Network and Cybersecurity

Kristoffer Klokker

2022

Contents

1 Computer networks

A system connected to the internet is called a **host / end system**. This is done thorugh a **communication link** and **packet switch**. By this it has a **transmission rate** which describe the speed measured in bits/second.
When data is send it is done through **packets** which consist of a data header and including data. A packet is send first to the packet switch often taking form of a router or link-layer-switch. Which then sends it further to the communication link.
ISP then interconnect the communication links and connects to a higher up ISP or between higher up IPS. The highest tier is tier 1 of which other ISP connect with, even multiple such a **multi-home** setup is done in case of failure. An ISP may alos connect to a **PoP** which simply are a group of routers from another ISP.
ISPS may also peer between eachother to reduce exhange with higher tier ISP and reduce cost. Most often are peering free and third party companies may even create a IXP where multiple ISP can connect and peer to each other.
Big companies like Google also has their own private network infrastructure, which spans the globe between their datacenters. They are therefore peering between same tier ISPs and are also connected to 1 tier ISP but with reduced exhange.
Edge network refers to the connections to all the end host.

1.1 Different types of connections

Home access can be done in different ways:

- DSL - Digital subscriber line is done through a telephone lines in higher frequencies such phone and network can coincide. Current max limit of downstream is 1 Gbps, but the whole system require small distance to the telco provider 8 - 16 km.

- Cable internet access - Internet done through the existing television coax cable or hybrid fiber coax (HFC), a modem is then used to translate the analog signal to digital signal. This allows up to 1.2 Gbps downstream.

- FTTH - Fiber to the home is a direct connection from the cable office to the home. Here are two types active optical network (AON) and passive optical network (PON), PON works such a netowk teminator at the home goes through a splitter between up to 100 homes into the terminator at the telco.

- 5G fixed wireless - This is a wireless solution which are beginning to gather popularity in cities.

At the home a Local Access Network is then created, and most often extended to a Wireless Local Access Network using a router usin the WiFi protocol.

1.2 Mediums of signals

When a signal is sent through a physical media it is called guided media whereas over a wireless media it is called unguided.
Example of mediums are:

- Twisted cobber wire - used for telephone connections an still in use, for ethernet and more. A communication link done with this is called unshielded twisted pair (UTP). Modern cables of category 6a can transfer up to 10 Gbps over hundreds of meters.

- Coaxial cable - much like the twisted cobber cable, but the two copper condctors work concentric rather than parallel.

- Fiber Optics - Works by sending light insted of electricity, can have speeds in up to hundreds of Gbps over distances up to 100km due to no electromagnetic interference.

- Terrestrial Radio Channels - radio signals which characteristics depend a lot on enviremental factors. Can be found in three categories, short distance up to 2 meters, medium distance up to a few hundred meters and long distance spanning tens of kilometers.

- Satellite Radio Channels - Works as a transmitter receiver between two points on earth. Either in a geostationary orbit or low-earth orbit. geostationary eliminate the need of always finding the optimal satelite but at consequence of 280 ms delay.

1.3 Packet switching

Most packet switches use **store-and-forward transmission**, such before forwarding a transmittion all packages has to be obtained and proccessed before transmissioning.
This therefore extend the tranmission time according to package number and

speed.
When the router has processed all packages they can be send as one unit.
This therefore can be described with:

$$d_{end-to-end} = N\frac{L}{R}$$

Where N is number of links, L is number of bits in a package and R is the speed.

In case of a package is alreadyu being sent the new package is sent to the **output buffer**, and in case of a full buffer **packet loss** will occur.
A packet switch most often have multiple queues for different forwarding packages. The packages are placed based on a forwarding map, using the packages IP addres.

1.4 Circuit switching

A circuit switch networks works on reservations. Instead if queues, a reservation is made to the receiver in both bandwith and that a port is open for the package.
This also has the advantage that the number of links can be ignored, in transmission time.
In this way a package can be guranteed to come at a specific time and no lost of any package.
To route more connections either **frequency-division multiplexing** or **time-division multiplexing** can be used.
FDM splits the avaliable frequecies into smaller bandwith of which data can be sent through.
TDM splits up the full bandwith into frames which are reserved with a number a number of slots for data. Therefore the speed is dependent on frame rate multipled by number of bits in a slot.
The downside of circuit switching compared to package switching, is the need for allocated sapce for every user, even in inactivity whereas package switching can work with changing demand.

1.5 Delay

There are different kind og delays the most important are:

- Nodal processing delay - delays in the router such as determine package header or bit error correction (μs)

- Queuing delay - delays in case of heavy traffic and non empty buffers (μs - ms)
- Transmission delay - delays that occur when waiting for a whole package and transmission rate is low (μs - ms) - $d_{trans} = \frac{L}{R}$
- Propagation delay - delays from the actual bit transportation limit by medium (ms) - $d_{prop} = \frac{m}{s}$

These in total are the total nodal dealy.

1.6 Protocols

A protocol defines the format and the order of messages exchanged between two or more communicating entities, as well as the actions taken on the transmission and/or receipt of a message or other event.

Protocols are therefore used everywhere, such everything works as expected in between the layers of the network.
When different layers of protocols work together it is called a protocol stack, this may be for the internet:

- Application - These are protocols which are sent pr apllication such as HTTP, SMTP, DNS and such, refered as message
- Transport - These protocols are for sending application messages, here TCP and UDP, refered as segment
- Network - Protocols for distributing the segments using IP protocols or other routing protocols therefore refered as IP layer
- Link - Protocols here are used for the IP layer to transmit to until reaching the physical layer, this is protocols such as Ethernet and WiFi, refered as frames
- Physical - Protocols for transmitting frames, here protocols depend on medium such as for fiber

When data goes through each layer, a new header is added and previus layer data is used as payload.

1.7 Network attacks

Some of the different kind of internet attacks include:
Denial of service attacks (DoS) which prevent access to network host or infrastructure.
It can work by 3 ways:

- Vulnerability attack - Few messages which use a exploit such the host either stops or crashes
- Bandwith flooding - A large amount of packages is sent to the host resulting in overfilled buffers
- Connection flooding - A large number of open TCP connections are made such every entry is occupied

For flooding attacks they may also be a distributed Dos (DDoS) attack thorugh multiple devices often in a botnet.

A packet sniffer is a type of passive attack of which a device will listen for packages and possibly find sensitive information in packages sent through it.
IP Spoofing is a package which source address is spoofed, and therefore faked to look as a different source.

2 Protocol layers

For an application communicates to network layer, a socket is setup, which indentifies by a port. An app-layer protocol defines

- Messages exchanges - requst, resonse,...
- Message syntax - what fields are in a message
- Message sematics - Meaning of fields
- Rules for send and reponds

Transport service requirements - The different requirements an application may set for the transport of data
The different types of requirements an application may set

- Data loss - May be strictly no loss or just some percentage
- Throughput - May need a constant throughput or can be elastic

- Time sensitive - May need a minimum latency for data
- Security - Some data may be more sensitive than other

2.1 Client server architecture

The server side is on a always on host, with permanent ip, which handles backend stuff.
Ex. are hosting api and database.
The client in this architecture communicates via HTTP protocols are some alike.
The clients are the dynamic part, may not be on same ip, or connect with each other

2.2 P2P

No host involved but clients communicating directly between each other.
Are scaled based on number of users.

2.3 TCP Service

Reliable transport with flow control to not overwhelm receiver
Able to throttle sender in an overloaded network.
Does not provide: timing, min. througput guarantee or security
Need a setup between client and server established before use.

2.4 UDP

Unreliable data transfer.
Does not provide: reliability, flow control, congestion control, timing, throughput guarantee, security
Smarter for data which can handler larger loss.
TCP may get stuck on a packet whereas UDP would simply skip the package and go forward for the next package.
UDP is also faster and connectionless therefore making it possible to handle more connections faster.
The header includes 4 16 bits data: sorce port, destination port, length of data, checksum
The checksum works by the data is split up into 16 bit words, then they are summed and overflow is wrapped around. Then the sum is converted to 1s

compliment (everything is flipped). This will then be the checksum.
If the checksum fails the data is either discarded or delivered with an error.

2.5 Domain name system

DNS binds name/string to an ip address.
This also includes alias names for mail server or sub domains.
Usefull for more dynamic ip setup and readability
Implemented via a distributed database.
The databases stores resource records (RR) in the format (name,value,type,ttl)
The DNS is implemented in a tree structure.
First are the root DNS which contains addresses of top level domain severs like .com, .org, .dk
They then contain then the DNS servers for domain such as google.com, which itself contains addresses for subdomain.
So when a request is made first, the cache is checked, then a request to a preinstalled root such as 1.1.1.1 is made.
This then returns a top level address for a DNS server, which a request is made to.
This is repeated until an ip matching the request is returned.
This is the itterative method.
The recursive query find the ip via the DNS servers which contacts eachother putting the burden on them
Local DNS servers does not belong to hierarchy but is hosted by an ISP.
TTL (time-to-live) is a numerical value representing the amount of server hops a packet can make before being outdated.
The local DNS handles the requesting and caching for an IP to the hiarachy DNS servers.

2.5.1 Types of resource records

To do a lookup the dig tool can be used

- Type=A - name: hostname, value: ip-address
- Type=NS - name: domain, value: nameserver
- Type=CNAME - name: alias, value: hostname
- Type=MX: name: '@', value: mail server

2.5.2 DNS protocol

Query and reply are in the form Message, header
The header consist 12 bytes dedicated to

- 16 bit identificaiton
- 16 bit flags
  - Query (0) or reply (1)
  - 4 bit opcode: standard query (0), domain name form ip (1), status request of server (2), (3) is reserved for status an not used
  - AA: The server is authoritative (1), non authoritative/cache (2)
  - TC: the message exceeds 512 bytes and are truncated (1)
  - RD: Recursion desired (1)
  - RA: Recursion avaliable from server (1)
  - Zero: 3 bits of zeros reserved
  - rCode: Respnse code, no error (0), format error (1), server failure (2), did not find name (3), request is supported (4), policy denies execution of query (5)
- 16 bit Number of questions in body
- 16 bit Number of answers RRs and is 0 from client and set by server
- 16 bit Number of authority RRs and is likewise 0 from client
- 16 bit Number of additional RRs

The body then cosist of

- Questions - query from client
- Answers RRs- Response to query from non authority
- Authority RRs - Response to query from authority
- Additional information

2.5.3 Security

A person could bombard the DNS servers with traffic and deny other trafffic in form of DDoS.
Not successful to date on root server but TLD (top-level domain) has successed
NXDOMAIN attack is requesting non existing domains and spending the DNS server ressources to find non valid addresses.
Random subdomain attack are like NXDOMAIN attack but with subdomains to target the namespace rather than root or TLD.
Phantom domain attack is setting up DNS servers which does not respond or very slow responses, such in a recursive lookup the TLD will have to wait for response.
TCP SYN is the attack of which a bunch of TCP request are opened but never used.
DNS domains lock-up is an extended TCP SYN attack where after a connection is established random packages is sent to the server, and the server will wait for a correct response.
DNS rebinding attack is used to get past browsers same-origin policy, this is done by first the user lands on a shady website, the website then make a request to itself, but the dns record is updated to point at a new site which the script now can be run upon.
DNS cache poisoning is where an attacker imposes as a nameserver, and then creates a request for the nameserver and answers before the real nameserver and thereby creating a fake lookup in the cache.

3 Web and HTTP

3.1 HTTP

A HTTP request is sent by the client, and server sends using HTTP protocol an object in response
The request is sent at port 80 using a TCP request.
Non-persistent HTTP sendt a single object and then closes
Persisten HTTP can send multiple files between client and server
Repsonse time RTT is the time for a small packet to travel from client to server and back.
Persistent has longer open connection but every referenced object can be sent at as little as one RTT
Non-persisten requires 2 RTT atleast pr referenced object, and looses alot of time to OS overhead for each established connection.

The general request consist of: method, url, protocol, headers
The general response consit of: protocol, status code, status phrase, headers, data
There is 4 method types for HTTP/1.1

- GET - get ressource
- POST - send ressource
- HEAD - meta data to check for updates
- PUT - Uploads file in entity body to url field
- DELETE - delete file in the url field

The most common response status codes

- 200 OK
- 301 Moved Permanently - object is moved to new location given in message
- 400 Bad Request - not understood by server
- 404 Not Found - Fil not found on server
- 505 HTTP version not supported
- 418 I'm a teapot - When a teapot is requested to bre coffee

3.2 Cookies

Cookies are the solution for http being stateless.
Cookies allows to store files in the browser of the user.
This can be used for saved storage like shopping cart or authority like a session cookie.

3.3 HTTP/2.0

Problem in 1.0 was request was treated in order, 1.1 made it a little better using pielining which allowed for multiple sequential request.
2.0 introduced streams, where request are numbered in odds and responses are given in even numbers.

3.4 Electronic mail

Electronic mail consist of 3 major components: User agents, Mail servers and Simple mail transfer protocol (SMTP)
User agents are essentially mail clients which allows for creation and reading of emails.
Mail servers have a mailbox for incomming messages and a message queue for outgoing mails
SMTP uses TCP on port 25 to transfer emails.
The protocol work on command response, where command are in ASCII 7 bit and reponse are status codes.
The mail message consist of header (To, from, subject), blank line and body (Only ASCII)
The user agents access the mail server via the IMAP protocol or HTTP

3.5 Video streaming and content delivery networks

To reduce the amount of data, coding is used on the data, spatial (groups pixels together) and temporal (only send difference of video frames)
CBR - constant bit rate
VBR - variable bit rate
DASH - Dynamic, Adaptive Streaming over HTTP
DASH divides video files into chunks, with each having different rates.
All chunks are managed in the manifest file which provides URL to each chunk.
The client then handles, when to get chunks, at what encoding rate and which server to request chunks.
CDN networks work by distributing the content to multiple servers around the globe.
To find the best server in a CDN preiod tests are made in the network of which speeds are in between each server.

3.6 P2P

No need for an always on server
End systems directly communicate.
Is more scaleable than a server setup.
For n clients on a server the sever time will be $D_{Client-Server} > max(NF/u_s, F/d_{min})$, it will therefore scale linearly.

Whereas in P2P the time will be $D_{P2P} > max(F/u_s, F/d_{min}, NF/(U_s + \sum u_i))$
Where U_s is the central server speed, u_i is user upload speed, d_i user download speed and F is filesize.
A torrent is a group of peers exchaning files divided into chunks of 256Kb.
The peers are then managed in a tracker, which also participate in the torrent.
A torrent client then uses those peers, request which chunks they have and missing chunks are downloaded from fastest connection.

4 Network security

The properties of secure communication

- Confidentiaity - Only sender and reciever should be able to read the message using encryption
- Message integrity - The message should not be altered by intent or accident which is checked using check sum
- End-point authentication - Insuring sender and reciever are able to confirm their idenitity
- Operational security - A firewall or depp packet inspection may be setup between local network and public network to counter malicious attacks

4.1 Principles of crypthography

A text is first plain text and after using the encryption algorithm it becomes a ciphertext.
The most used method used today are based on keys between sender an receiver.
The a symmetric key system both sender and receiver has to same key to decrypt an encrypt.
In a public key system sender and receiver has 2 keys a public K^+ and private K^-. The sender then encrypts using the public receiver key which then can be decrypted by the receiver using the private key.

4.2 RSA

RSA is an algorithm to implement public key system.
To generate a public and private key the following is done

- Two large primes are choosen to larger to higher encryption though longer de,- and encryption times, but the product should be in the order of 1024 bits
- Computer $n = pq$ and $z = (-1)(q-1)$
- A number e is choosen which is $e < n$ and $gcd(e, n) = 1$
- Find a number d such $ed - 1 \mod z = 1$
- $K^+ = (n, e)$ and $K^- = (n, d)$

The encryption c is then done by

$$c = m^e \mod n$$

where $m < n$
The decryption is then done by

$$m = c^d \mod n$$

This does take a lot of data and time therefore often sessions keys are generated and shared using RSA.
This works by when a message is encrypted and decrypted it is expressed as

$$\begin{aligned} m &= (m^e \mod n)^d \mod n && (1) \\ &= m^{ed} \mod n && (2) \\ &= m^{ed \mod z} \mod n && (3) \\ &= m^1 \mod n && (4) \\ &= m && (5) \end{aligned}$$

(2) can be done as proberty of modulo. (3) is possible because be definition p and q are prime and $n = pq$ and $z = (p-1)(q-1)$ makes it such $x^y \bmod n = x^{y \bmod x}$.
(4) can be done due to z was choosen by definition to be $ed \mod z = 1$. (5) is done since $m < n$.
Since e and d are multiplied we would get the same result if the message is encrypted or decrypted first.

The safety of RSA relies on there is no known fast factoring of number alogirthm

4.3 Symmetric key encryptions

- Caesar cipher - Every letter is shifted a number in the alphabet with wrap araound
- Monalphbetic cipher - Every letter is mapped to another letter
- Polyalphabetic - Using multiple monoalphabetic ciphers in a certain order
- Block cipher (ECB) - The message is split up into blocks of size k which then is mapped to an encrypted block in same size
- Advanced block cipher (OFB) - Split message up to smaller block of size k_1 and then again to size k_2, the k_2 is then scrambled in place. This is repeated n times
- Cipher block chaining (CFB) - First a Initization vector is genereated and send, then every block is XOR'ed with the last send message and encrypted using shared key.

To attack a simple encryption there are different methods

- Ciphertext-only attack - Can be attacked using statistical analysis
- Known-plaintext attack - If some of the plaintext is known and therefore can be matched to the encryption
- Chosen-plaintext attack - If the attacker has access to the plaintext and can therefore get information about the encryption

4.3.1 Feistel cipher

The cipher works by

1. Plaintext is splitted into 2 chunks left and right
2. A function is then performed on the right using key 1
3. The left chunk is then XOR'ed with the output of the function
4. The XOR output is now the right chunk and the original right chunk is the left chunk
5. 2 - 3 is repeated n times with the last chunk as input and the key is itterated

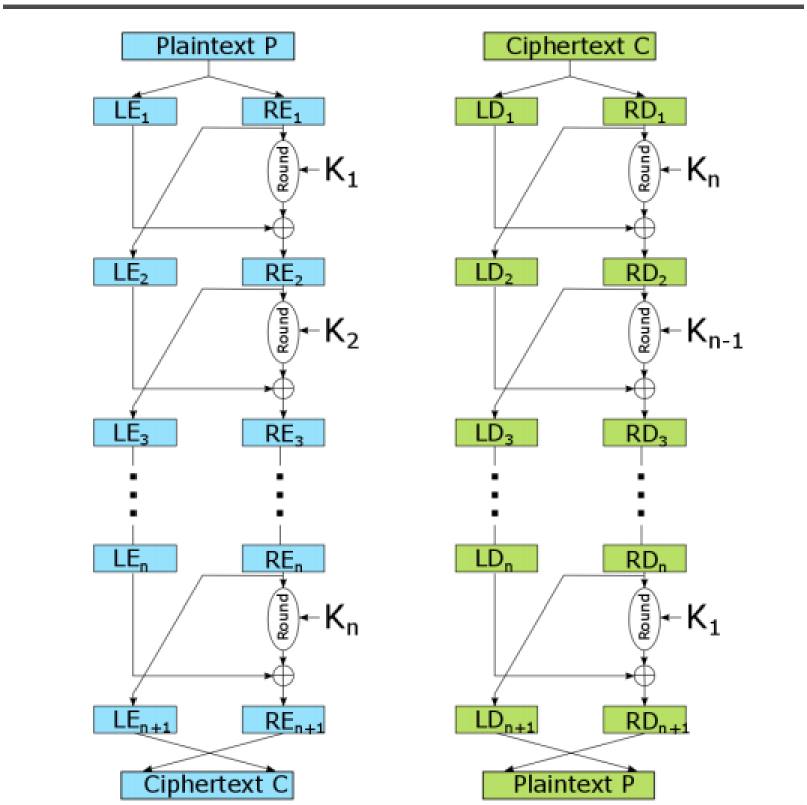


Figure 1: Feistel cipher illustration by terodee

6. Left and right are switched

To then decrypt the same algorithm is used with the reverse order of the keys.
This will work due to XOR being reversible

4.3.2 DES

A symmetric encryption system, such input and output are the same size
First the plaintext is divided into 64 bit chunks.
A 64 bit key is generated.
Then an inital permutation is done on the 64 bit text using a predetermined vector of value mappings.
A feistel cipher is then done with 16 rounds, where the function is defined as:

1. The 32 bit input is expanded to 48 bit

2. The 48 bit is XOR'ed using the input key
3. The 48 bit is then substituted using the S-Box table to 32 bits. This is done by splitting the 48 bits into 6 bits chunks. Then bit 1 and 6 is the row number and 2,3,4 and 5 is the column number.
4. The 32 bits are then permutated using the P-table.

After the feistel cipher, a final permutation is done by with the inverse of the first permutation.

To find the subkeys for each of the 16 rounds the following is done to the key.
First the key is reduced to 56 using permuted choice 1 table, where two chunks of 28 is created is also divided.
Each chunk is then circular left shifted once on itteration 1,2,9 and 16 and shifted twice on every other itteration.
Then the second permutation table is used to generate the 48 bit sub key.
By this method a bit will be used in 14 out of the 16 keys.

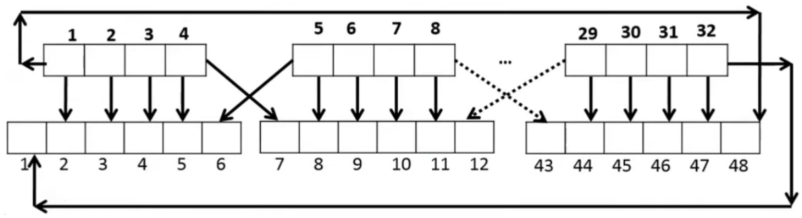


Figure 2: Des expansion visulization

Column Number

Row No.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

P

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

Figure 3: Des S-Box lookup table to the left and P-box table to the right

4.3.3 Triple DES

To get a longer key 3 DES algorithms can be chained.
The encrypt it is done by the three keys $k1, k2, k3$ such ciphertext= $E_{k3}(D_{k2}(E_{k1}(\text{plaintext})))$

PC-1

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

PC-2

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

Figure 4: Des permutation choice tables for creating sub keys

For the decryption the encrypt and decrypt is reversed.
The chaining of encrypt decrypt encrypt, is such if a program only implements DES then the triple DES will still work.
Triple is required over double due to double being vulnarable to a meet in the middle attack.
If an attacker has access to the input and output of encryption and decryption and knowns a pair lets say A→B
Then a DES bruteforce is done with A as input and all outputs are saved.
Then B is bruteforce decrypted and all output are saved.
Then a matching pair can be found. This therefore result in two bruteforce has to be done on DES therefore only making the combinations 2^{57} and not 2^{112}

4.3.4 AES

Plain text is split up into chunks of 128 bits.
There is 3 levels of encryption where

1. key: 128 bits, rounds: 10
2. key: 192 bits, rounds: 12
3. key: 256 bits, round 14

The aglorithm works by representing the input as a 4x4 matrix of bytes where the matrix is filled in by columns such b2 is in row 1 column 0, in the following steps

1. Plaintext is XOR'ed with key 0
2. Sub-bytes - A S-box lookup table is used for substituting bytes
3. Shift rows - Each row is cyclic left shifted (in bytes) x times where x is equal to the row number

4. A mix column matrix is multipled to each column of the matrix

5. The round key is XOR'ed to the matrix

Step 2-5 is repeated, until last round where step 4 is not performed.
To decrypt the keys order is reversed.

To find the round keys first a key is generated equal to 128/192/256 bits.
For the 128 bits key These are then divided into 4 words of 32 bits denoted w_i
Therefore making $k_0 = [w_0, w_1, w_2, w_3]$
We can here denote $k_{0-1} = w_1$
To find round i's key the following is done

- $k_{(i-1)-3}$ is left cyclic shifted
- Then it is substituted using the same S-box for the algorithm
- Then the round constant is XOR'ed
- This is then XOR'ed with $k_{(i-1)-0}$ and is equal to k_{i-0}
- To find k_{i-1} $k_{(i-1)-1}$ XOR'ed with $k_{(i)-0}$
- To find k_{i-2} $k_{(i-1)-2}$ XOR'ed with $k_{(i)-1}$
- To find k_{i-3} $k_{(i-1)-3}$ XOR'ed with $k_{(i)-2}$

The round constants are as follows

4.4 Cryptographic hash functions

To ensure the integredity of a message a checksum is used.
The is done by the message being sent into a hash function which generates a unique value for the message.
To ensure authenticity of a message, not only is the messaged hashed but a shared key is added to the message. Called message authentication code (MAC)
Then when the message is received the message is also hashed and checked to be equal to the sent hash value.
This shared key could then be shared using RSA.

To get a digital signature the private key can be used to encrypt a document which then people can verify by decrypting using the public key.

This can then be combined with the MAC, such a method can both have integrity and verified arthour. After generating the hash the senders private key is used to encrypt the hash, which then can be decrypted and verified.
A certification authory (CA) is used to hold public keys and verify to whom a public key belong to.
An authentication protocol can work by first teh sender sends an initiating message, the receiver response with a nonce (never used number), the sender encrypts using private key and receiver decrypts using public key. If they match an uthentication is established.

4.5 Diffie Hellman key exhange

Allows two parties that have no prior knowledge of each other tojointly establish a shared secret key over an insecure channel.
The exhange is based on:

1. A common ground such as a number
2. Client and server adds their own number to the common ground
3. Client and server switch keys
4. Client and server again adds their own number
5. Client and server now has the same number

5 Transport layer

The transport layer is the messenger between the network layer and applicaiton layer.
Therefore the transport layer is only at presense in the end systems of the network.
The transport layer provides protocols for UDP (User datagram protocol) and TCP (Transmission control protocol)
Transport layer packet is called a segment.
The IP is an unreliable service which does not guarantee segment delivery, in order and integreity
This is therefore encounted for in the transport layer.

5.1 Multiplexing and demultiplexing

Demultiplexing is when a segment redirects a data to the correct socket.
Multiplexing is creating segments with header information for the demultiplexing.
The segment header therefor has two fields of 16 bits the source port and detination port.
The well-known port numbers are from 0 - 1023 and are restricted, for things like HTTP and FTP
In case of a server setup with TCP, the server then uses the four parameters in the TCP request (src ip, dest ip, src port, dest port) to setup a demultiplexing on a new port.
Such two client may connect to the same port and unknowingly be switched to a new port without having to change TCP parameters.

5.2 Reliable data transfer

To verify (positive aknowledgments) or request repear (negative acknowledgment) data transmission is known as ARQ (Automatic Repaeat reQuest) protocols
An ARQ protocol requires error detection, receiver feedback and retransmission
This type of protocol will only send new data when the receiver aknowledge the package was received, therefore the name stop-and-wait protocols.
The problem is if the aknowledgment is corrupted, to counteract this the package is sent with a sequence number and in case of unidentifiable aknowledgment the packet can be sent again.
Likewise the acknowledgment also gets the sequence number to counter act dublicate acknowledgments.
The sequence is simply a single bit, to know that the packet is in the right order relative to the last packet.
In case of a lost package or aknowledgment the sender has a time of which after it will resend the package
Choosing the time is hard and must be estimated to be fast enough to not wait too long but still not send too many dublicate packages.
Pipelining is the act of sending multiple packages at once, and increasing the sequence numbering.

5.2.1 Go-Back-N

GBN is a protol which allows the sender to send n packages at once.
To protocol states that the base is the sequence number which is next to be aknowledged
The nextseqnum is the next package to be sequenced and sent.
The protocol can also be called sliding-window protocol because it can be seen as a windows swhich slides over the packages to send.
GBN only use a single timer for all package refering to the last not aknowledged package. If the time runs out all non aknowledged packages are resend.
If the receiver get an out of order package the package is discarded and a negative aknowledgment is sent, where the sender will resend all non aknowledged packages.
Selective repeat combats this by only sending negative aknowledged or timeout packages from which the receiver must keep a buffer of correct and always aknowledge good packages.

5.3 TCP

TCP is duplex thus allowing data in both directions at once.
TCP is a hybrud if selective repeat and Go-Back-N
TCP is point-to-point at therefore only two host can be part of a TCP connection
First a three way handshake is made with no payloads.
Then a send buffer is initialized and the application fills it up.
Then the TCP chooses when the create a segment with buffer data.
The maximum segment size (MSS) is determiend maximym transmission unit (MTU) (1500 bytes in ethernet and PPP) minus header data from IP and TCP (typically 40 bytes)
The segment header includes the same as UDPs (src, destination port number, checksum)

- 32 bit sequence field
- 32 bit sequence aknowledgment field
- Receive window indicating number of bytes receiver is willing to accept
- 4 bit header length field representing number of 32 bit header lengths
- Optional field include timestamp, agreed MMS, windows scaling factor

- 6 bit flag field:
  - ACK bit for aknowledgement
  - RST, SYN, and FIN for connection setup and teardown
  - CWR and ECE for congestion notification
  - PSH pass data to upperlayer immediately
  - URG for urgent segments

5.3.1 Sequence numbering

The TCP sequence numbering is on every byte and the segments numbering is the first bytes number.
Aknowledge numbering works by the response is the next expected aknowledge number.
Say the host recieves a packet segment 0, length 447 bytes, then the host responds with aknowledge number 448
TCP is therefore said to provide commulative acknowledments.
In case of out of order packages the individual implementation decide to either buffer it or discard it.

5.3.2 Timeout time

To find segment timeout the first the round trip time (RTT) is calculated using the formula

$$\text{EstimatedRTT} = (1 - \alpha) \cdot \text{EstimatedRTT} + \alpha \cdot \text{SampleRTT}$$

Where $\alpha = 1/8$ and it can be noted that the new sample is more weighted.
The variation is also important and is calculated using

$$\text{DevRTT} = (1 - \beta) \cdot \text{DevRTT} + \beta \cdot |\text{SampleRTT} - \text{EstimatedRTT}|$$

Where the recommeded value is $\beta = 0.25$
The timeout is then determined as

$$\text{TimeoutInterval} = \text{EstimatedRTT} + 4 \cdot \text{DevRTT}$$

For the first package it is set to 1 second and when a timeout occurs the value is doubled. When measuring only orginal packages are measured such a false time is not measured from a newly sent and old aknowledgment

Fast retransmit can occur if a triple duplicate aknowledge is gotten. Three is choosen since a double could accour in the event of out of order, but three will most likely be a missing segment.
Then the timer is ignored and the second is sent as soon as possible.

5.3.3 Flow control

Flow control is used in case the application is reading data slower than TCP is receiving the therefore overflowing the receiver buffer.
The receiver buffer spare room called receive window will always be equal to

$$\text{rwnd} = \text{RcvBuffer} - \text{LastByteRcvd} - \text{LastByteRead}$$

Where LastByteRead is the byte numbering read by the application.
Therefore the sender will not send more bytes than LastSendByte-LastAckedByte
When the rwnd is 0 the sender have to send bytes until acknowledgment and a non zero rwnd is sent by the receiver

5.3.4 Connection management

A TCP connection is established by

1. A segment without application data is sent with SYN flag set to 1 and a random segment number
2. The server allocates space for variables and buffer and send a segment with SYN = 1, ACK = segment number +1 and a segment number the server itself chooses randomly
3. The client allocates space for variables and buffer and sendst the first data with ACK = server segment number +1 and SYN = 0

When a TCP connection is shutdown the following is done

1. Client send a segment with FIN flag = 1
2. Server acknowledges the FIN flag
3. Server sendt a segment with FIN flag = 1
4. Client achknowledges the FIN flag

When the last step is done 30 seconds is waited before closing fully down.
When TCP acknowledges a FIN segment the variables and buffer space deallocated.
The TCP from client has 6 stages

- SYN_SENT - Send SYN
- ESTABLISHED - Receive SYN & ACK, send ACK

- FIN_WAIT_1 - Send Fin
- FIN_WAIT_2 - Receiver ACK
- TIME_WAIT - Receive FIN, send ACK
- CLOSED - Wait 30 seconds

The TCP for server has 6 stages

- Listen - new open socket
- SYN_RCVD - Receive SYN, send SYN & ACK
- ESTABLISHED - Receive ACK
- CLOSE_WAIT - Receive FIN, send ACK
- LAST_ACK - Send FIN
- CLOSED - Receive ACK

In the case of the server receiving a TCP request on a non TCP port a segment with RST flag is sent back.
For a server to prevent SYN flooding, a hash of the client is made from IP, sorce and port number called cookie and is sent as segment numbering and forgotten.
When receiving the SYNACK the cookie is recalculated and only if the ack is cookie +1 will the server allocate space and recognize it as a legitimate request.

5.3.5 Congestion Control

Classic TCP uses end-to-end congestion control rather than network assisted (Where router indicates to host if congestion is a problem).
To exercise congestion control the sending rate is limited to

$$\text{LastByteSent} - \text{LastByteAcked} \leq min(\text{cwnd}, \text{rwnd})$$

Where cwnd is congestion window.
cwnd is determined by

- If a timeout occours or a retransmission the cwnd is lowered
- A successfull delivery will increase the cwnd

The algorithm therefore works by first starting in slow start with cwnd = 1 and is knwon as additive-increase,multiplicative-decrease (AIMD)

- Slow start - First the rate is 1 MSS, then for every acknowledged segment 1 MSS is added to cwnd essentially doubling it until a problem occour. When that happends a variable ssthresh is initiated to cwnd/2 and cwnd is set to 1. Then again slow start is begun on it until it reaches sshtresh from which congestion avoidance is begun. In case of 3 dublicates fast receovery is ran with ssthresh = cwnd/2 and cwnd = sshtresh + 3MSS

- Congestion avoidance - Now cwnd is incremented only by MSS/cwnd until a problem occour. If a dublicate occour three times sshtresh is updated to cwnd/2, cwnd is set to ssthresh + 3 MSS and fast recovery is ran. If a timeout occurs sshtresh is set to cwnd/2 and cwnd is set to 1.

- Fast recovery - cwnd is incremeneted by a single MSS. In a timeout sshtresh = cwnd/2, cwnd=1 and slow start is begun.

The Cubic version is an alternative where the congestion avoidance saves the last max limit it reached and then cubicly reaches it such the farther the distance to the maximum the more cwnd is increased.

The network assisted version works by, the router sets the ECN (Explicit congestion notification echo) flag on a segment to the receiver, which also sets it in the acknowledgment. Which result in hafling the congestion window
Delay based congestion control works by using the RTT, if the sender then measures a significantly less tha nthe uncongested througput rate, then it slows down the sending rate. Essentially it works by avoiding queues.

It can be seen TCP will fairly share bandwith and conestion controle will be equal in multiple TCP connections over time.
This is unlike UDP which will not behave fair and force TCP to get less bandwith without itself taking less.

5.4 QUIC

QUIC is an extension to the UDP protocol, which provides features from TCP.
A handshake is initiated, but it only needs a single send and receive to setup therefore saving a whole RTT over TCP.
Even QUIC-0 can be used in case of recent connection where no handshake

is needed as saved params are used.
QUIC provides encryption of packages and uses http3.
QUIC uses reliable, and congestion controlled data transfer features from TCP

5.5 TLS

Transport Layer Security (TLS) is an extension to TCP which gives it security in the form of: encryption, integreity and authentication.
Websites using TLS is recognized by the use of https.
TLS acts as a sublayer in the TLS socket and applies it security.
When running TLS the data is split up into records which is encrypted and hashed using HMAC.
The TLS record consist of: Type, Version, Length, Data, and HMAC, where data and HMAC is encrypted with E_B
The type field is used for a copy of the TCP type, such as FIN flags but by using it in the TLS, it is included in the hashing such no tampering of the TCP flags will destroy the connection.
Likewise is the sequence numbe and length of message also part of the hash .
A TLS connection is as follows:

1. A list of supported cryptographic algos and a nonce is sent from client
2. Server chooses a cryptographic alg. for symmetric key, public key and hmac. Then responds with choices, certificate and server nonce
3. Client confirms certifcate and gets server public key, generates Pre-Master Secret (PMS), and sends encrypted PMS with server public key
4. Using the choosen algos the server and client generates Master Secret (MS) from PMS and nonce. The MS is sliced up into: 2 encryption keys, 2 HMAC keys, initialization vectors in case of symmetric cipher
5. Client send HMAC of all handshake messages
6. Server send HMAC of all handskake messages

The HMAC handshake messages are used to ensure integrety of the non encrypted handshakes before. Likewise is the nonce used to prevent replay attacks.
The keys generated from the MS is denoted

- E_B session encrypted key for data from client to server
- M_B session HMAC key for data from client to server
- E_A session encryption key for data from server to client
- M_A sessuin HMAC key for data from server to client

6 OWASP - Top ten seciruty risk

Open Web Application Security Project conducted a list of biggest security risk based on data on attacks. The following the that top then in order.

6.1 Broken Access Control

The biggest problem is allowing attacker acces to data or controle which was not intended.
This may be can be through websites, API or any other access to data or controle.
Problem often occur due to amount of access points and lack of testing.
Attackers often use modification of accessible data, such as cookies or url which may not be sanitized.

6.2 Cyptographic Failues

Many systems and companies may use old cryptography or non at all on sensitive data.
This can also be in form of transportion of data both internal and external.
Lack of this will result in case of leaks actually exposed data rather than encrypted data.

6.3 Injection

Injections occur when user data is sent to an interpreter without sanitizing the input.
The can be queries for databases or bad designs for inputs which can result in bad input.
This can be in buffer overflows which may allow injection of code and gaining access to higher priveledge.

6.4 Insecure Design

Insecure design is not the implementation but rather the security choices. This can be password recoveries which is too weak or intern protocols which may be too weak when handlign sensitive data

6.5 Security misconfiguration

This may be outdated libraries or default setting which is unchanged and therefore leaves a vulnarability.
This may also be in case of settings for development end up in production leaving a open backdoor or like wise.

6.6 Vulnarable and outdated components

Likewise seciruty misconfiguration but this focus more on the oudated packages or libraries which may be in use.
The libraries may contain vulnerabilities which can gain acces to the code or data.
This is prevented by subscribing to bulletins of included libraries or packages and keeping them up to date.

6.7 Identification and authentication failues

Bad implementation of authentication which gives attackers opportunities to gain access via session tokens, keys or passwords.
This can be in the form of, allowing brute force attacks, allowing weak passwords or having a weak password recovery
A good solution is often implementing two step authentication.

6.8 Software and data integrity failues

This happends when an attacker identifies as software or plugin and forces an update upon working code.
This may also be if an attacker simply notices a data stream which does not check for integrety and themselves may send attacking data.

6.9 Security logging and monitoring failures

Most attacks are not detected before 200 days after.
This is due to lack of securty logging and monitoring attacking attemps.

A better monitor or log may capture attemps and warn of risky areas before an attack happen

6.10 Server side request forgery

When a web application is fetching remote ressource without validation.
This may lead to an attacker beign able to load attack ressources onto the application.
This is prevented having a firewall which block all trafic beside the known remote places.

7 Penetration testing

The two most comon forms of penetration tests are: Application penetrations testing and infrastructure penetration testing.
Other types are

- Mobile application
- Device penetration testing (laptops, consumer devices)
- Wireless penetration testing
- Telephony or VoIP

There are three different types of intruders

- Hackers - Experimentally minded programmers targets security loopholes in it system
- Crackers - Exploits weak points of it systems to gain illegal advantages
- Insiders - Type of crackers with inside knowledge
- Script kiddies - Uses attack tools

There are mainly 3 different types of attacks

- Network based attacks - attacks using network protocol functionalities this includes port scanning, IP spoofing, sniffing, session hijacking, DoS attacks, buffer overflow and format string attacks
- Social engineering - Maniupulating people to revela security related information

- Circumventing of physical security measures - By using physical attacks and gaining access to physcial data or authority

A company should have secuirty policies and secuirty concepts, without that a pen test would not even be helpfull.
The procedure of pen testing are in the following steps

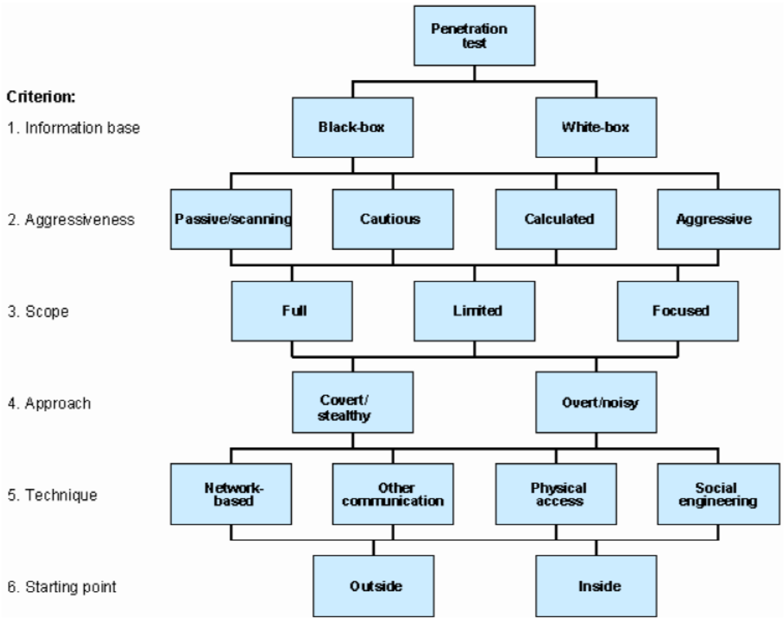
1. Research information about the target system - information like official IP address
2. Scan target systems for service on offer - Using a port scan internet services can be found
3. Identify systems and applications - By using fingerprints in the prt scan
4. Research vulnerabilities - Finding exploits in systems with open ports from fingerprints
5. Exploirt vulnarabilities - Using the known vulnerabilities and gaining access for further attacks

7.1 Classifying pen tests

Starting point - Attack of penetration most common: firewalls, web servers, RAS access points and wireless networks.
External servers are known as normal workstations.
Web serbers are often vulnarable due to there minufold of functions
Pen testing is usefull since security audits and IT audits most often focus in IT infrastucture in terms of compliance, efficiency nad efectivness.
The goals of pen testing should be clear and can be divided into four categories

- Improving security of techinal systems - This can be finding improvements in firewalls, routers, web servers, etc..
- Identifying vulnarabilities - This is the actual objective of the test
- Having IT security confirmed by an external third party - The IT system and infrastructures current security
- Improving security of orginazational and personnel infrastructure - This is often in form of social engineering

The limtis of a pen test is the fast discovery of new vulnarabillities making pen test only validation of security currently but not in the future. Likewise does the pen test reduce the probabillity of a successfull attack. The type of attack is also classified to test a given scenario



- Information base - The amount of known information, where black box represent the unknown insider information unlike white box
- Aggressiveness
  - Passively - Vulnerabilities are detected but not exploited
  - Cautious - Vulnarabilites are only tested when it will not result in system suffering
  - Calculated - Exploits vulnarabilities which may result in system disruptions
  - Aggressive - Exploit every vulnarability even if deactivating systems or overloading
- Scope - The amount of testing which directly relates to time spent, focused sub network after modifcation, limited number of systems, full everything avaliable

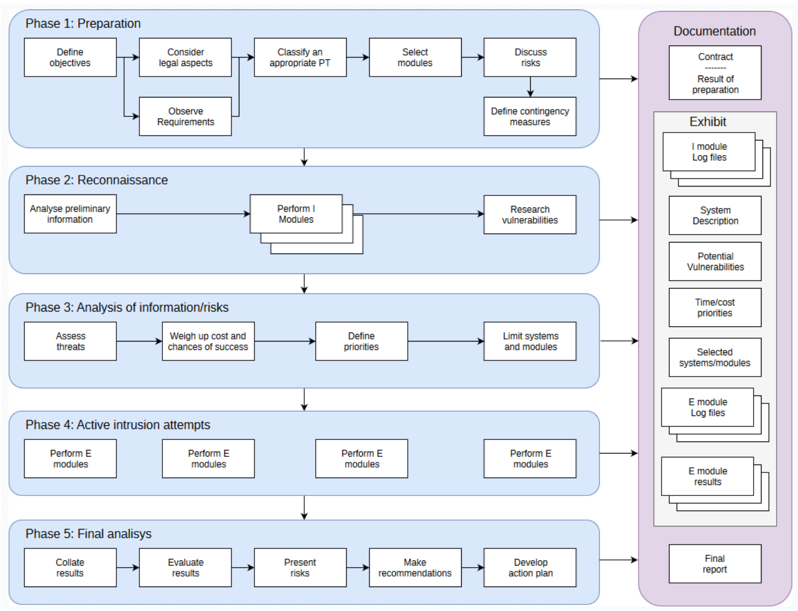
- Approach - How visible the attack is, covert stealthy, overt often with staff such fixes can fast be done
- Technique - Networks based using network protocols also known as IP-based pen test, Other communcation networks, mobile, fax, wireless networks etc., physical in case of good enough firewall, social engineering people are frequently the weakest link
- Starting point - Is the test performed from an outside perspective, or inside networks where firewalls do not have to be overcome

Most often a combination of pen tests are adviceable.

7.2 Legallity and ethical issues

There are different legal issues which rises from pen testing.
For a company pen testing may be needed, to ensure data security to compart with laws.
When a pen test is performed approval by client in all areas is required.
Likewise often 3rd oarty also need to approve the test
The test should have a liability to cover claims from 3rd parties.
Ethical issues also araise with social enginering in case of wanted anonymity.

7.3 Phases of pen testing



Preparation - agree on scrope and cost based on classification, contracts in plase and discuss risks
Reconnaissance - passive test, where information is obtained and overview of system.
Tool for data collection

- whois
- https://website.informer.com/
- namp
- https://spokeo.com/
- osint examples
  - https://www.shodan.io/
  - https://www.spiderfoot.net/
  - https://github.com/laramies/theHarvester

Analyzing infromation and risks - use data to find risk and determine which to test based on goals and time
Active intrusion attempt - This phase test the vulnarabillities found, it is here important to consider actual risks and whihc risk are not affecting the product/organization
Final analysis - Evaluate the vulnarabilities and where they where found and recommandation to eliminate them

7.4 Tools

- Kali linux - OS with preinstalled tools
- Nmap - port scanning
- NCAT - Read/ write network connection using TCP or UDP
- Metasploit - Frameowrk woth most common exploits
- SQLmap - Detecting and xploiting sQL injections flaws
- NIKTO - Scan for harmful files, misconfigurations, outdated software installations on web server
- BURPSUITE - Pen testing web application tool

- John the riper - Tool for cracking password
- NESSUS - vulnerability scanner (not free)
- Wireshark - Packet sniffer and network analyzer tool
- AIRCRACK-NG - wireless network tool
- Tool list - https://sectools.org/

8 The network layer: Data plane

The role of the network layer is the route packets from host to receiver.
Routing is done by the all routers by each forwarding the packet from input to correct output link.
This is done using routing algorithms.
The result of routing algorithms is a forwarding table which determines which output link the router should forward to.
There are two approaches to creating the forwarding table either the routers share their tables between eachother using a routing protocol.
Or the SDN approach where an external remote controller creates tables for routers and send them it. This can be done by third parties or ISP.
SDN stands for software-defined networking.
The networks service model is defined to be best-effort service therefore many features are not implemented other than trying to deliver.

8.1 What is insde a router

Input ports is used for terminatign incomming physical link.
The input performs the lookup function to the forward table via the switching fabric.
In case of a control packet the packet is forwarded to the routing processor.
The switching fabroc connects the input and output ports.
The putput ports store the sending packets and perform the necessary link-layer and physical layer functions.
THe routing processor maintains the router, by updating the forwarding tables or in case of SDN connects to remote and update forward tables.
The forwarding tables matches a prefix of the packets destination address.
If a match does not exist it exists via the default port.
If multiple matches the longest is choosen.
If the ports is in use the fabric blocks the package and queues it.

All this has to happend in hardware to keep up with transfer speeds.
The input port also has to, physical and link layer processing, check the checksum and time to live, update counters and time to live.

8.1.1 Switching

The act of switching can be done through different methods.
The simplest version switching via memory, copies inputs into processor memory and let the processor due to moving to output.
This is limited to memory bandwidth and therefore the transfer will be limited to half the memory bandwaidth.
Switching via a bus transfers packets via a bus, then every packet get appended a label and the matching output port keeps the packet the rest throws it away.
The label is then removed before sending.
This is then limited to the bus speed since only one package can be worked with at a time.
Switching via an interconnection network, uses a crossbar switch, where a bus from every input goes over output ports bus, such an intersect is present for each.
Then the intersect between input and output is turned on to forward the package.
This results in the package only being blocked in case of the wanted output is taken.
More sophisticated approached uses a stacked interconnect such the output can contain multiple packages

8.1.2 Queue and buffers

Queue can occuor both on output and input, input will happen when either too many packages comes into one input at once or head of the line (HOL) blocking, where another input waits for output and therefore block the input buffers.
Output will occur if the transmission rate is slower than the forwarding and input.
When queue occuor either the latest (drop tail) is dropped or a random picked using an active queue management algorithm
The drop tail has the advantage of faster congestion signalling to sender.
The size of the buffer has to be large enough to prevent package lost too fast in case of a burst, but smaller than queuing delays are too high.

The rule of thumb is a buffer size equal to

$$B = RTT \cdot C/\sqrt{N}$$

Where C iis the link capacity and N is the number of independent TCP flows.
TCP is part of the equation since in case of a buffer being filled up, since TCP will recieve an ACK for every sent from the buffer the buffer will never empty known as bufferbloat.
For emptying the queues different approaches can be taken

- First in first out FIFO - The packages is handled in the same order as arrival
- Priority queuing - The packages are classified upon arrival which then is handled in priority order
- Round Robing and Weighted fair queung - Like priority queuing, but to ensure every package is sent at some point the queues is served in a circular manner. The class weight then determines the time used pr. queue such higher priority equals more time spent in the queue before moving on.

8.2 The interprotocol (IP)

8.2.1 IPv4

The datagram of an IPv4 is

- Version number - 4 bits to determine version number like IPv6
- Header length - IPv4 datagram contains variable headers therefore 4 bits is used for indicating size
- Type of service - Used to distinguish if the package is a real-time datagram or an non-real-time traffic like FTP, in here two bits are also used for congestion control
- Identifier, flags, fragmentation offset - Used for IP fragmentation
- Time-to-live - Counter for number of routers which the package can be sent through before being dropped
- Protocol - A number representing which protocol is used after the IP the handle the data like TCP or UDP

- Header checksum - Checksum of the integer value of every second header values sum. Recalculated for each router step since headers changes like time to live
- Source and destintion IP
- Options - Allow to extend the IP headers further
- Data

The header is 20 bytes without options and with IP the size is 40 bytes
An interface is the boundary between link and host and has each their own IP.
With a 32 bit long IPv4 the total number of IPs is 2^{32} or aprox 4 billion
The IP is formatted into 4 chunks bytes which is written in decimal with dots between the decimals.
Subnets can be defined such a prefix is the identifier and then the whole subnet uses the prefix and the rest is for intern routing.
This is in the format a.b.c.d/x where x is the number of bits used for the prefix and the rest is the subnet.
This is known as Classless interdomain routing (CIDR). This allows for subnets to have way smaller forwarding tables.
Before CIDR the subnets size was bounded to 8, 16 or 24 bits making the jump from 8 to 16 very large.
255.255.255.255 is used for sending a package to every device in the subnet.
To get a subnet the isp is contacted which will give part of their subnet to the company.
The isp subnet is managed by regionals which is managed by Internet corporation for assigned names and numbers (ICANN).
ICANN is a NPO which mangages DNS root servers, assigning fmain names and resovlg=ving dfomain name disputes.

8.2.2 Obtaining an address

Dynamic Host Configuration Protocol DHCP is a protocol used to configure devices by admins.
DHCP allows to assign a static IP to devices or temporary IP.
DHCP requires a server or device which reulays the DHCP protocol.
The protocol consist of

- DHCP server discovery, which is an udp packet send to port 67 to 255.255.255

- DHCP server offer(s) - Once a discovery is gotten a offer is sent back to the subnetwork, and contains transaction ID, proposed IP, network mask and IP address lease time (The time the ip is valid)
- DHCP request - Client chooses from offers and respond to the wanted by echoing the content
- DHCP ACK - an acknowledment message is sent confirming the device

The problem with DHCP is the requirement for the protocol for every subnet which is a problem in case of more routers in large areas.

8.2.3 Network address transaltion (NAT)

NAT is a method for using a routers as a single device.
This allows for a single IP which is shared among the local network.
This works by when a device sent a package to the router, the request is contains the wanted IP and a socket to the device from route.
The router then rewrites the source and source ip to the wan ip.
This allows up to 60,000 devices on one network with 1 ip address.
This is argued against since is reuses port numbers for addressing purposes.
This also requires for server like application a NAT traversal tool for finding open ports in the local network.
ANother problem is it ruins the archtiecture of the internet, by not assigning devices an IP but rather a number of devices.

8.2.4 IPv6

The problem with IPv4 is the limited amount of IPs.
The last IPv4 address space was given out in 2011 to a reginal registry.
There is open IPv4 address in regionals but the global pool is empty.
IPv6 uses 128 bits to create IPs making sure it will never be problem again.
The format goes as

- Expanded addressing capabilities - This allows to send a package to an IP group
- A streamlined 40 byte header
- Version - nubmerical value 4 for IPv4 and 6 for IPv6
- Traffic class - 8 bit field for priority certain diagrams

- Flow label - Allowing to label packages as flow and given them optional priority
- Next header - Like to protocol in IPv4
- Hop limit - Like time-to-live in IPv4
- Source and destination addresses
- Data

8.2.5 Openflow

Openflow is the mechanism which performs the match and action for packages.
The mathc is done over the whole datagram, including link,- network,- and transport layer.
The there means

Ingress port				
src MAC	dst MAC	eth Type	VLAN ID	VLAND Pri
IP src	IP dst			
IP proto	IP TOS	TCP/UDP src port	TCU/UDP dst port	

From this a flow table is generated which states rules, and matching will be used in priority order.
Each row then has a list of action which in done in the given order.
The include forwarding, dropping and modifying fields.
This can be used to create firewalls by blocking IP or methods by dropping the matching columns

8.2.6 Fragmentation

The fragmentation of a packet is the act of splitting up the package such they can fit into the maximum transfer unit (MTU)
The header will have offset set such the packets order can be seen and is divided by 8 bytes to save space.
The offset will come from the previous packages data length (not inclduding header 20 bytes if no options)
The fragflag is not raised on the final package.
The fragflag is raised and the ID will match across the framentations.

9 Control plane

9.1 Routing algorithms

Routing algorithms is used to find the least cost routes between sender and receiver.
The cost may reflect, physical length, speed or mentary cost of links.
Two nodes cost is said to be $c(x, y)$ and in case of not being connected the cost is infinite.
The graphs is considered undirected.
Centralize routing algorithmms use global knowledge of the network.
To communicate a link-state broadcase algorithm is used. From this the Dijkstras aglorithm can be used to find least cost to each node.

The algorithm will run at a centralized unit and the algorithms is refered to as link-state (LS) algorithm
Decentralized rounting uses iterative, distributed manner by the routers.
The routers only have knowledge of neightbour routes and the cost to them.
The routers then exhange information about their neightbours to calculate the cost further.
This is called distance-vector (DV) algorithm.
This is done async by to nodes does need to work at same time, and iterative by how each nodes send and updates on changes.
The network cost is then calculated using bellman-fords algorithm.

Algorithms can also be classifed by being static which change slowly over time.
And dynamic which changes for every traffic loads or topology change.
Load sensitive algorithms takes the package load into account to congestion.

9.1.1 Problems

Oscillations is where a network of nodes oscillates between two optimal path and therefore never end termination.
To prevent this the routers send their link advertisment at random intervals to also prevent self-synchronizing.
Count to infinity problem occur if the nodes A,B and C is connected with cost 1.
If B and C is disconnected and does not send an update before A sends an update to B and advertise its length to C at 2.
This will then result in B updating to 2, A updating to 3 since it used B to

get to C. This will then keep happening until they count to infinity.
A solution which works with small node envirements is the poisoned reverse, which advertises when routes is broken or increased.

9.1.2 BGP

BGP is a protocol for connection between AS
In the AS there are two types of routers, gateway routers on the edge connecting between AS and internal.
The BGP protocol uses TCP since it needs to transfer large amount of data, here the forwarding tables in other AS.
The BGP protocol comes in two types iBGP and eBGP
iBGP is for internal use, such when a new forwarding table has come into the AS it is shared between internal routers using iBGP
eBGP is external BGP which communicates between different ASs.
A new external forwarding table may include everything to AS1 goes through routerA.
The internal then simply uses the optimal path to get the a router which is a gateway.
BGP routing is done through different methods

- Hot potato - The AS focuses only on reducing cost internally and ignore outside policies.
- Route-selection - This is the used algorithm used in case of multiple routes and the route is selected in the given order
  - Check local preferences on attributes which is determined by policies
  - The shortes AS path is chosen
  - Hot potato is used
  - BGP identificers is used to select

BGP is also used to find the nearest DNS server, using IP-anycast.

9.1.3 SDN control plane

Software defined network (SDN) network with seperated controller.
A SDN network can be identified by the following characteristics

- Flow based forwarding - Packet forwarding can be don based on any number of header fields

- Seperation of data plane and control plane - Instead of the routers doing both data and controle plane, is the control plane done by a seperate entity
- Network control functions - Communicate between a SDN controller and network control applications
- A programmable network - Using API in the SDN controller the network switches can be programmed

A SDN controller consist of

- A communication layer - Communication between SDN controller and controllers using open flow or other protocols
- A network-wirde state-management layer - Control decisions, such as flow tables, load balancing or firewalling capability
- The interface to the network-control application layer - Interfaces for read/write network state and flow table

In practise the SDN controller consist of distrubted set of servers for fault tolerance, hight availability and performance.

9.1.4 Openflow

Openflow uses the TCP protocol over port 6653
Openflow allows for messages from SDN controller to controlled switch

- Configuration - Set a switch configuration params
- Modify-state - Add/delete entries in the flow table
- Read-state - Collect statisticts from flow table and ports
- Send-packet - Send a packet using the controller

From the controller switch the SDN controller can get the packages

- Flow-removed - Flow entry on table has been removed
- Port-status - Change in ports
- Packet-in - Packets wich does not match table or should be sent to controller