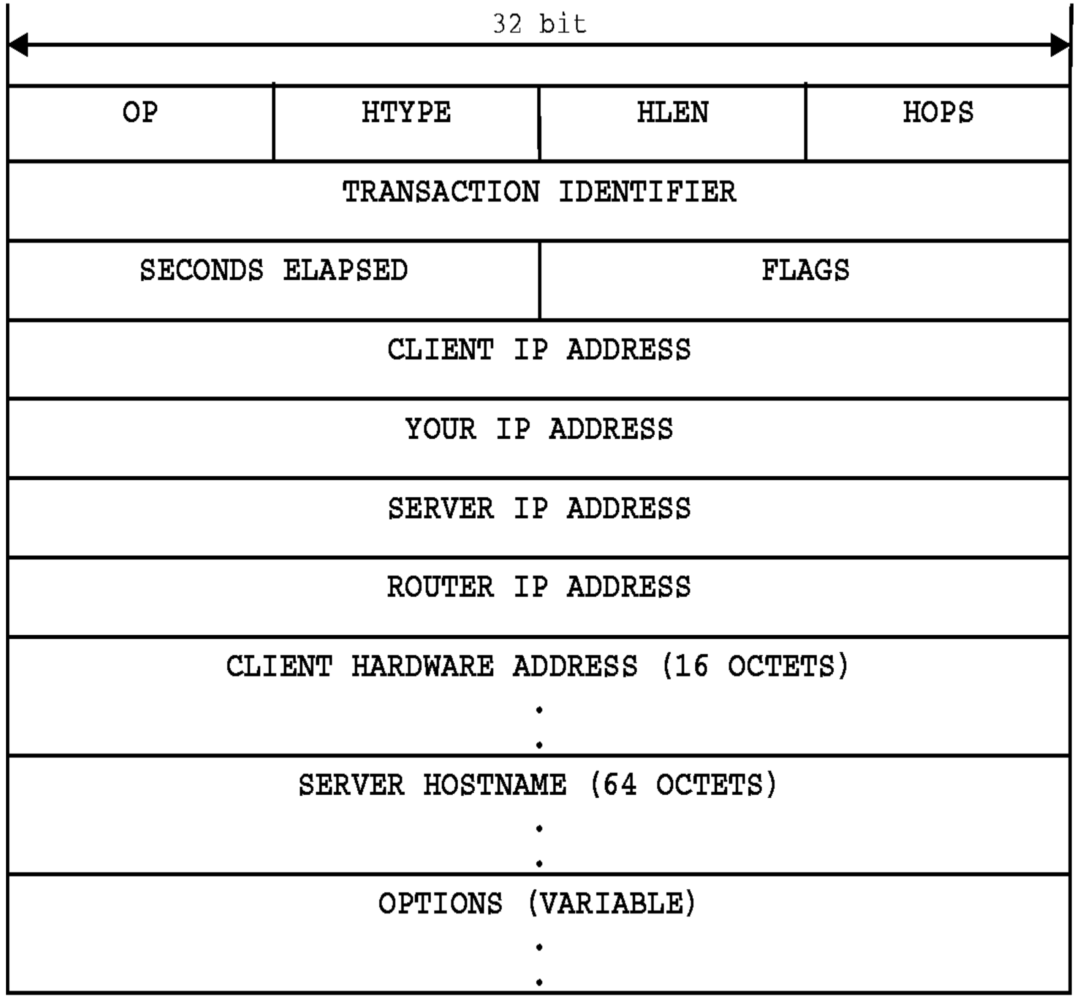
DHCP - Functions and Protocol

- There are special broadcast packets defined for MAC layer (remember ARP and its connection to MAC addresses)
  - MAC uses 48bit addresses, like 00:e0:23:e5:2a:2d
  - Highest address: ff:ff:ff:ff:ff:ff, special address -> all network adaptors accept these packets and interpret them
- DHCP request: IP packet with the named MAC and the IP broadcast address of 255.255.255.255 is put together (both are destination addresses)
- For source addresses in these packets: own MAC and special IP address 0.0.0.0
- Addresses 0.0.0.0 and 255.255.255.255 might be in use temporarily with more than one machine

DHCP

- For this reason connectionless UDP is used and protocol operates on different UDP ports
  - Concept: send out packet and forget about it
  - Server and client packets are easily distinguishable because of different ports used (for that reason the client does not see its own sent packets)
  - 3-way handshake of TCP (we will see introduction to TCP later on this lecture) would be impossible with these special addresses
  - Client would be able to process more than one server reply in the same time period

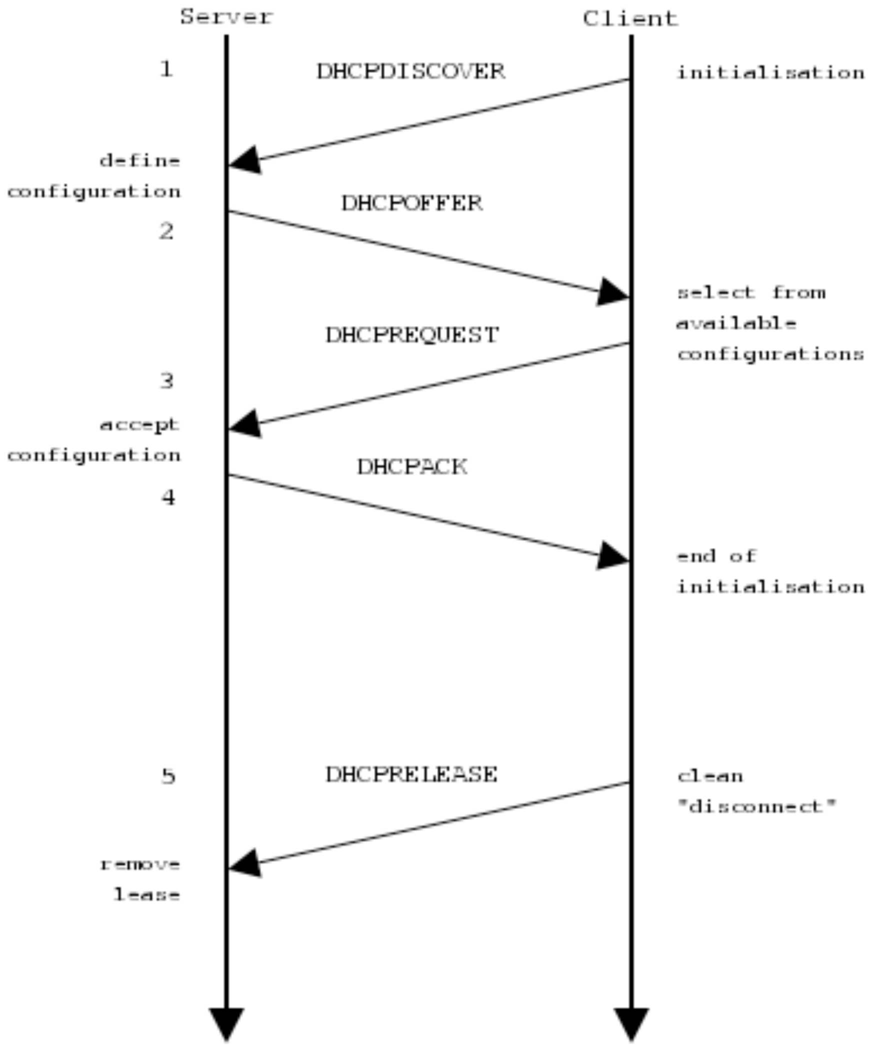
DHCP – Protocol Header



DHCP – Client Server Communication

- Client starts: DHCPDISCOVER sent to 255.255.255.255, port 67 to get a list of available DHCP servers
- Server (or more of them): reply with a suggestion of parameters as a DHCPOFFER message to 255.255.255.255 on port 68
- Client chooses from probably more than one offer and sends DHCPREQUEST to the server
- Server acknowledges with DHCPACK packet to the client or denies the request with a DHCPNACK packet
- 255.255.255.255 and 0.0.0.0 IP addresses unused again after the end of communication
- See next slide ...

DHCP – Client Server Communication



DHCP – Client Server Communication

- Wireshark snapshot of a DHCP request of a Win XP machine
  - Option 53 DHCP message type
  - Option 55 – list of configuration options requested from server
  - Client IP address (address held at that moment, could be 0.0.0.0)
  - Client hardware address

```
▽ Bootstrap Protocol
    Message type: Boot Request (1)
    Hardware type: Ethernet
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x4738a83a
    Seconds elapsed: 16179
  ▷ Bootp flags: 0x0000 (Unicast)
    Client IP address: 132.230.9.160 (132.230.9.160)
    Your (client) IP address: 0.0.0.0 (0.0.0.0)
    Next server IP address: 0.0.0.0 (0.0.0.0)
    Relay agent IP address: 0.0.0.0 (0.0.0.0)
    Client hardware address: 00:08:74:4d:6f:3f
    Server host name not given
    Boot file name not given
    Magic cookie: (OK)
    Option 53: DHCP Message Type = DHCP Request
  ▷ Option 61: Client identifier
    Option 12: Host Name = "zx-spektrum"
    Option 81: Client Fully Qualified Domain Name (15 bytes)
    Option 60: Vendor class identifier = "MSFT 5.0"
  ▷ Option 55: Parameter Request List
    End Option
```

DHCP – Client Server Communication

- Some more DHCP packet types
- Client: DHCPDECLINE, if the client has some problems/ difficulties with the servers offer
- DHCPRELEASE could be sent from the client, to return the handed out address before the end of lease
- DHCPRENEW packet for requesting renewal of a held lease
- If the server complies to it -> DHCPACK again
- How the server is able to distinguish between different clients?
  - MAC address of the end system
  - Randomly generated transaction number

DHCP – Relay

- DHCP typically operates in the same subnets to configure
- What to do in large networks with many subnets like the university?
- Not a good idea to run a service in each subnet
  - DHCP relay service
  - Proxy functionality: Typically implemented on the network like IP routers
  - Collecting the DHCP request broadcast messages
  - Rewrite the message (for proper return path) and forward it as unicast to one or more DHCP servers
  - Collect the response and translate them back to produce proper DHCP offer/ack messages

DHCP Implementations

- By now implemented in most OS (with “server” orientated capabilities)
- Server and clients:
  - Historical: bootpd, bootpc (several implementations for every UNIX flavour)
  - Now: most OS vendors use ISC DHCPv3, 4 (or derivates)
    - Standard with most Linux distributions
    - Services: dhcpd und dhclient
    - Other common clients to be found: pump (e.g. knoppix) , dhcpcd (e.g. SuSE)
  - The clients on Windows platforms are built in with ipconfig command (ipconfig -renew) or winipcfg

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Communication Systems

Christian Schindelhauer

University of Freiburg
Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

CoNe Freiburg

IIF
INSTITUT FÜR INFORMATIK FREIBURG