# Reference links

<https://medium.com/@skshukla.0336/mtls-everything-you-need-to-know-e03804b30804>

Entity – here entity means server or client

## **Certificate Request**

A Client or Server in order to acquire a valid certificate first need to create a **CSR** (Certificate Request) file which they need to submit to the **Certificate Authority** which can sign and issue a valid certificate

## 2.2 Certificate Authority (CA)

Simple, authority means power, here certificate Authority means the power to issue the certificate,

So certificate authority is the one who will Issue the Certificates to the Entities (Servers, Clients, Websites)

CA sometime may include a physical visit by them to the business premises and then issue the public certificate

And certificate authority is the one who can confirm whether it’s the same Website which it claims it is.

A Certificate Authority is a trusted organization who confirm the authenticity of a website (Server). Their primarily responsibilities are,

Few of the Certificate Authorities are [VeriSign](https://www.verisign.com/), [DigiCert](https://www.digicert.com/" \t "_blank).

## 2.3 Certificate

A (public) certificate will be issued to server or client,

Since CA issues the certificates the CA can verify whether it’s valid certificate or not or whether it is signed by them or not

## 2.3 SSL vs TLS

TLS= more updated version of SSL, but both will do same thing of encrypting the data transfer

SSL (Secure socket layer) is standard technology for securing an internet connection by **encrypting the data** that is being sent between website and browser

It prevents hackers from seeing or stealing any information transferred

TLS (Transport layer security) is a protocol that helps secure communication over computer network, **TLS is also used to encrypt the communication** between web app and server

So data is always encrypted during transit

Both functionalities are same, but internal working implementations are different, like SSL uses different hashing function and TLS uses different hashing function

SSL uses hashing function such as MD5, SDH-1 but these are old sometimes different for diff inputs result hash is same

Hi=110

Bye=110

Whereas TLS uses SHA-256 algorithm, diff inputs always gives diff hash values

|  |  |
| --- | --- |
| SSL | TLS |
| Uses data encryption standards DES, RC4 | Uses AES standard |
|  |  |
|  |  |

SSL, Stands for, Server Socket Layer however TLS, Stand for, Transport Layer Security. SSL is just the older name.

## 2.4 Public/Private keys

* **Public Key** is used to Encrypt the data. (Think of certificates. No harm passing the public key, certificate, to anyone publicly)
* **Private Key** is used to Decrypt the data, encrypted by Public key.
* Private Key is like a password which should never cross the boundary of a system who has generated it. Consider it very secure. Think of it a system password which you don’t even share with your manager or any other person in authority.

## 2.5 MTLS (Mutual TLS)/ 2 way authn

Mutual means both, so MTLS means both client and server will mutually verify others identity /other certificates,

Like client will verify server certificate and server will verify client certificate with the help of CA who can check the certificate

So in postman when u are hitting an MTLS URL u need to attach certificate

Because both server and client needs to verify others certificate

**How Client will validate server certificate???**

Simple client will go and ask the certificate authority (the one who issued the certificate) saying whether it had issued that certificate to that website or not

Generally in one way communication,

1. Client will verify the servers certificate to confirm whether its talking to right client or not
2. And server accepts all clients, whereas in MTLS server will also validate the client certificate

Whereas in MTLS, not only client, whereas server also will verify the client authenticity

1. While making a call to the server, Client uses its ***Certificate*** and ***Client*** ***Key*** for the communication which is validated at the Server side to ensure the identity of client. Server consults Trust Store whether it’s a valid client to be given access or not. Please note, ***Client*** ***Key*** does not travel over network, it’s a private key and used to decrypt the data over the TCP tunnel.

## 2.6 key store and trust store

Key store contains our key and certificate

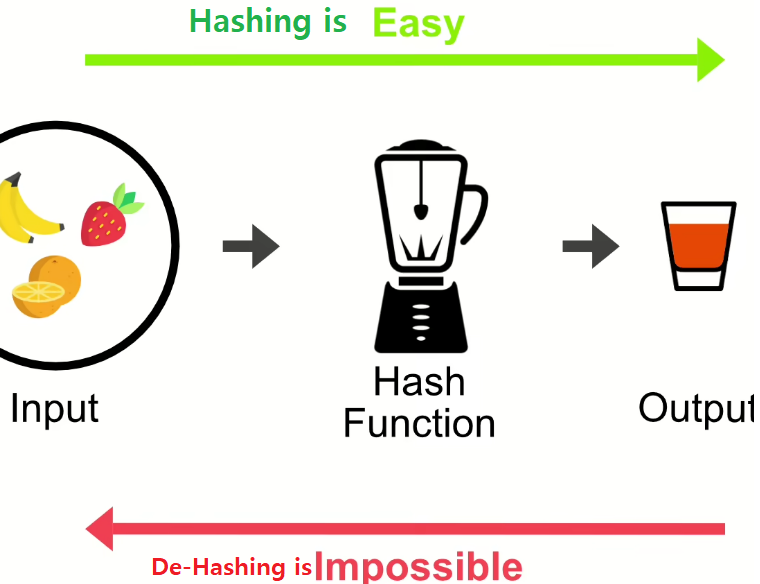
What is Trust store – trust store is used to trust only some websites

Importing other party appln to trust store

Assume if our appln want to allow 1dstr application, then we should import the 1dstr app certificate (we need to import the certificate that has been issued to 1dstr) into our server,

## Hashing and De hashing

Hashing is a one way cryptography and it is irreversible



Once u hash u cant reverse that, ex:- Hash 256, once passwords are hashed u can’t un hash that

Passwords hashed can’t be de-hashed/reversed/decrypted

## Encoding Vs. encryption vs. hashing

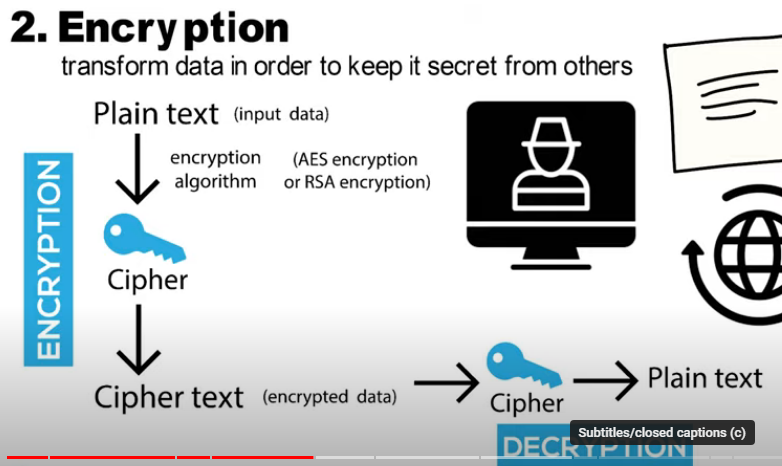
Encoding:- it’s a basic data representation to enable info **which doesn’t involve keeping secrets**

**In encoding security is not the primary concern**

**Encrypting: -** here data can’t be consumed by anyone other than intended recipients

It is to transform the data in order **to keep** it secret from others

**To encrypt and decrypt we need some key**



### Symmetric encryption vs. asymmetric encryption

Symmetric encryption:- With same key we will encrypt and decrypt

Asymmetric encryption: - we will encrypt with public key and we can only decrypt with private key

Everybody will have access to public key & only intended user will have private key