The IP Protocol Suite

Miklós Máté, 2021

What we've seen so far

- IPv4, IPv6 user plane: packetizing, addressing, forwarding
- IP routing: Distance Vector or Link State
- Layer 2: Ethernet, VLAN, Spanning Tree
- Layer 4: UDP, TCP

- Now:
  - Control and management plane for IP
  - Other stuff :)

Agenda

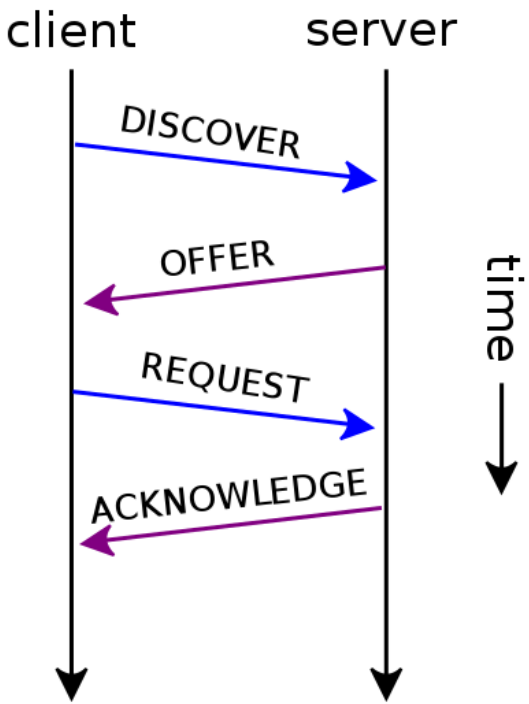
- DHCP
- ARP
- ICMP v4
- ICMP v6
- DNS
- DCCP, SCTP
- HTTP

DHCP

- We connect a PC to an Ethernet network
- The interface has a hardware address
- How do we get an IP address?
  - The address has to be unique
  - We also need a netmask (prefix length) and default gateway
  - And a DNS nameserver (more on this later)
- Dynamic Host Control Protocol
- We need a DHCP server running in the LAN (reach it via L2)

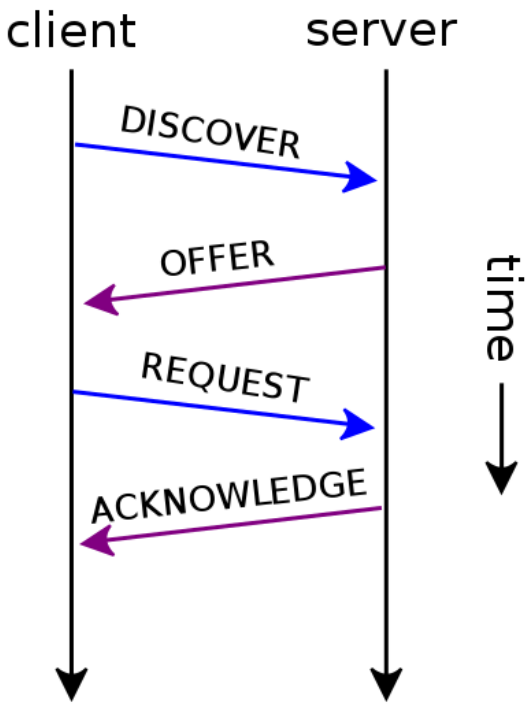
DHCP

- Discover: is there a DHCP server here?
  - Destination: Ethernet broadcast
- Offer: yes, I'm a DHCP server in this LAN
  - Destination: MAC of the asker
  - Offers an IP address to the client
- Request: I'd like to have this IP address
  - Chooses one from the received offers
- Acknowledge: it's yours



DHCP

- Assigned IP addresses are valid for a limited time
- Client must request again to renew the lease
- Client can release manually
- Servers typically remember the associations
- Security: rogue DHCP redirects the traffic



DHCP

- What is a default gateway that DHCP supplies?
- When we want to send an IP packet, the destination can be
  - In the same IP subnet as we
  - In a different subnet
- The default gateway is our neighbor that can route packets to other subnets
  - It is in our subnet
  - Has access to at least one other subnet

ARP

- An application wants to send an IP packet to a certain IP address
- We need to use Ethernet unicast, so we need to know the destination MAC
- Address Resolution Protocol
  - Request: Who has DstIP? Tell SrcIP, SrcMAC
  - Reply: DstMAC has DstIP
- Gratious ARP: announce my MAC-IP without request
  - Typically when IP address changes
- Discovered neighbors are remembered for a limited time
- End stations typically only communicate directly with their default gateway
- Security: ARP spoofing

ICMP

- Internet Control and Management Protocol
- Diagnostic and control messages, encapsulated in IP header
  - Identified by Type and Code
- Destination Unreachable (Type=3)
  - Subcodes: network unreachable, host unreachable, port unreachable, prohibited, ToS unavailable
- Redirect (Type=5): used when rerouting

ICMP

- Echo Request (Type=8), Echo Reply (Type=0)
  - Ping is the most important network diagnostic tool
- Time Exceeded (Type=11): TTL reached zero
  - Traceroute: send ping with TTL=1, TTL=2 etc. see who replies
- Request address mask, default router etc.
  - Autoconfiguration, but most of these are deprecated (use DHCP)
- Security
  - Discover topology, running services on the victim’s network
  - Outgoing ICMP can be disabled for this reason

ICMPv6

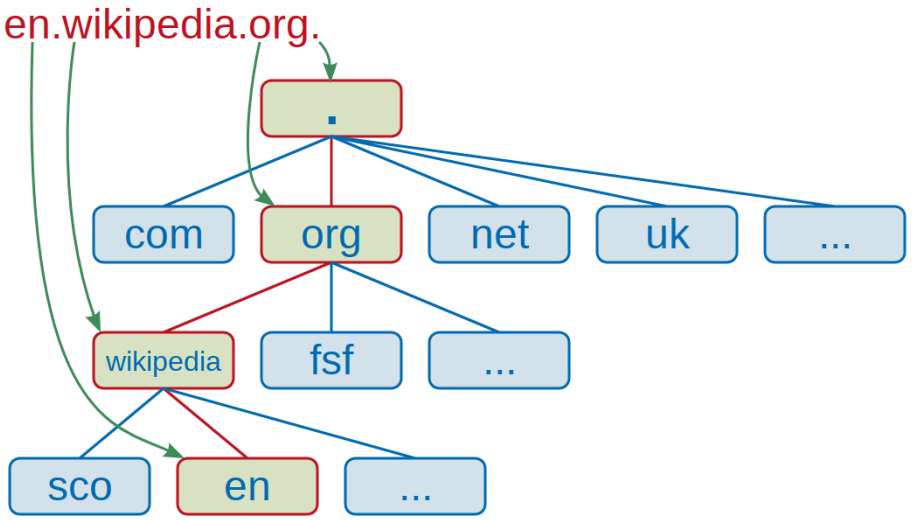
- Same as ICMPv4 but for IPv6, plus
  - (Type numbers rearranged)
- Path MTU discovery: IPv6 doesn't support fragmentation
- Multicast Listener Discovery (MLD): we've seen this
- Neighbor Discovery Protocol (NDP)
  - Replaces ARP and DHCP
  - Stateless autoconfiguration (no server needed)
    - Ask for IPv6 prefix via Router Solicitation/Advertisement
    - Host chooses its own IPv6 address suffix (from MAC or cryptography)
    - Probe the new address for duplicates via Neighbor Solicitation/Advert

Domain Name System (DNS)

- Nobody wants to memorize IP addresses
  - Tiring for IPv4, impossible for IPv6
  - We want nice textual names for Internet sites (e.g. bme.hu)
- Hierarchical, decentralized name resolution service
- Precursor: /etc/hosts file lists names for specific IP addresses
  - Still works, overrides the network query
  - On windows: c:\windows\system32\drivers\etc\hosts
  - https://someonewhocares.org/hosts/

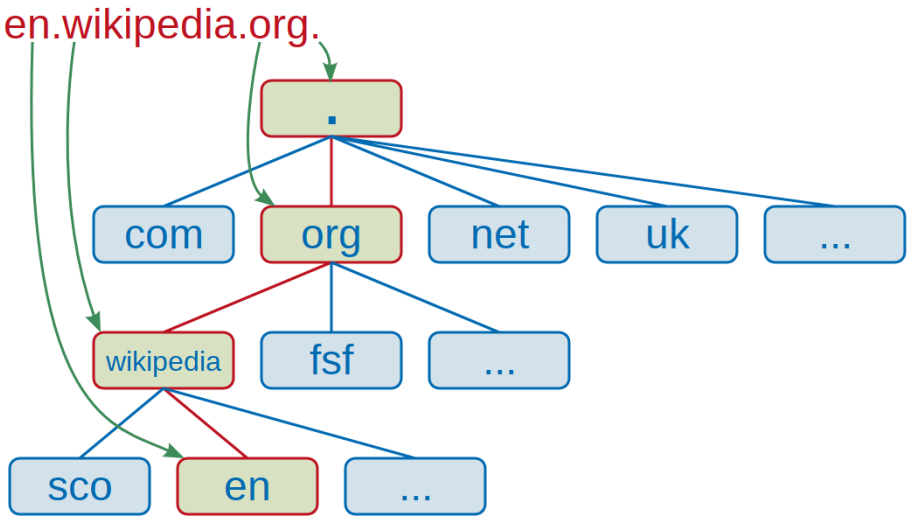
Domain Name System (DNS)

- Domain Name: an identification string
  - Hierarchical
  - Levels joined by dot
  - Uniform Resource Locator (URL)
  - Internationalized string
- Fully Qualified Domain Name: no parts omitted
- Top Level Domain (TLD)
  - Commertial
  - Organization (non-profit)
  - Network (provider)



Domain Name System (DNS)

- Lookup mechanism is also hierarchical (DNS resolution)
  - Name server responsible for a branch (e.g. *.telekom.hu)
  - Query the local server
  - Query forwarded upward until match, then downward on the other branch
  - Authoritative Name Server
  - Tools: nslookup, host
  - DNS cache



Domain Name System (DNS)

- DNS records: the result of a lookup
  - A: IPv4 address
  - AAAA: IPv6 address
  - MX: mail server
- Domain registry: for top level names (WHOIS records)
  - Handled by several competing companies (e.g. godaddy.com,
  - They own the names, we can rent from them
- Security
  - Initially nothing
  - Cache poisoning, hijacking
  - DNSSEC: signed response

Transport Layer

- We’ve seen: UDP (SOCK_DGRAM), TCP (SOCK_STREAM)
- Datagram Congestion Control Protocol (DCCP) SOCK_DCCP
  - Connection-oriented, reliable, like TCP
  - Not stream, but individual datagrams, like UDP
  - No reliable ordering
- Stream Control Transmission Protocol (SCTP) SOCK_SEQPACKET
  - Also message-based, but reliable
  - Multihoming, multipath, multistream
  - Designed for telecommunication network core links
- Transparent Inter-Process Communication (TIPC)

Application Layer

- HyperText Transfer Protocol (HTTP)
- Originally for web browsing: get files one-by-one from a server
  - Client: GET, POST methods, specifies a file name
  - Server: replies with the document the client asked for
- URL: www.bme.hu/index.html
- Versions:
  - 1.0: separate request on for each file one-by-one (1996)
  - 1.1: reuse existing TCP connection (1997)
  - 2.0: compressed metadata, multiple streams for parallel download (2015, Google)
- Nowadays everything over http: so simple we can use it to transfer anything

HTTP

- Security: https:// means we use SSL/TLS
  - TCP(TLS(HTTP(data)))
  - After TCP handshake we need a TLS handshake
  - Certificate: the public key of the server (trusted identity, encryption keys)
- Cookie
  - HTTP is stateless: every file request is independent
  - How to do something like a basket in a webshop, login to webmail?
  - Server instructs the client to store (key,value) pairs
  - Client sends these back in subsequent requests
  - Only for the domain that issued the cookie (security)