DCS Revision

Contents

Application Architectures

Network Application

- Uses a network
- Runs on two or more hosts
- Runs at the application layer
- Uses lower layer protocols

Client-server architecture

Server

- Always-on host
- Permanent IP address
- Has clients connecting to it

Client

- Communicates with a server
- May be intermittently connected
- Does not directly communicate with other clients

Terminal-host

- All processing is done on the host
- Transmission is expensive
- Slow response time

Peer-to-peer architecture

- No always-on server
- Arbitrary end systems directly communicate
- Peers are intermittently connected and change IP address
- Highly scalable

Web services

Benefits

- Offers a way to standardize interactions between objects over the internet
- Can make distributed computing far simpler once web service standards are fully developed

Concerns

- High overhead (very chatty)
- Standards immaturity

Address Resolution Protocol (ARP)

IP address

- Network layer address for interface
- Used for layer 3 (network layer) forwarding

MAC address

- Used locally to get frame from one interface to another physically connected interface

ARP

The table contains IP/MAC address mappings. Also has a time to live (TTL) after which the mapping will be forgotten.

Steps

1. A wants to send a datagram to B. But B's MAC address is not in A's ARP table.
2. A broadcasts an ARP query packet contains B's IP address. Destination MAC address is FF-FF-FF-FF-FF-FF because it is unknown.
3. B receives ARP broadcast and replies to A with its MAC address.
4. A caches B's MAC address for future use.

Border Gateway Protocol (BGP)

Hierarchical routing

- Routers are grouped into “autonomous systems” (AS).
- Routers in the same AS run the same routing protocol (the intra-AS routing protocol).
- Routers in different AS can run different intra-AS routing protocol.

Inter-AS tasks

- Each router must learn which destinations are reachable through the gateway routers
- Propagate this information to each router in the AS.

Inter-AS routing

BGP provides each AS a means to:

- Obtain subnet reachability information from neighbouring AS’.
- Propagate reachability information to all AS-internal routers.
- Determine good routes to subnets based on reachability and policy.
- Allows subnet to advertise its existence to rest of internet.

BGP sessions

- Pairs of routers (BGP peers) exchange routing information over semi-permanent TCP connections.
- When one AS advertises a prefix to another AS it is promising that it will forward datagrams towards that prefix.
- eBGP is used to distribute prefix reachability to other AS’.
- iBGP is used to distribute that information within the AS.

Path attributes and BGP routes

The advertised prefix will include BGP attributes: prefix + attributes = route.

Import policy

When gateway router receives a route advertisement, it uses an import policy to accept/decline.

Attributes

- **AS-PATH**: contains the AS’ for which *this prefix* advertisement has passed through.
- **NEXT-HOP**: indicates the next internal-AS router to the next-hop AS.

Route selection

A router may have to eliminate certain routes if it knows of more than one to get to a prefix. The criteria it could use for elimination are:

1. Local preference (e.g. set by a policy)
2. Shortest AS-PATH
3. Closest NEXT-HOP router
4. Additional criteria

BGP messages

- Exchanged using TCP.
- **OPEN**: opens TCP connection to peer and authenticates sender
- **UPDATE**: advertises new path (or withdraws an old one)

- **KEEPALIVE**: keeps connection alive in absence of UPDATE. Also acknowledges an OPEN request.
- **NOTIFICATION**: reports errors in previous message, also used to close connection.

Defining a network

Protocols

Protocols define format, order of messages sent and received among network entities, and actions taken on message transmission and receipt.

Internet Protocol stack

- **Application**: supporting network applications.
  FTP, SMTP, HTTP

- **Transport**: process-process data transfer
  TCP, UDP

- **Network**: routing of datagrams from source to destination
  IP, routing protocols

- **Link**: data transfer between neighbouring network elements
  Ethernet, 802.11 (WiFi), PPP

- **Physical**: bits “on the wire”

OSI model

- **Presentation**: allow applications to interpret meaning of data.
  Encryption, compression

- **Session**: synchronisation, check pointing, recovery of data exchange.

Sources of packet delay

Nodal processing

- Check bit errors
- Determine output link
- Typically less than 1 ms

Queueing delay

- Time waiting at output link for transmission
- Depends on congestion level of router

Transmission delay

- The amount of time it takes to get the bits on the wire
- packet length / link bandwidth

Propagation delay

- The amount of time it takes to get the bits to the destination
- length of physical link / propagation speed in medium

Link layer

- **Nodes**: hosts and routers
- **Links**: communication channels (wired links, wireless links)
- **Frame**: a layer-2 packet

- Responsible for transmitting packets from one node to another
- Performs error checking

Switch

- A **link layer** device, selectively forwards incoming frames to MAC addresses to one or more outgoing links
- Hosts are unaware of switch presence
- Remembers entries via a **switch table** which is generated when a switch receives a frame
- If it doesn't know where to forward the frame, it forwards it to everyone except the interface on which the frame arrived

Router

- Network layer
- Runs routing algorithms (RIP/OSPF/BGP)
- Forward datagrams from incoming to outgoing link

Dynamic Host Control Protocol (DHCP)

Allows a host to dynamically obtain its IP address from a network server when it joins the network.

DHCP can also return the **address of the first-hop router** for the client, the **name and IP address of the DNS server** and the **network mask**.

Discover

The client has arrived and wants an address, it broadcasts a discover packet to find out where the DHCP server is and what address it can have.

- **SRC**: 0.0.0.0:68 (no address)
- **DEST**: 255.255.255.255:67 (broadcast to all)
- **YIADDR**: 0.0.0.0
- **ID**: 654 (unique identifier for this request)

Offer

The DHCP server offers an address to the client.

- **SRC**: 223.1.2.5:67 (DHCP server)
- **DEST**: 255.255.255.255:68 (broadcast to all)
- **YIADDR**: 223.1.2.4 (the address that is available)
- **ID**: 654
- **TTL**: 3600 (seconds)

Request

Client uses the information that the DHCP server gave to send a request for that IP address. Effectively reading back the information to ensure it is correct.

- **SRC**: 0.0.0.0:68 (no address)
- **DEST**: 255.255.255.255:67 (broadcast to all)
- **YIADDR**: 223.1.2.4 (confirming the address)
- **ID**: 654
- **TTL**: 3600

Acknowledgement (ACK)

The DHCP server acknowledges the request and returns the information again.

- **SRC**: 223.1.2.5:67 (DHCP server)
- **DEST**: 255.255.255.255:68 (broadcast to all)
- **YIADDR**: 223.1.2.4 (the address that is available)
- **ID**: 654
- **TTL**: 3600 (seconds)

Domain Name System (DNS)

- Uses UDP
- Once a name server learns a mapping, it is cached until the TTL expires

Client wants IP for **mail.google.com**:

1. Client queries a **root server** to find com DNS server
2. Client queries **com** DNS server to find google.com DNS server
3. Client queries **google.com** DNS server to find mail.google.com IP address

Iterated query

The local name server is responsible for asking all the other name servers for the resolution. If the server doesn't know the exact name, it will pass a **recommendation** to the local DNS server.

Recursive query

The contacted name server is responsible for asking all the other name servers for the resolution. If the server doesn't know the exact name, it will look ask another name server itself.

DNS record types

DNS is a distributed database storing resource records in the format: **name, value, type, ttl**.

A

name = the host name
value = the host IP address

CNAME

name = the alias for some canonical (real) name
value = the canonical name

NS

name = the domain
value = the hostname of the authoritive name server for this domain

MX

name = the name of the mail server
value = the location of the mail server

Hypertext Transfer Protocol (HTTP)

Request

METHOD URL VERSION\r\n
HEADERS\r\n
\r\n

e.g.

GET /test.html HTTP/1.1\r\n
Host: www.google.com\r\n
\r\n

Response

VERSION STATUS PHRASE\r\n
HEADERS\r\n
\r\n

e.g.

HTTP/1.1 200 OK\r\n
Server: Apache/2.2 (Ubuntu)\r\n
\r\n

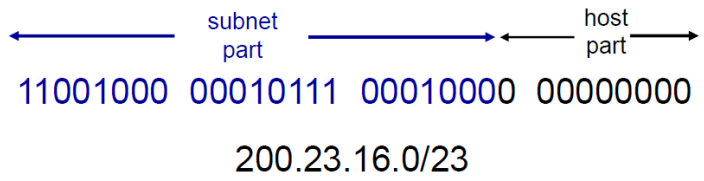
Internet Protocol (IP)

In the network layer

Classless Interdomain Routing (CIDR)

Subnet portion of address of arbitrary length.

Address format: **a.b.c.d/x** where x is the number of bits in the subnet portion of the address.



Internet Corporation for Assigned Names and Numbers (ICANN)

- Allocates addresses
- Manages DNS
- Assigns domain names, resolves disputes

Internet Control Message Protocol (ICMP)

Used by hosts and routers to communicate network-level information.

An ICMP message contains the **type**, **code** and the **first 8 bytes** of the IP datagram causing the error.

Traceroute

Source sends a series of UDP segments with increasing TTL. The TTL is decreased after every hop, so eventually it will reach 0.

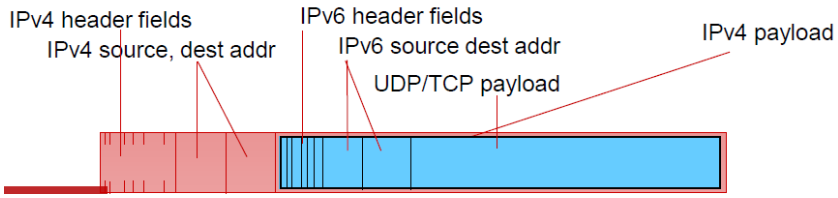
When the TTL expires, the destination will send back a “TTL expired” message that contains the routers name and IP address.

IPv6

128-bit addresses

Tunneling

IPv6 datagrams carried as payload in an IPv4 datagram



Subnets

Used to break the network into smaller, more efficient subnets to prevent excessive rates of Ethernet packet collision in a large network. Routers are used to manage traffic and constitute borders between subnets.

A subnet is device interfaces that have the same subnet part of the IP address. They can physically reach each other without an intervening router.

Subnet masks

Subnet mask quick reference							
Host Bit length	math	Max hosts	Subnet mask	Mask octet	Binary mask	Mask length	Subnet length
0	2^0=	1	255.255.255.255	4	11111111	32	0
1	2^1=	2	255.255.255.254	4	11111110	31	1
2	2^2=	4	255.255.255.252	4	11111100	30	2
3	2^3=	8	255.255.255.248	4	11111000	29	3
4	2^4=	16	255.255.255.240	4	11110000	28	4
5	2^5=	32	255.255.255.224	4	11100000	27	5
6	2^6=	64	255.255.255.192	4	11000000	26	6
7	2^7=	128	255.255.255.128	4	10000000	25	7
8	2^8=	256	255.255.255.0	3	11111111	24	8
9	2^9=	512	255.255.254.0	3	11111110	23	9
10	2^10=	1024	255.255.252.0	3	11111100	22	10
11	2^11=	2048	255.255.248.0	3	11111000	21	11
12	2^12=	4096	255.255.240.0	3	11110000	20	12
13	2^13=	8192	255.255.224.0	3	11100000	19	13
14	2^14=	16384	255.255.192.0	3	11000000	18	14
15	2^15=	32768	255.255.128.0	3	10000000	17	15
16	2^16=	65536	255.255.0.0	2	11111111	16	16
17	2^17=	131072	255.254.0.0	2	11111110	15	17
18	2^18=	262144	255.252.0.0	2	11111100	14	18
19	2^19=	524288	255.248.0.0	2	11111000	13	19
20	2^20=	1048576	255.240.0.0	2	11110000	12	20
21	2^21=	2097152	255.224.0.0	2	11100000	11	21
22	2^22=	4194304	255.192.0.0	2	11000000	10	22
23	2^23=	8388608	255.128.0.0	2	10000000	9	23
24	2^24=	16777216	255.0.0.0	1	11111111	8	24

Java

```java
/*
    TCP SERVER
*/
ServerSocket srv = new ServerSocket(1234);
Socket client = srv.accept();

BufferedReader inFromClient = new BufferedReader(new InputStreamReader(client.getInputStream()));
DataOutputStream outToClient = new DataOutputStream(client.getOutputStream());

String clientMessage = inFromClient.readLine();
outToClient.write(clientMessage.toUpper().getBytes());

/*
    TCP CLIENT
*/
Socket soc = new Socket("1.2.3.4", 1234);

DataOutputStream outToClient = new DataOutputStream(soc.getOutputStream());
outToClient.write("this is a message".getBytes());

BufferedReader inFromClient = new BufferedReader(new InputStreamReader(connectionSocket.getInputStream()));
System.out.println(inFromClient.readLine());


/*
    UDP SERVER
*/
DatagramSocket soc = new DatagramSocket(3000);
byte[] data = new byte[8000];
DatagramPacket dp = new DatagramPacket(data,data.length);
soc.receive(dp);
String receivedString = new String(dp.getData());


/*
    UDP CLIENT
*/
DatagramSocket soc = new DatagramSocket();
byte[] data = "return message".getBytes();
DatagramPacket sendPacket =  new DatagramPacket(data, data.length, "1.2.3.4", 3000);
soc.send(sendPacket);

```

Network Address Translation (NAT)

- Just one IP address for each household
- Can change addresses of devices in local network without notifying outside world

Outgoing datagrams

Replace source IP address and port number of every outgoing datagram to the NAT IP address and new port number.

Add this mapping to the NAT translation table.

Incoming datagrams

Replace the incoming destination address and port number with the original source IP address and port number from the NAT translation table.

Network Performance

Quality of Service (QoS)

An indicator of network performance.

Metrics

- **Availability**: the percentage of time a network is available for use.
- **Error rate**: the percentage of packets/bits that are lost or damaged.
- **Latency**: delivery delay, measured in milliseconds.
- **Jitter**: variation in latency between successive packets.

Network Security

- **Confidentiality**: only sender, intended receiver should understand message contents (to prevent man-in-the-middle).
- **Authentication**: sender, receiver want to confirm identity of each other (easy to spoof IP, MAC).
- **Message integrity**: sender, receiver want to ensure messages not altered (in transit or afterwards) without detection.
- **Access and availability**: services must be accessible and available to users (denial of service).

Open Shortest Path First (OSPF)

Routing algorithms

Global

- All routers have a complete knowledge of topology
- They have the link cost knowledge that all other routers have
- A **link state algorithm** uses global knowledge.

Decentralised

- Router knows physically-connected neighbours and link costs to its neighbours
- Information is exchanged with neighbours
- A **distance vector algorithm** only uses knowledge acquired from directly connected neighbours.

Static

- Routes change slowly over time
- Updated by manually entering new route information at router.

Dynamic

- Routes change more quickly
- Updated in response to link cost changes.

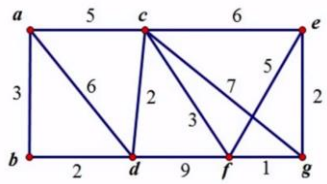
Dijkstra's algorithm

- Shows the net topology, link costs known to all nodes. This is accomplished via a link state broadcast. All nodes have the same information.
- Computes the least cost paths from one node to all other nodes. This is used to update the **forwarding table** for the node running the algorithm.

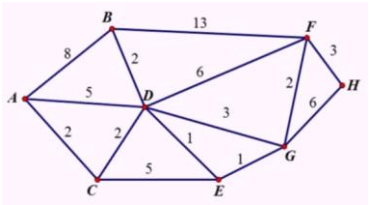
Notation

- $\boldsymbol{c(x, y)}$: link cost from node *x* to *y*. ∞ if not direct neighbours.
- $\boldsymbol{D(v)}$: current value of cost path from source to destination (v).
- $\boldsymbol{p(v)}$: predecessor node along path from source to v. For example if the path is a-b-c-d, then c is the predecessor to d.
- $\boldsymbol{N'}$: set of nodes for which the least cost path is known.

How it works



	V	a	b	c	d	e	f	g
a	0	3_a	5_a	6_a	∞	∞	∞
b	0	3_a	5_a	5_b	∞	∞	∞
c	0	3_a	5_a	5_b	11_c	8_c	12_c
d	0	3_a	5_a	5_b	11_c	8_c	12_c
f	0	3_a	5_a	5_b	11_c	8_c	9_f



v	a	b	c	d	e	f	g	h
a	0	8_a	2_a	5_a	∞	∞	∞	∞
c	0	8_a	2_a	4_c	5_c	∞	∞	∞
d	0	6_d	2_a	4_c	5_d	10_d	7_d	∞
e	0	6_d	2_a	4_c	5_d	10_d	6_e	∞
b	0	6_d	2_a	4_c	5_d	10_d	6_e	∞
g	0	6_d	2_a	4_c	5_d	8_g	6_e	12_g
f	0	6_d	2_a	4_c	5_d	8_g	6_e	11_f
h	0	6_d	2_a	4_c	5_d	8_g	6_e	11_f

Advertisements

- Each router advertises its knowledge of costs to neighbouring routers.
- Advertisements are disseminated to the entire AS via flooding
- Carried in OSPF messages directly over IP
- Broadcasts occur when link state changes.

Hierarchical OSPF

- **Area border routers**: summarise distances to nets in own area, advertise to other area border routers.
- **Backbone routers**: run OSPF routing limited to backbone.
- **Boundary routers**: connect to other AS'.

Comparison of Link State and Distance Vector algorithms

Link state

- With n nodes, e links O(nE) messages sent
- Node can advertise incorrect link cost
- Each node only computes its own table
- $O(n^2)$ algorithm requires O(nE) messages.

Distance vector

- Exchange between neighbours only
- Each DV node can advertise incorrect path cost
- Each node's table used by others, error propagates through network
- May be routing loops
- Count to infinity problem.

Advanced OSPF features

- **Security**: all messages authenticated.
- **Multiple same-cost paths** allowed.
- Supports **hierarchical OSPF** in large domains.

Routing Information Protocol (RIP)
//////////////////////

Email

Simple Mail Transfer Protocol (SMTP)

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250  Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```

Post Office Protocol (POP3)

Deletes emails off server when read

IMAP

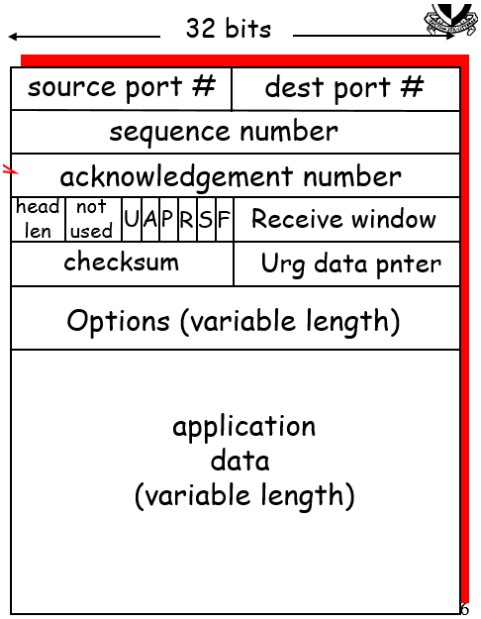
Doesn't

Transmission Control Protocol (TCP)

TCP is point-to-point, it uses a reliable, in-order byte stream, it is pipelined, it has send and receive buggers, it allows bi-directional data flow in the same connection (duplex), it is connection-oriented and it is flow controlled (sender will not overwhelm receiver).

TCP is reliable because it has cumulative acknowledgements and it retransmits data.

Header



- U:
- A: ACK = acknowledged
- P:
- R:
- S: SYN = data is in a sequence
- F: finished

Sequence Numbers

Both switch and increment ACK

Initial value for SEQ is the first byte of the stream

ACK = sequence number of next expected byte

// how used

// when and why acknowledgement is exchanged

Time out

Calculating the time out

Estimated round trip time + safety margin

Larger variation in estimated RTT = larger safety margin

// how calculated

// why important

// when change

How recovers from lost segments

Fast retransmit

If the transmission times out (lost ACK)

Duplicate ACK (x3) tells us that segments may be lost

// why needed

// when it would be used

Flow control

ReceiveWindow = ReceiveBuffer – (LastByteReceived – LastByteRead)

Receiver tells the sender about how much it can possibly take in. It does this by checking how many bytes were unread after the last datagram.

// what it is

// how achieved

// benefits

Connections

This is necessary to initialise TCP variables (sequence numbers, buffers, flow control info).

Client

1. **CLOSED**
2. **SYN_SENT**
3. **ESTABLISHED**: receive syn and ack, send ack
4. **FIN_WAIT_1**: after sending FIN
5. **FIN_WAIT_2**: receive ACK, send nothing
6. **TIME_WAIT**: after receive FIN, send ACK
7. Wait 30 seconds then go to 1.

Server

1. **CLOSED**
2. **LISTEN**
3. **SYN_RCVD**: after receive SYN and rend SYN and ACK
4. **ESTABLISHED**: after receive ACK
5. **CLOSE_WAIT**: after receive fin, send ack
6. **LAST_ACK**: after send find
7. Receive ACK then go to 1.

Congestion

Too many sources sending too much data too fast for the network to handle.

Lost packets (buffer overflows at routers).
Long delays (queueing in router buffers).

Maximum Segment Size: maximum transmission unit + the segment header length.

Rate of transmission

Rate of transmission = congwin/RTT bytes per second.

Congestion Window

The number of bytes that can be acknowledged at any time.

AIMD = additive increase, multiplicative decrease

Increase congwin by one MSS (maximum segment size) every RTT (round trip time) until loss detected. Cut congwin in half after loss

Slow start

Start with congwin=1, increase rate exponentially fast until first loss event

Carrier Sense Multiple Access (CSMA)

Listen before transmit.

If channel sensed idle: transmit entire frame, if it's busy – defer the transmission.

Collisions can still occur because one node may not hear another node's transmission until it is too late.

CSMA/CD (collision detection)

Collisions detected within a short time. Colliding transmissions aborted, reducing channel wastage.

Easy in wired LAN: measure signal strengths, compare transmitted and received signals.

Difficult in wireless LAN: received signal strength overwhelmed by local transmission strength.

1. NIC received datagram from network layer, creates frame
2. If NIC senses channel idle, starts frame transmission. If channel is busy, waits until channel is idle then transmits
3. If NIC transmits entire frame without detecting another transmission, NIC is done with frame.
4. If NIC detects another transmission, aborts and sends JAM signal.
5. After aborting, NIC enters binary exponential backoff (waits longer backoff interval with more collisions).

Ethernet (802.3)

Header

- Preamble: a series of 7 octets going 10101010 (to allow synchronisation signal for the receiver's clock).
- Start of frame delimiter field (ends in 11 instead of 10).
- Destination and source MAC address (48 bits each)
- Length of data field
- Data field
    - LLC subheader that describes the contents of the data field
    - The packet, usually an IP packet
- Padding to get it up to 46 octets
- Trailer
    - Frame check sequence, calculates a number on each end and compares them, if it is different then the frame is dropped.

User Datagram Protocol (UDP)

- Connectionless
- UDP segments may be lost

Virtual Local Area Network (VLAN)

Port-based VLAN

- **Traffic isolation**: frames to/from port 1-8 can only reach ports 1-8.
- **Dynamic membership**: ports can be dynamically assigned among VLANs.
- **Forwarding between VLANs**: done via routing (just as with separate switches).

Wireless Protocols and Wireless Security

Infrastructure mode: wireless hosts that are associated with a base station
Ad hoc: no base station

Request To Send (RTS) and Clear To Send (CTS)

This reduces frame collisions with the hidden node problem. It requires that all the wireless nodes first request permission to send their data.

Wireless Link Characteristics

- **Decreased signal strength**: radio signal attenuates as it propagates through matter (path loss)
- **Interference from other sources**: frequencies are shared by other devices too (e.g. phone).
- **Multipath propagation**: radio signals reflects off objects, ground, arriving at destination at slightly different times.

Code Division Multiple Access (CDMA)

All users share same frequency, but each user has their own "chipping" code.

Encoded signal = original data XOR chipping sequence

Decoding = inner product of encoded signal and chipping sequence.

Allows multiple users to coexist and transmit simultaneously with minimal interference

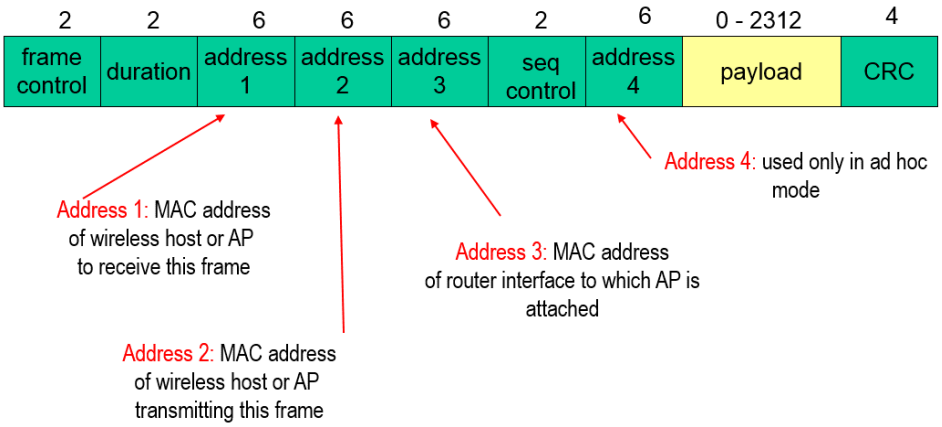
Passive scanning

1. Beacon frames sent from Aps
2. Association request frame sent from host to selected AP.
3. Response frame sent from AP to host.

Active scanning

1. Probe request frame broadcast from host
2. Probe response frames sent from APs
3. Association request frame sent host to select AP
4. Association response frame sent AP to selected host

802.11 frame



802.15 Personal Area Network

Ad-hoc network, with master and slaves

Evolved from bluetooth