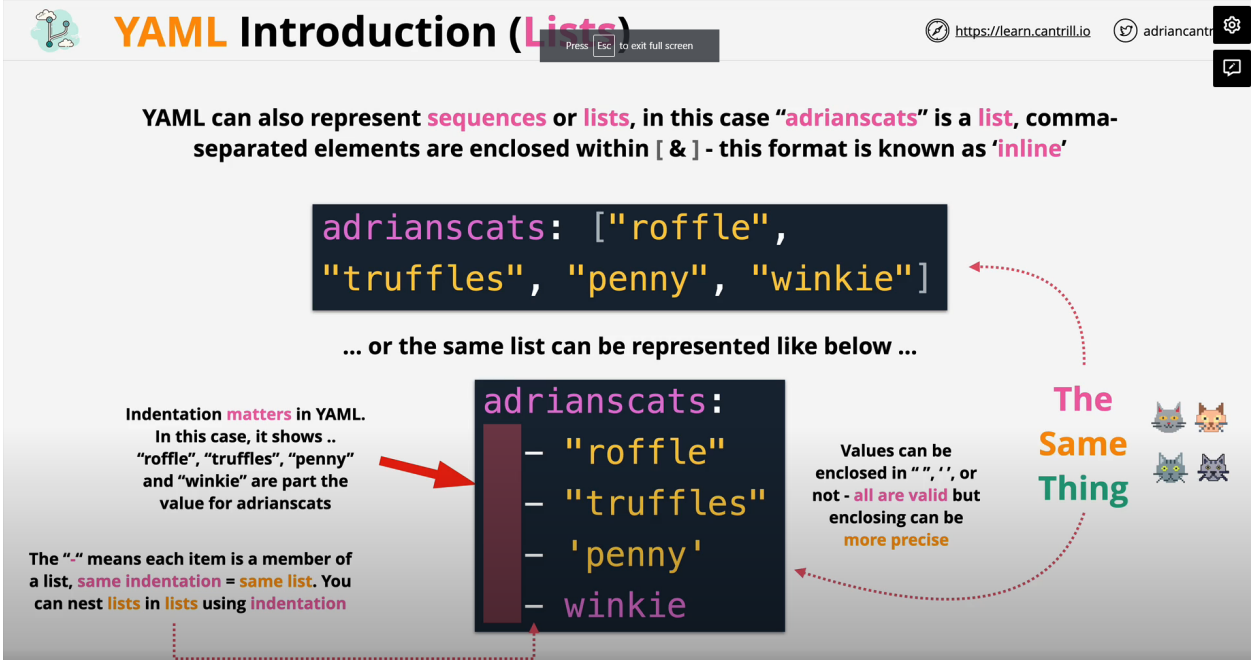
3. Tech Fundamentals

Created	@May 31, 2022 11:45 AM
Class	
Type	
Materials	
Reviewed	☐
Property	
Date	

Tech Fundamentals

YAML- mostly used for cloud formation



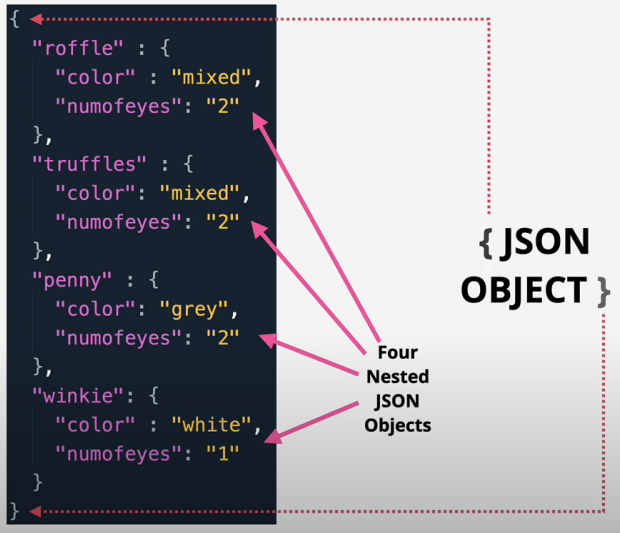
JSON: used for cloud formation plus other policy docs

JSON Introduction

https://learn.cantrill.io adri

The top level is a collection of unordered key:value pairs, where the value is a JSON object

Each Nested Object is a collection of key: value pairs where the value can be a scalar, a list or a JSON object



JSON Introduction (CloudFormation)

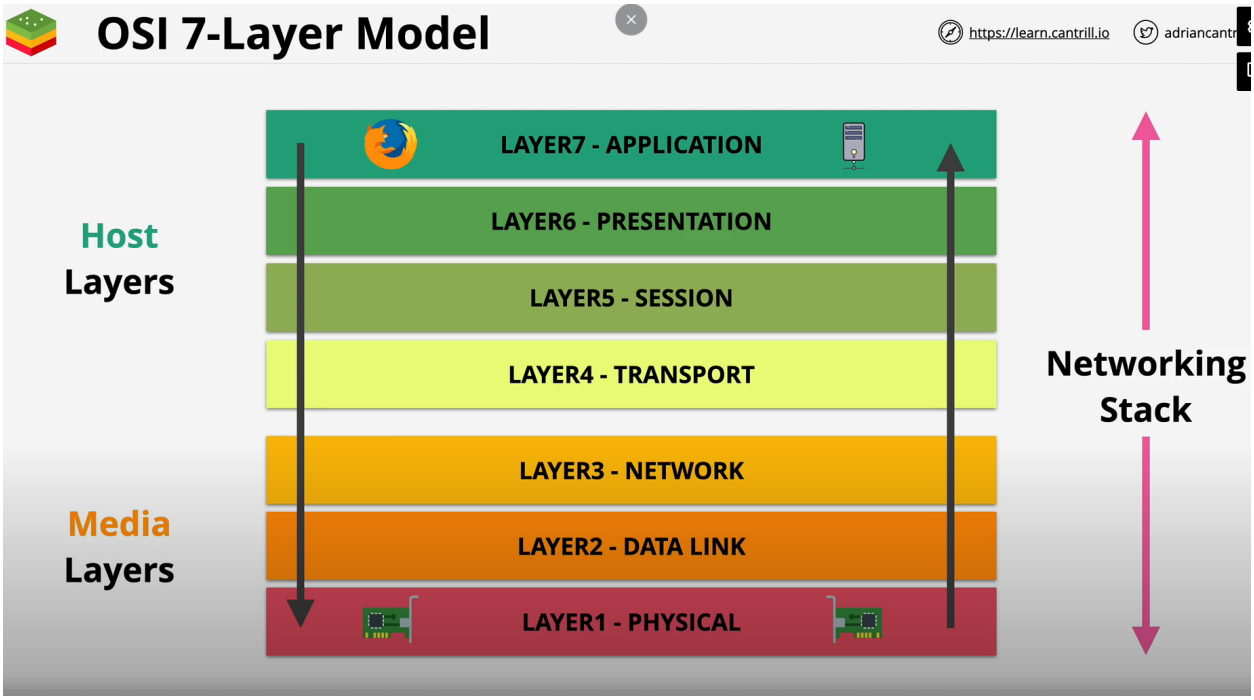
https://learn.cantrill.io adriancant

This JSON template has a 'Resources' key, its value is a JSON OBJECT
Within it, a key 's3bucket', its value is a JSON OBJECT
..containing 'Type' and 'Properties' keys. Type has a string value
....Properties is a JSON OBJECT containing 'BucketName'
.....BucketName is a key with the value of "ac1337catpics"

```
{
  "Resources": {
    "s3bucket": {
      "Type": "AWS::S3::Bucket",
      "Properties": {
        "BucketName": "ac1337catpics"
      }
    }
  }
}
```

OSI layers

All People Seems To Need Data Processing

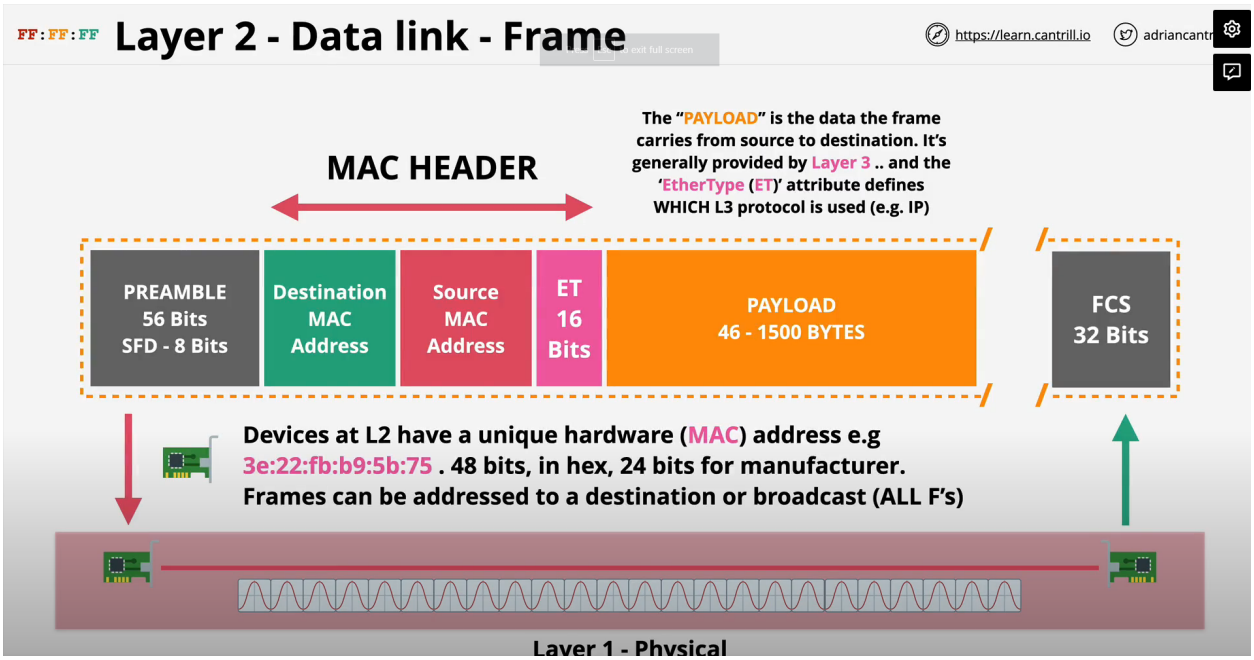


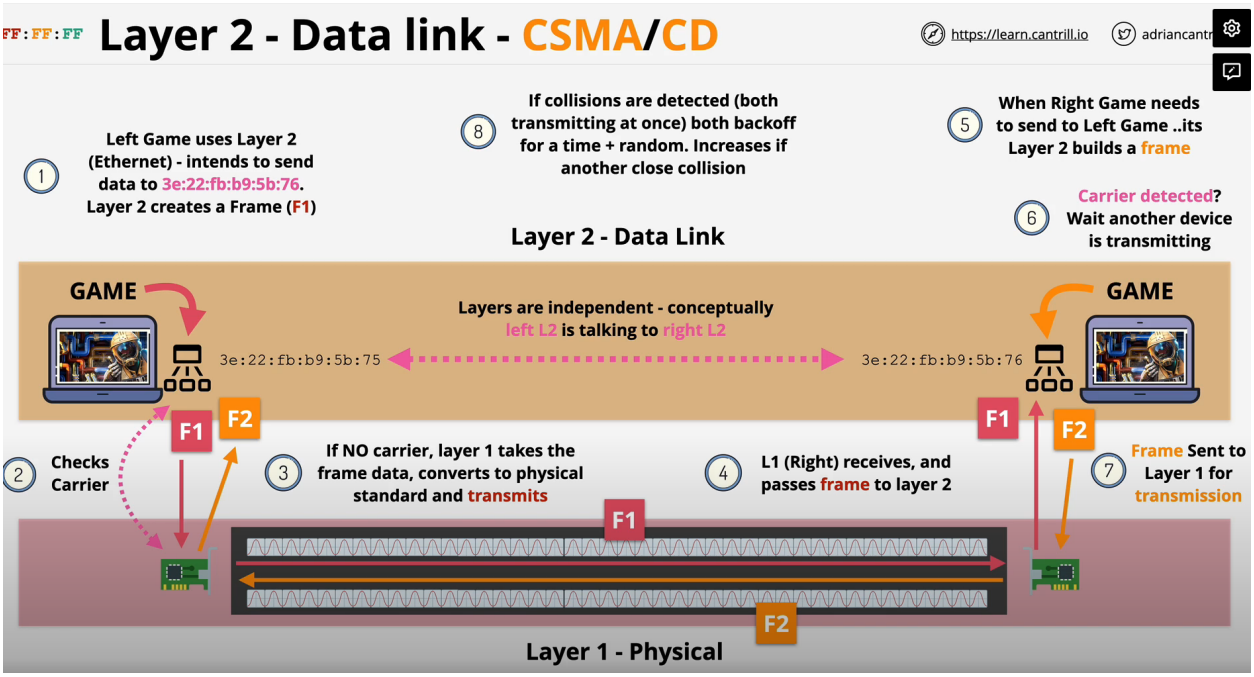
L1: Physical: 01010 no intelligence all get the same info
It cant detect the particular link so all data is broadcasted

L2: Data Link Layer
L2 has Mac add- 48 Bit

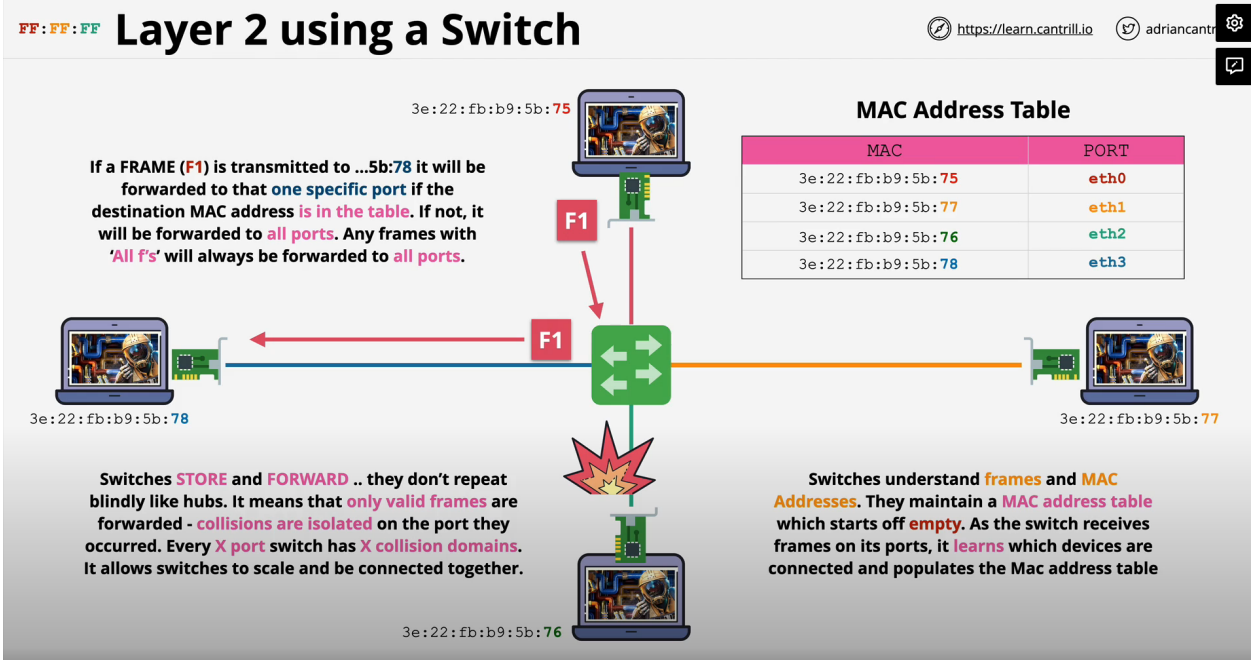
L2 provides frame and L1 handles the transmission without knowing it

ET: Ethertype is used to know which L3 protocol is original putting data into that frame

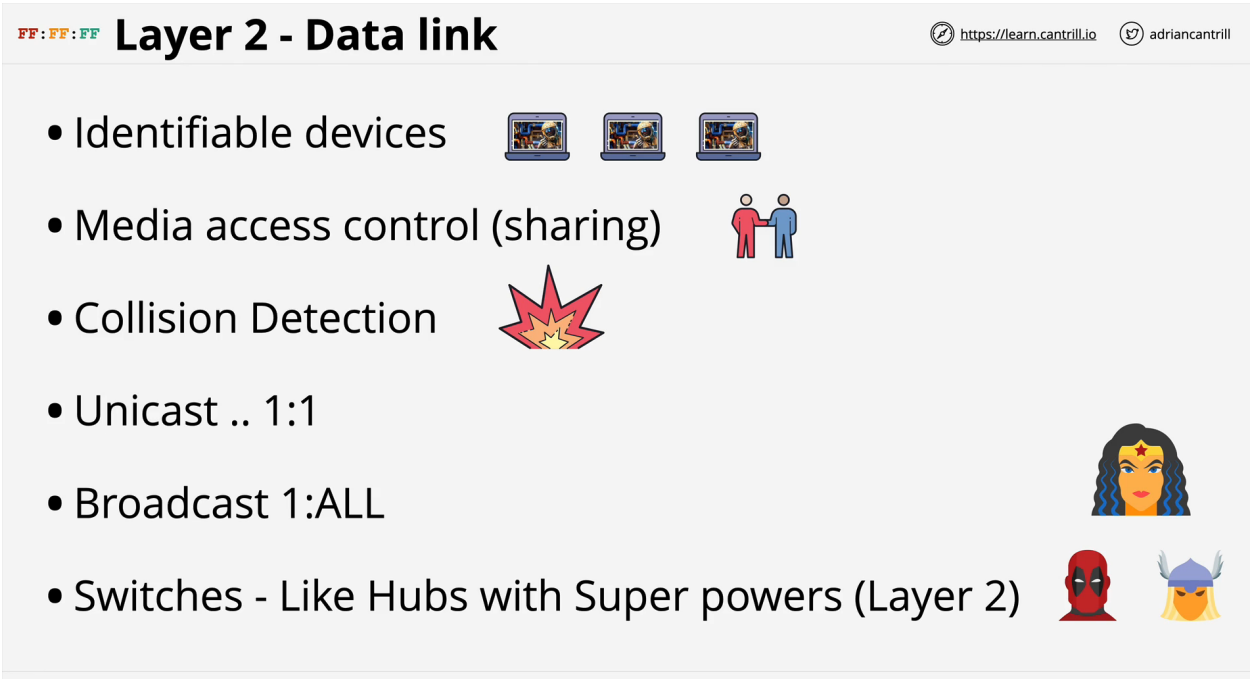




L2 with switch


FF:FF:FF Layer 2 using a Switch
https://learn.cantrill.io
adriancantr
3e:22:fb:b9:5b:75
If a FRAME (F1) is transmitted to ...5b:78 it will be forwarded to that one specific port if the destination MAC address is in the table. If not, it will be forwarded to all ports. Any frames with 'All f's' will always be forwarded to all ports.
F1
MAC Address Table
MAC
PORT
3e:22:fb:b9:5b:75
eth0
3e:22:fb:b9:5b:77
eth1
3e:22:fb:b9:5b:76
eth2
3e:22:fb:b9:5b:78
eth3
F1
3e:22:fb:b9:5b:78
3e:22:fb:b9:5b:77
Switches STORE and FORWARD .. they don't repeat blindly like hubs. It means that only valid frames are forwarded - collisions are isolated on the port they occurred. Every X port switch has X collision domains. It allows switches to scale and be connected together.
Switches understand frames and MAC Addresses. They maintain a MAC address table which starts off empty. As the switch receives frames on its ports, it learns which devices are connected and populates the Mac address table
3e:22:fb:b9:5b:76

Use of L2:


FF:FF:FF Layer 2 - Data link
https://learn.cantrill.io
adriancantrill
• Identifiable devices
• Media access control (sharing)
• Collision Detection
• Unicast .. 1:1
• Broadcast 1:ALL
• Switches - Like Hubs with Super powers (Layer 2)

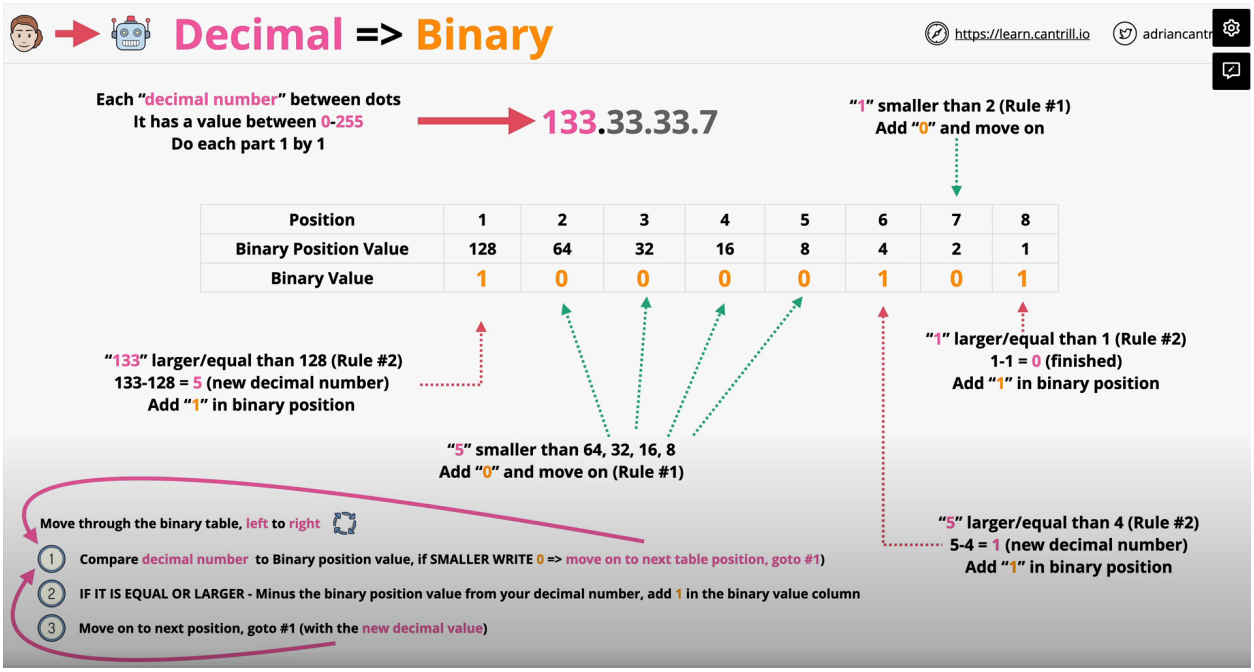
Decimal to Binary or visa versa

IP is 32 bit

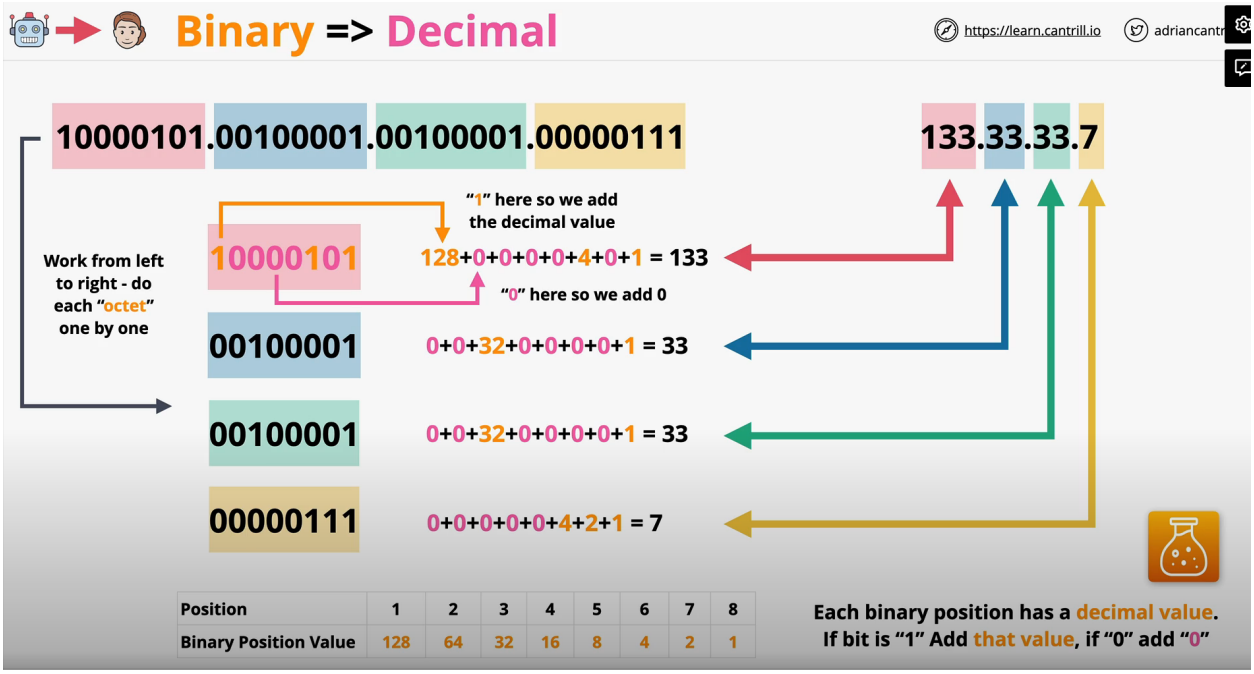
to do this you have to understand the table of 8 bit which goes

1 2 4 8 16 32 64 128

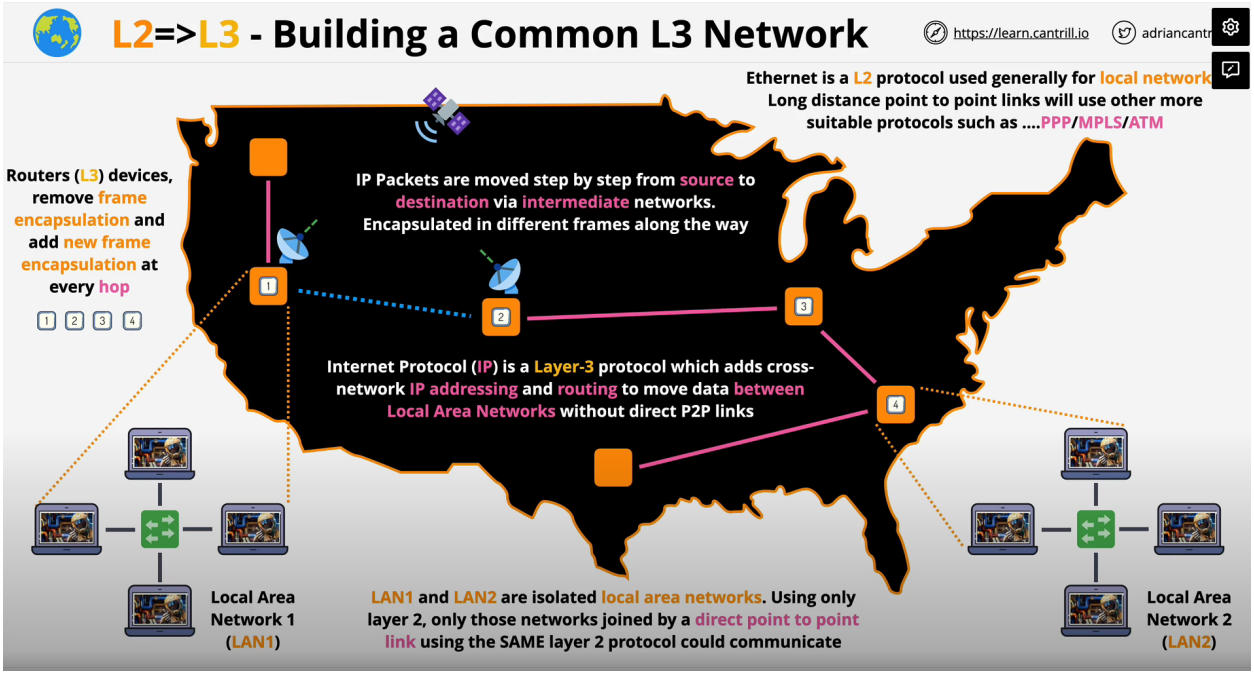
1	2	3	4	5	6	7	8
128	64	32	16	8	4	2	1

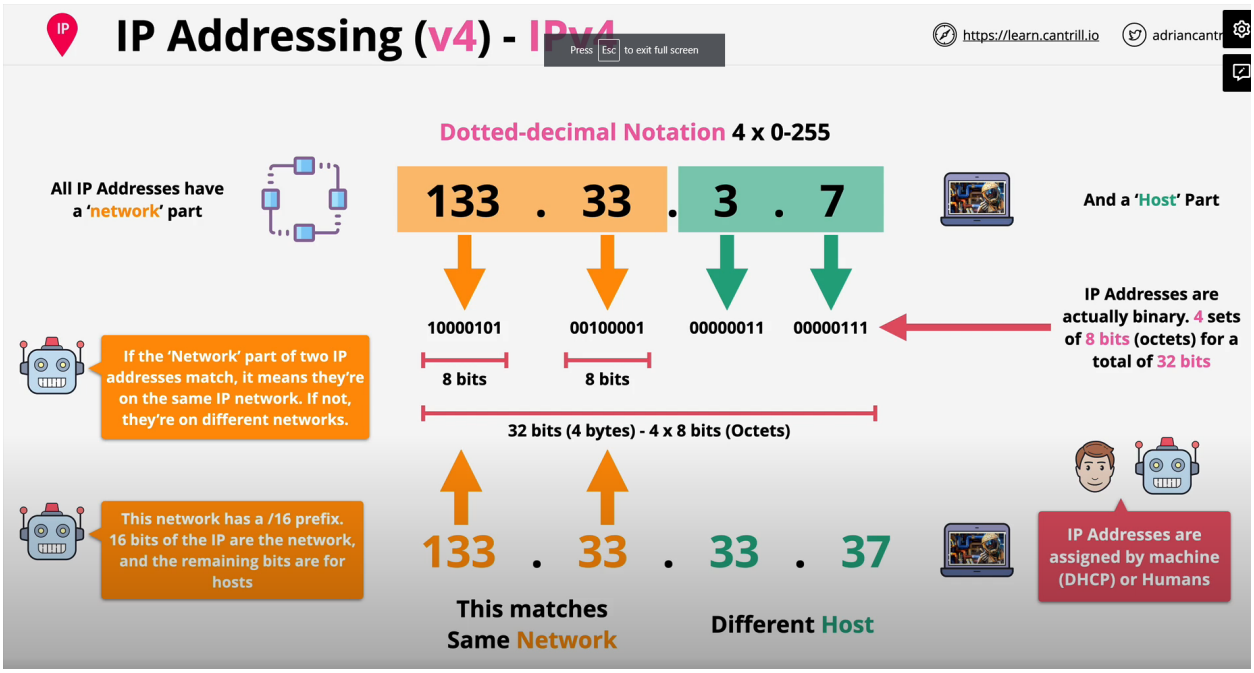


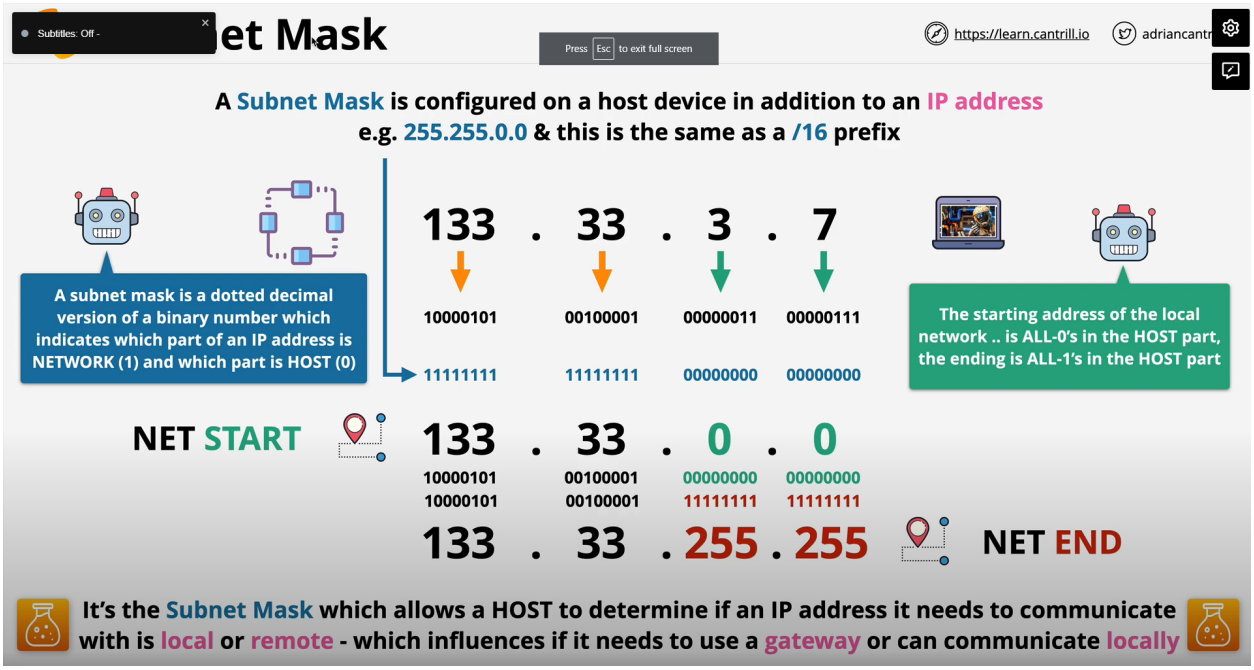
binary to dec same table



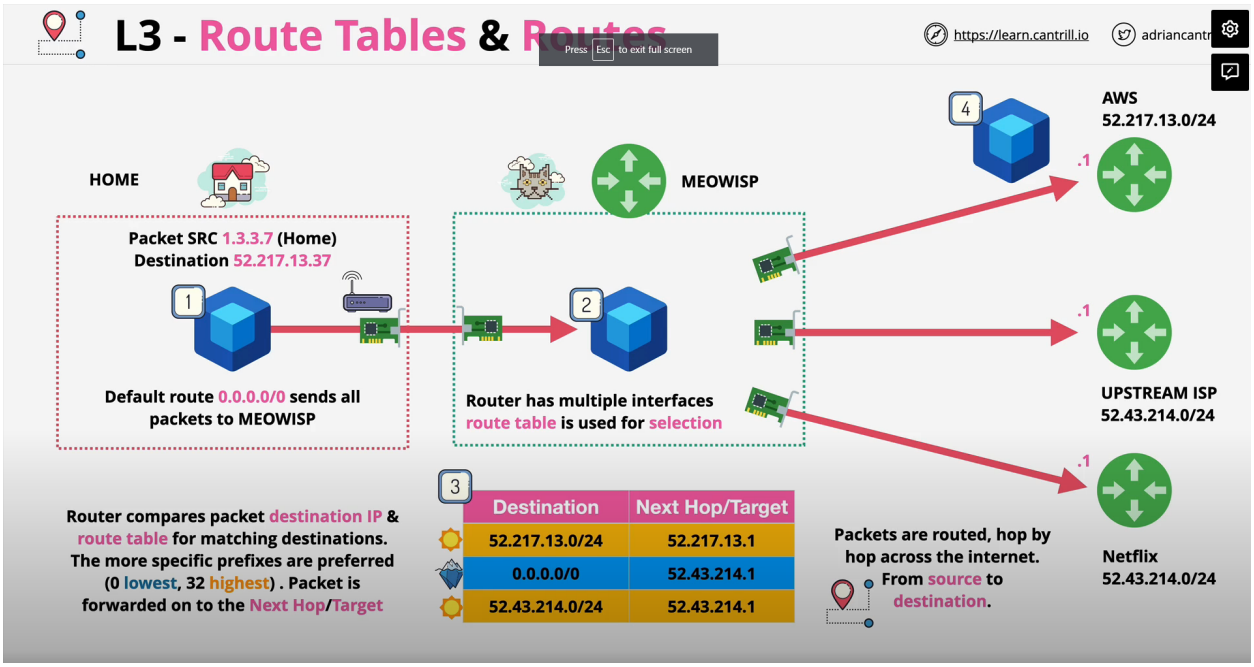
Layer 3 : Network layer



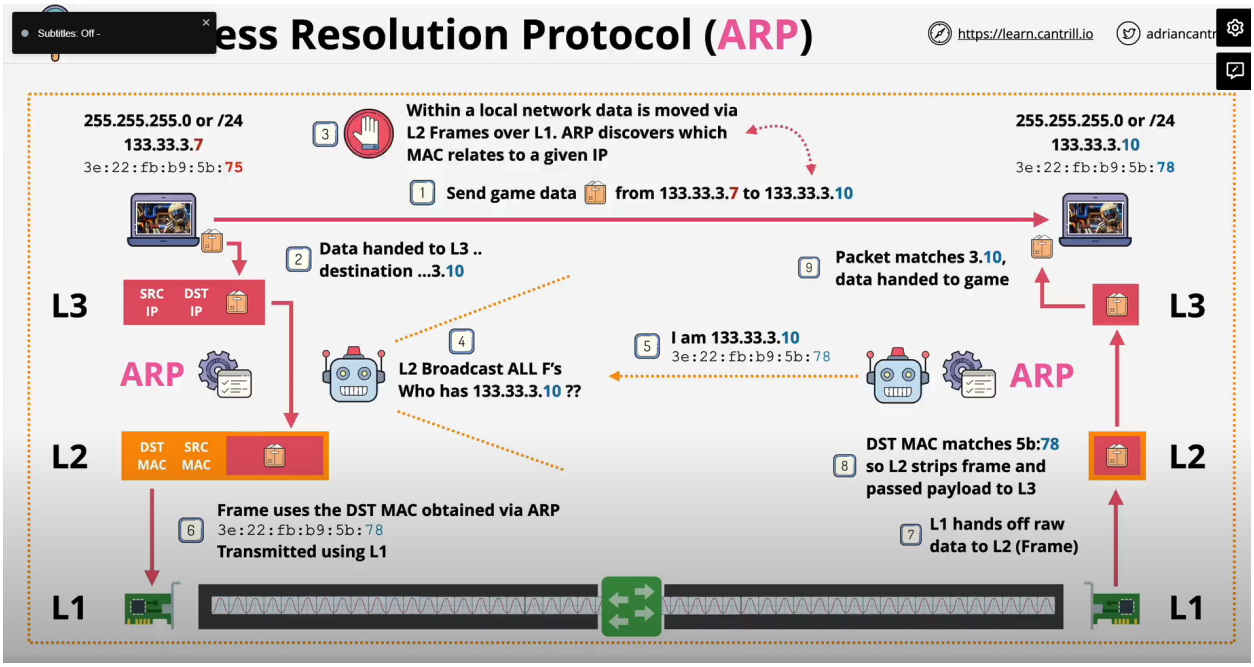


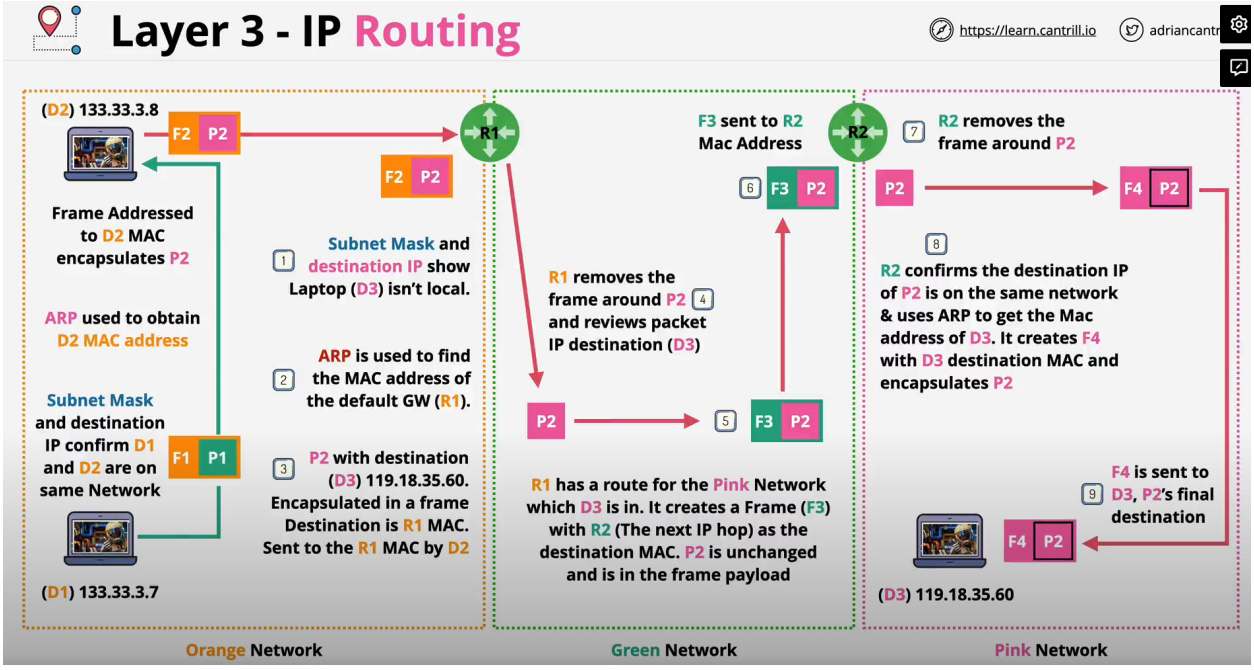


Route Table & routes

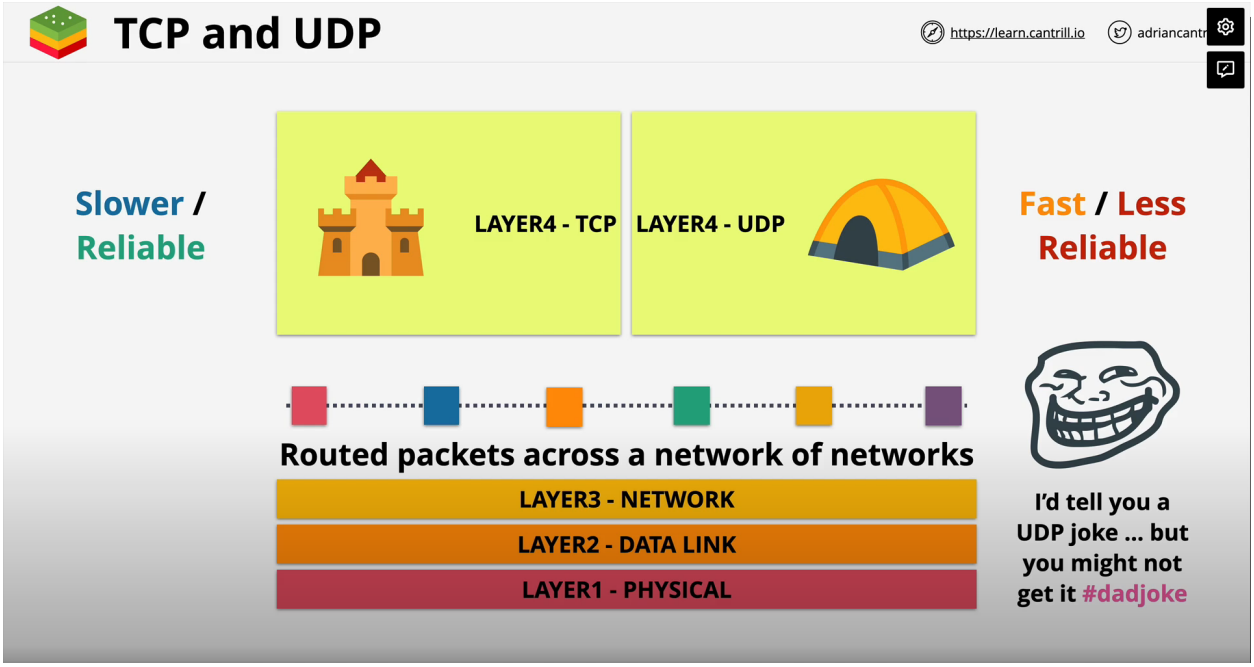


ARP: Address Resolution Protocol





Layer 4: Transport layer



TCP header

segments are only for TCP

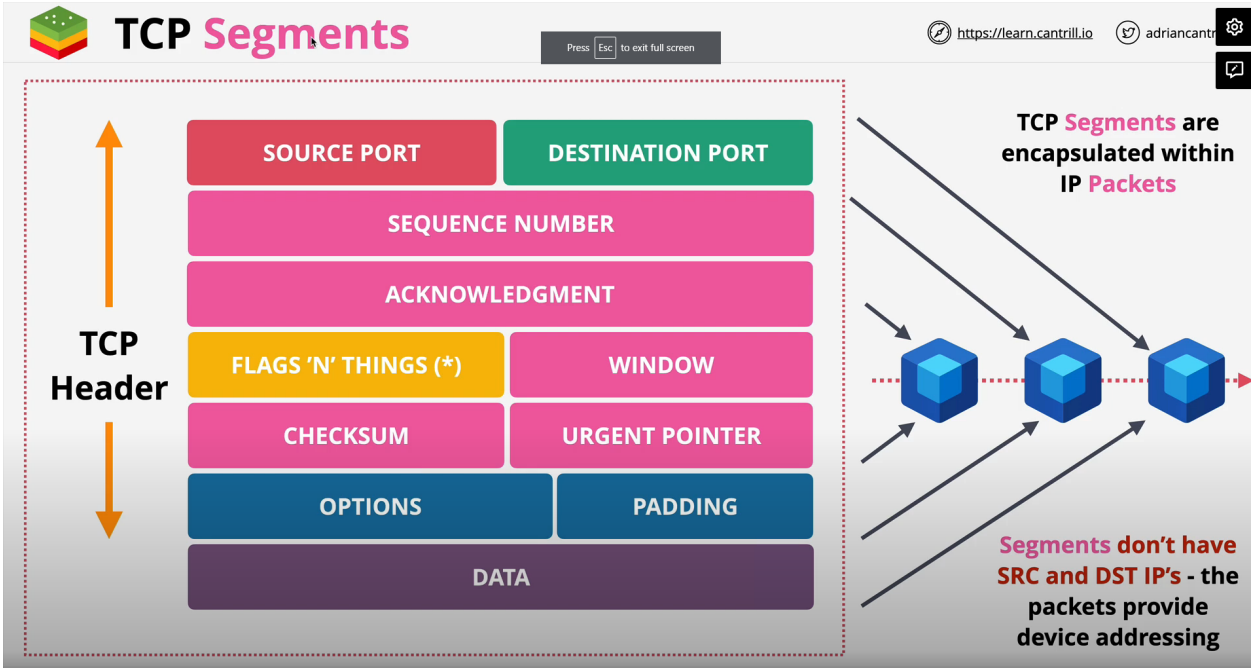
src port n destination port for sending and receiving data on the port

sequence no: is used to keep track of the sequence which is incremental and used to know if we are missing any packages for example lets say we sent seq 1-80 first and the destination has to acknowledge that he has received package 1-80

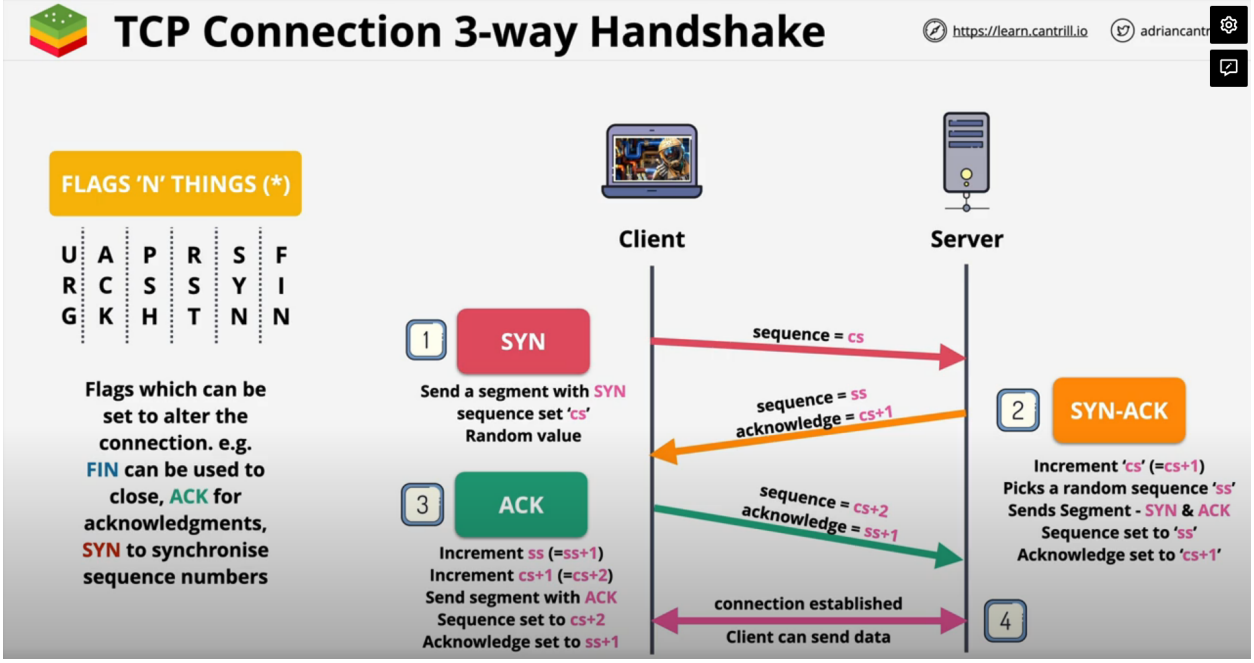
flag n things consist of flags n things which is used to close the connection

window is used to flow control based on receivers speed to avoid the traffic

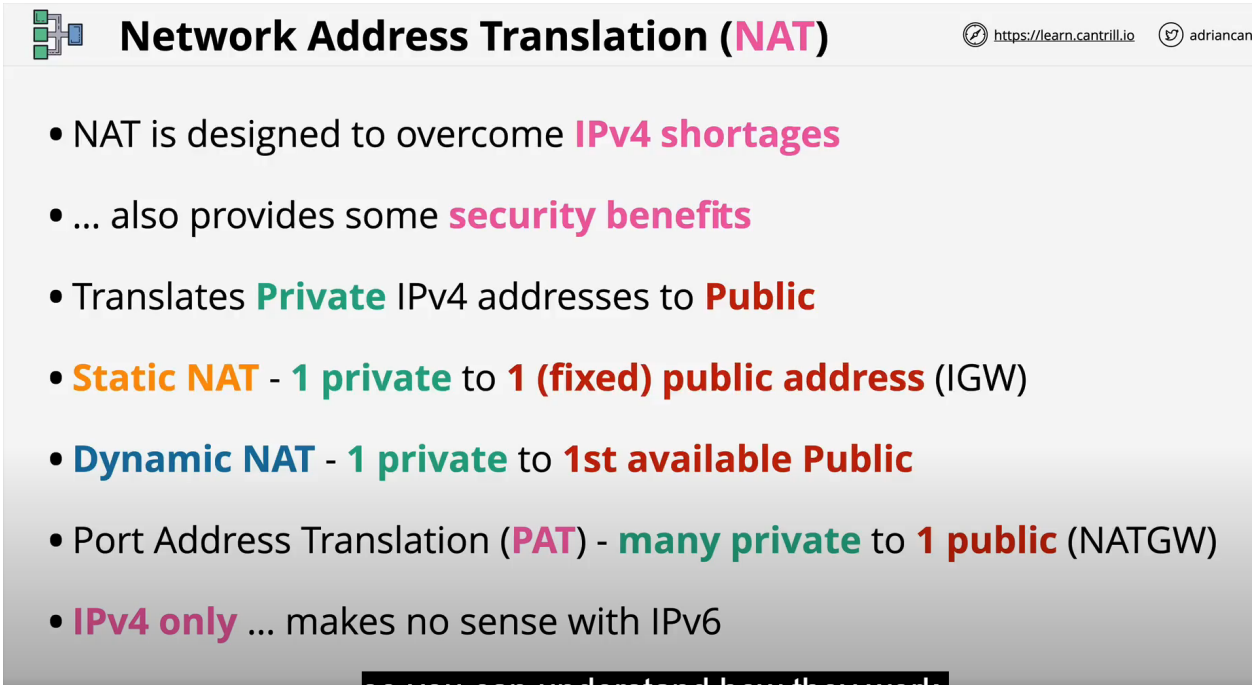
checksum: error checking



3 way handshake for TCP



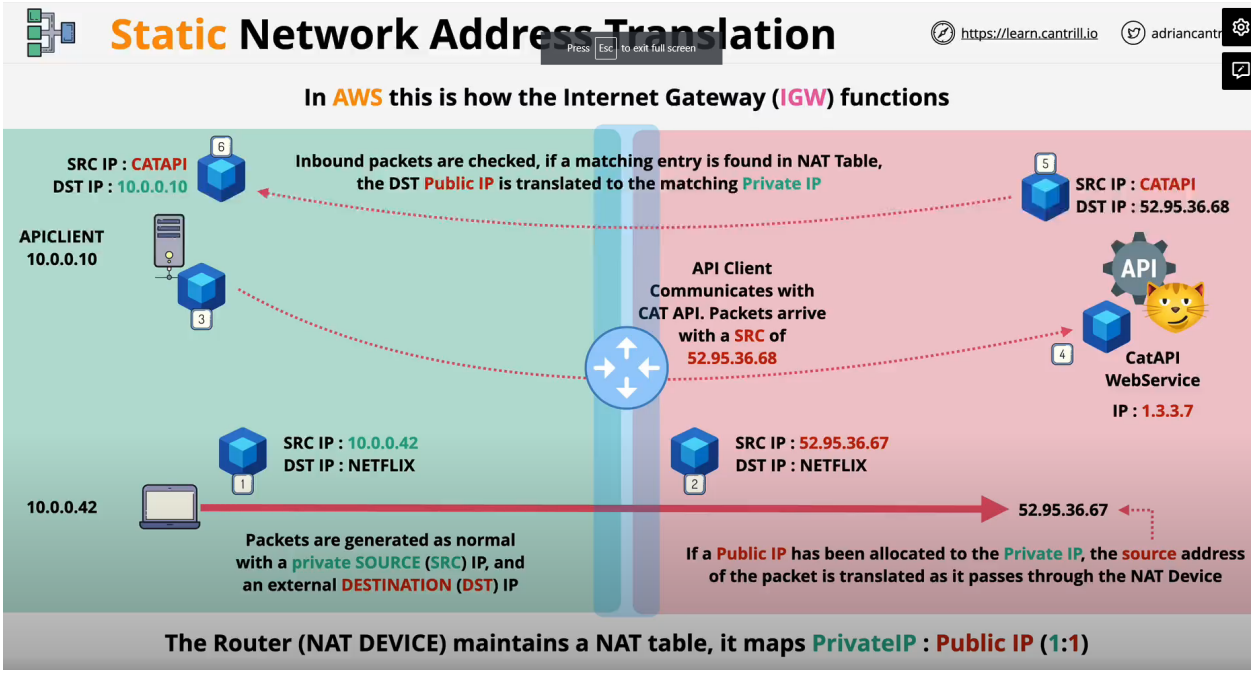
NAT : Network address translation



static NAT:

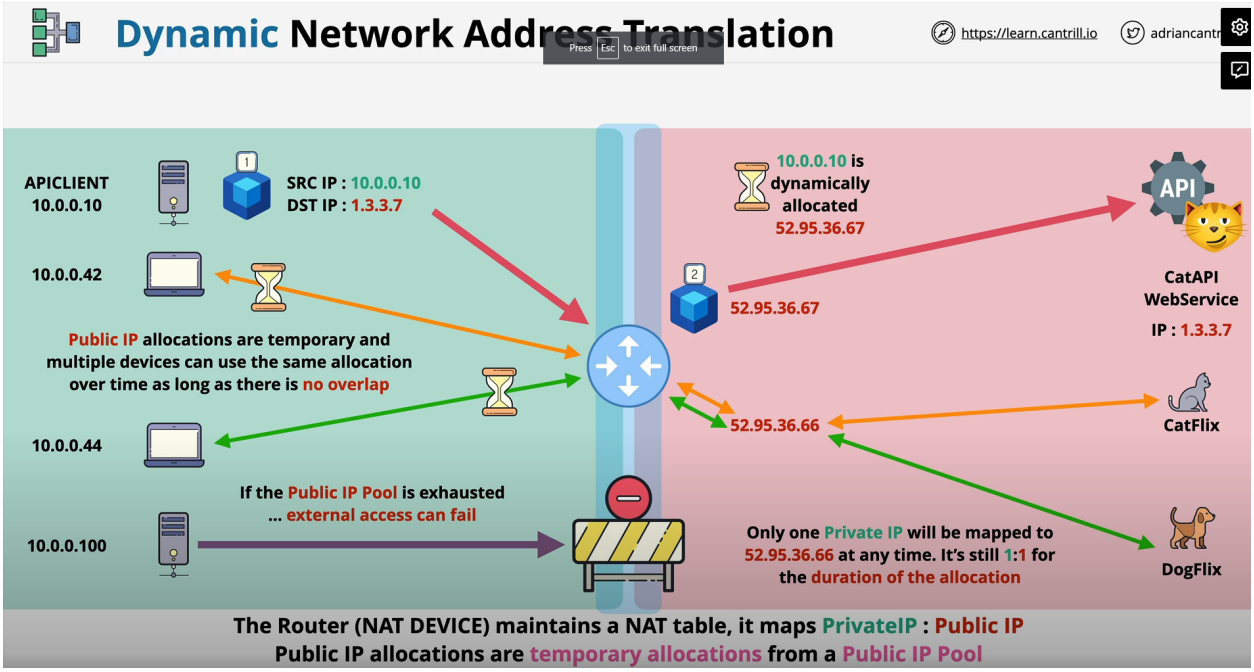
static NAT uses different public ip for different different private IP

and NAT table keeps the log



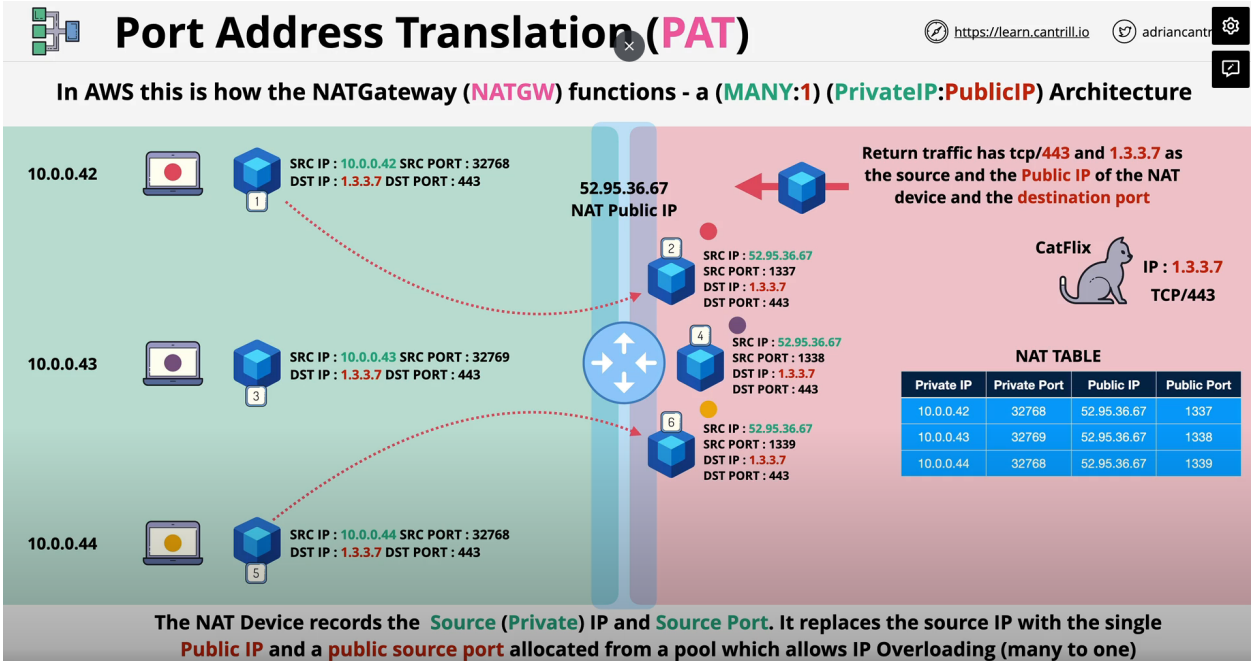
dynamic NAT: Uses number of Public address for multiple private IP

this can be used when we have limited amount of public IP available



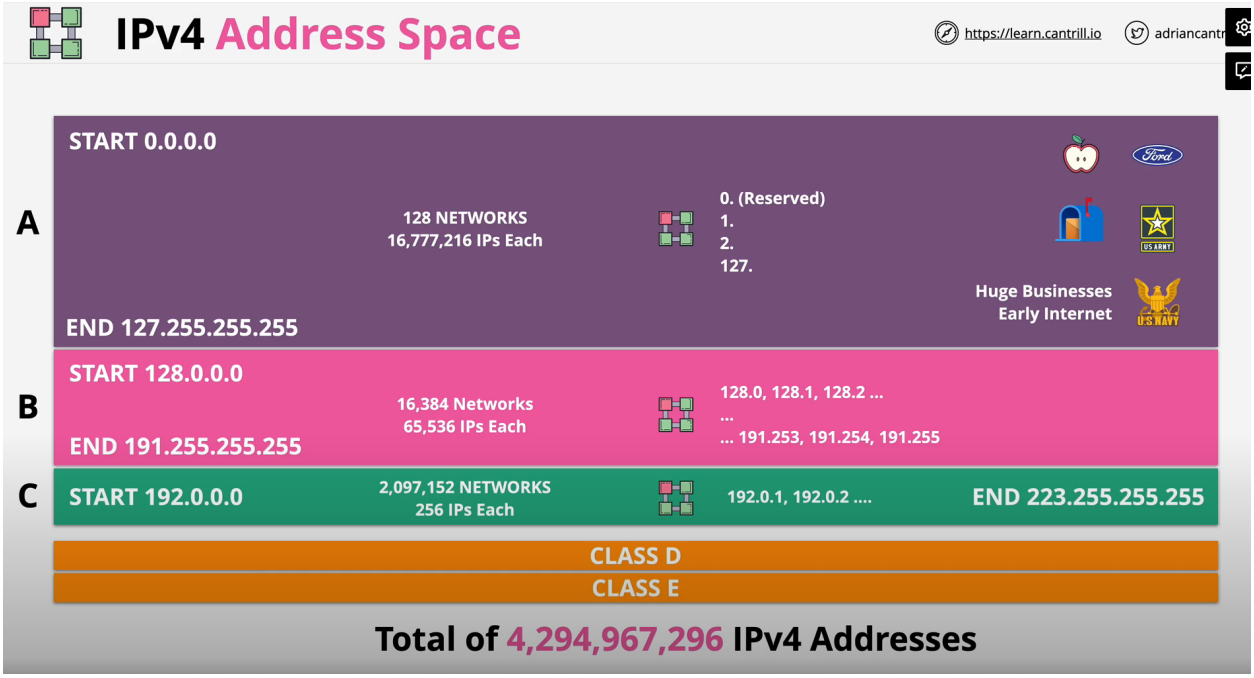
PAT: Port Address Translation

multiple private IP uses same public ip

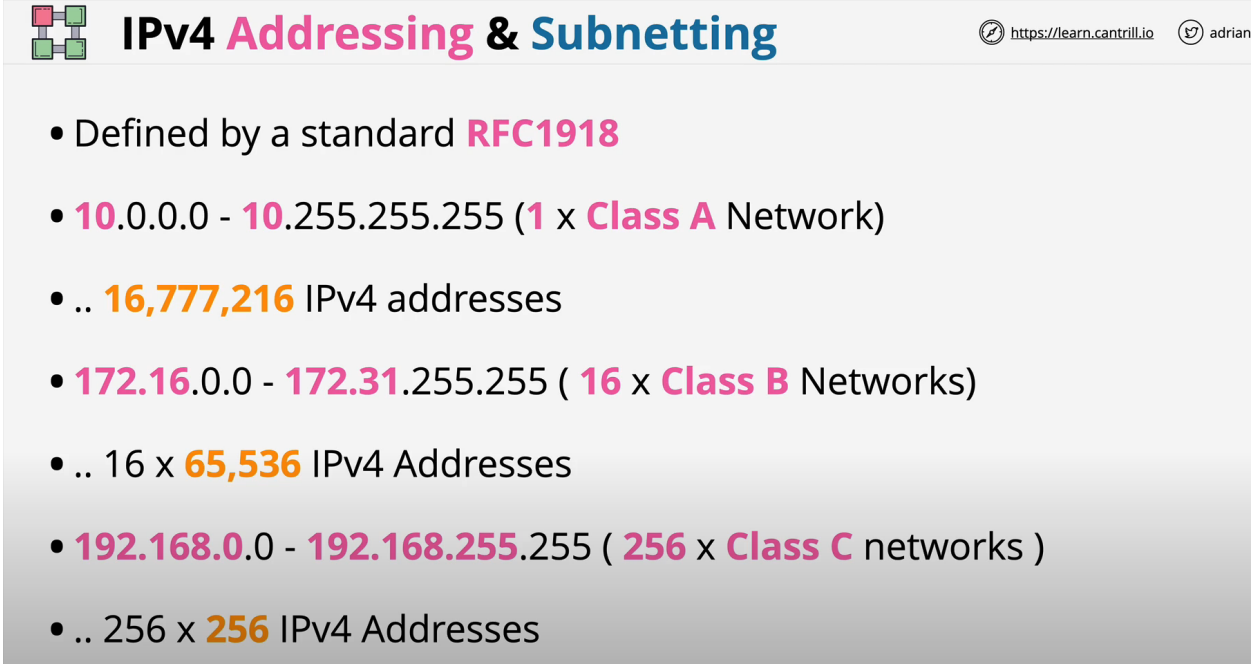


IP n Subnetting

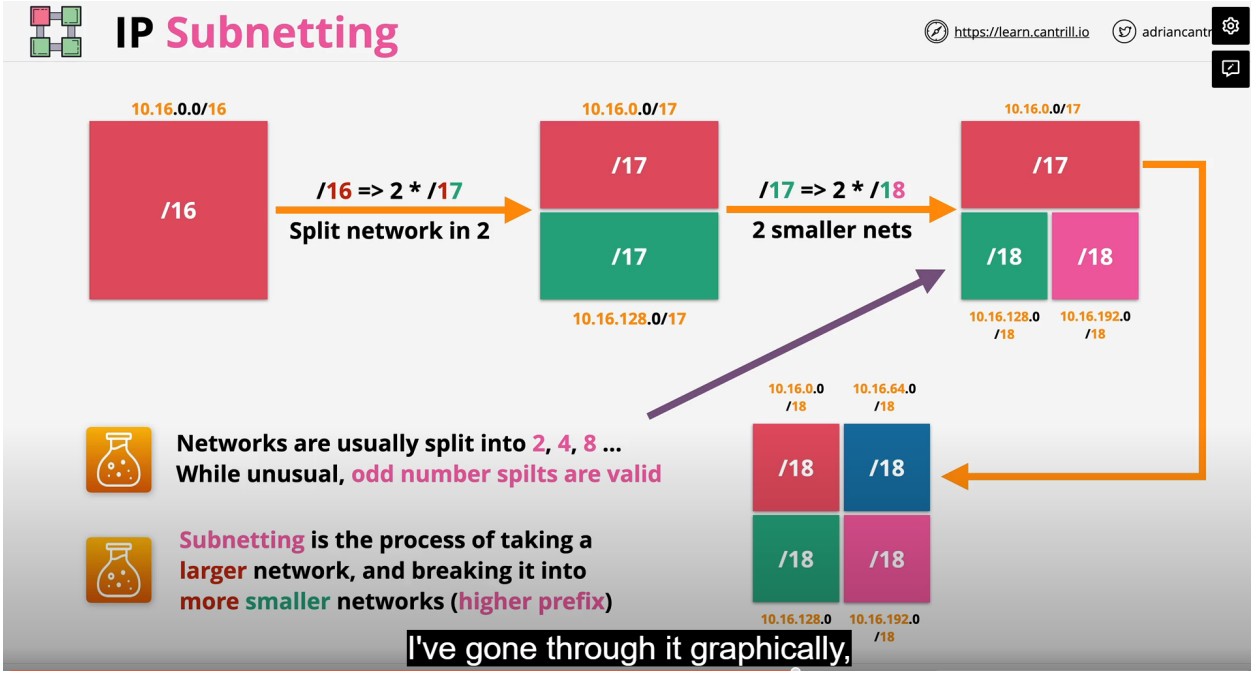
public network



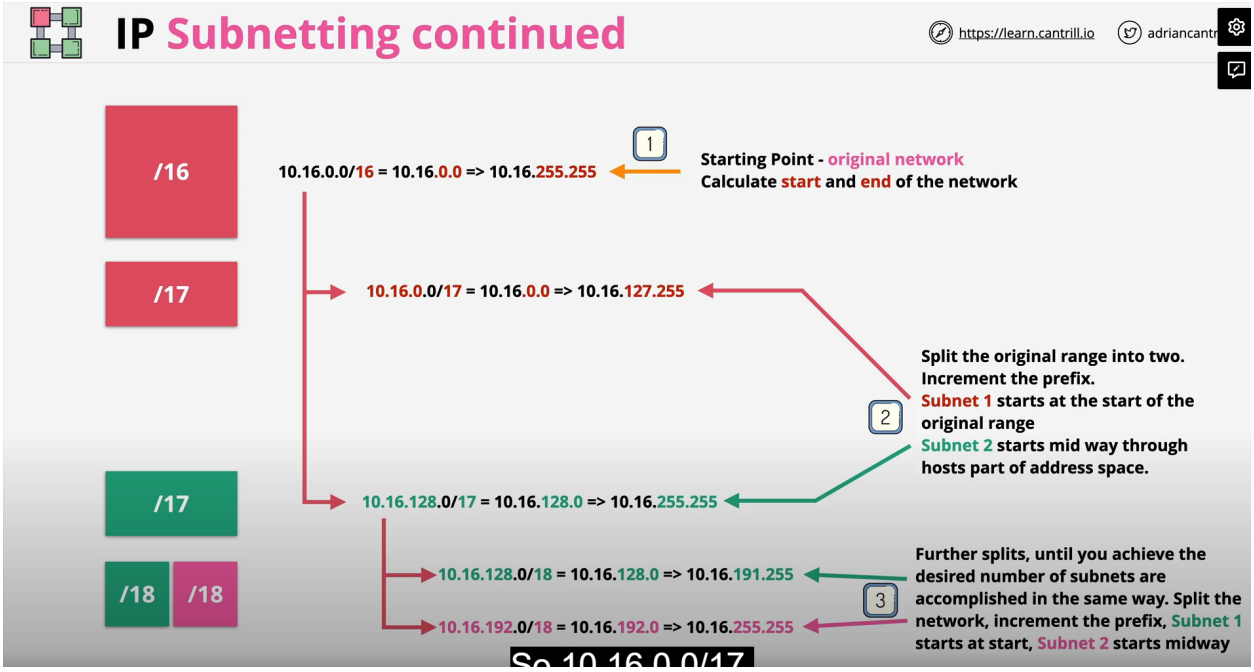
Private network



subnet



example

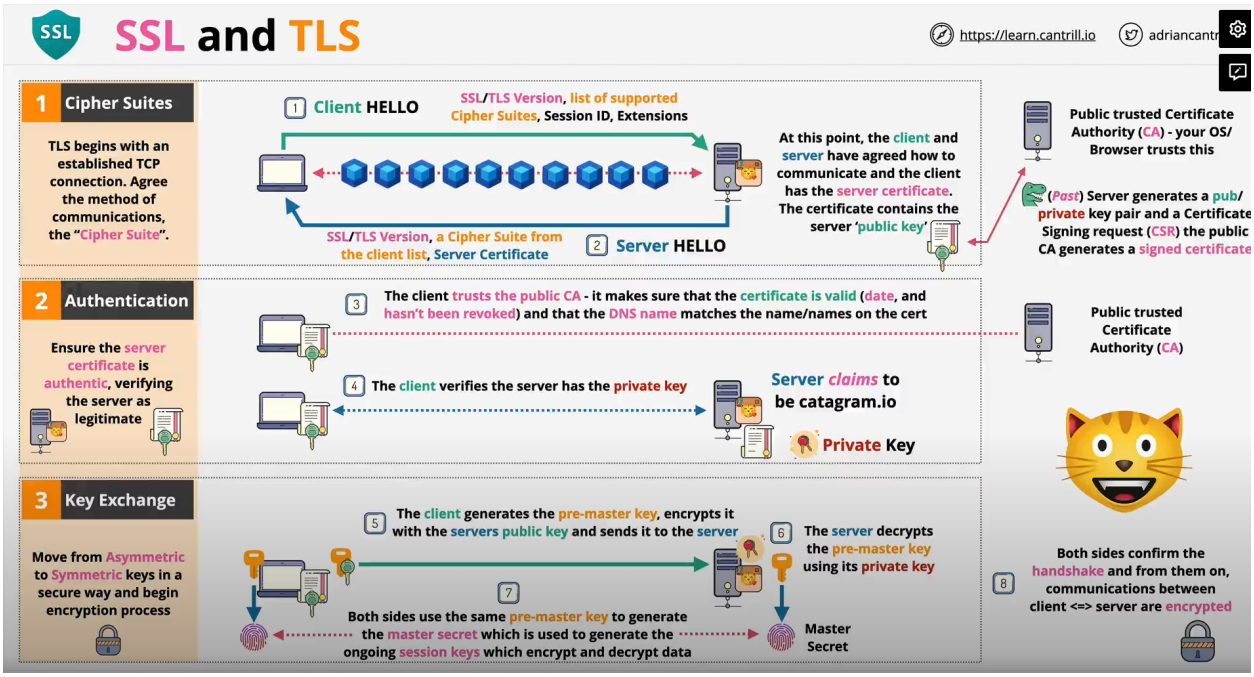


SSL AND TLS

SSL and TLS

SSL:Secure socket layer

TLS: Transport Layer Security


SSL and TLS
https://learn.cantrill.io
adriancantr
1 Cipher Suites
TLS begins with an established TCP connection. Agree the method of communications, the "Cipher Suite".
1 Client HELLO
SSL/TLS Version, list of supported Cipher Suites, Session ID, Extensions
SSL/TLS Version, a Cipher Suite from the client list, Server Certificate
2 Server HELLO
At this point, the client and server have agreed how to communicate and the client has the server certificate. The certificate contains the server 'public key'
Public trusted Certificate Authority (CA) - your OS/ Browser trusts this
(Past) Server generates a pub/ private key pair and a Certificate Signing request (CSR) the public CA generates a signed certificate
2 Authentication
Ensure the server certificate is authentic, verifying the server as legitimate
3 The client trusts the public CA - it makes sure that the certificate is valid (date, and hasn't been revoked) and that the DNS name matches the name/names on the cert
Public trusted Certificate Authority (CA)
4 The client verifies the server has the private key
Server claims to be catagram.io
Private Key
3 Key Exchange
Move from Asymmetric to Symmetric keys in a secure way and begin encryption process
5 The client generates the pre-master key, encrypts it with the servers public key and sends it to the server
6 The server decrypts the pre-master key using its private key
7
Both sides use the same pre-master key to generate the master secret which is used to generate the ongoing session keys which encrypt and decrypt data
Master Secret
8 Both sides confirm the handshake and from them on, communications between client <=> server are encrypted