10th September, 2021

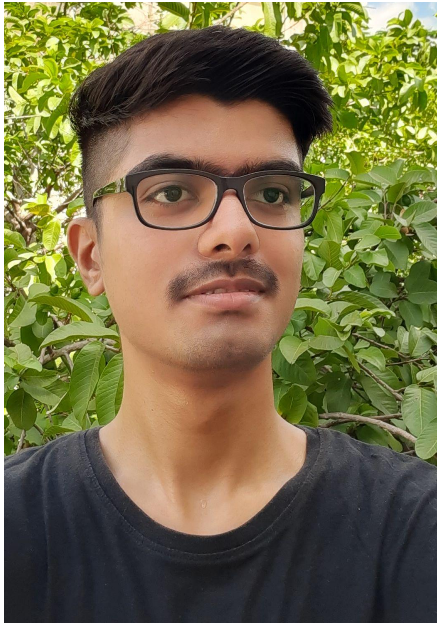
Thanos

TLS in Brief!

NamanLakhwani

Namanl2001

Naman Lakhwani



- IT Undergrad @IIIT Gwalior, India
- Contributor @Thanos | CNCF
- Google Summer of Code 2021 participant with Thanos
- Linux Foundation Mentee at Kubernetes

NamanLakhwani

Namanl2001

Agenda

- What is TLS ?
- How TLS works?
- What is mutual TLS ?
- Certificate Authority
- Why to use mutual TLS ?
- Mutual TLS in Thanos
- Certificate Rotation
- Q&A

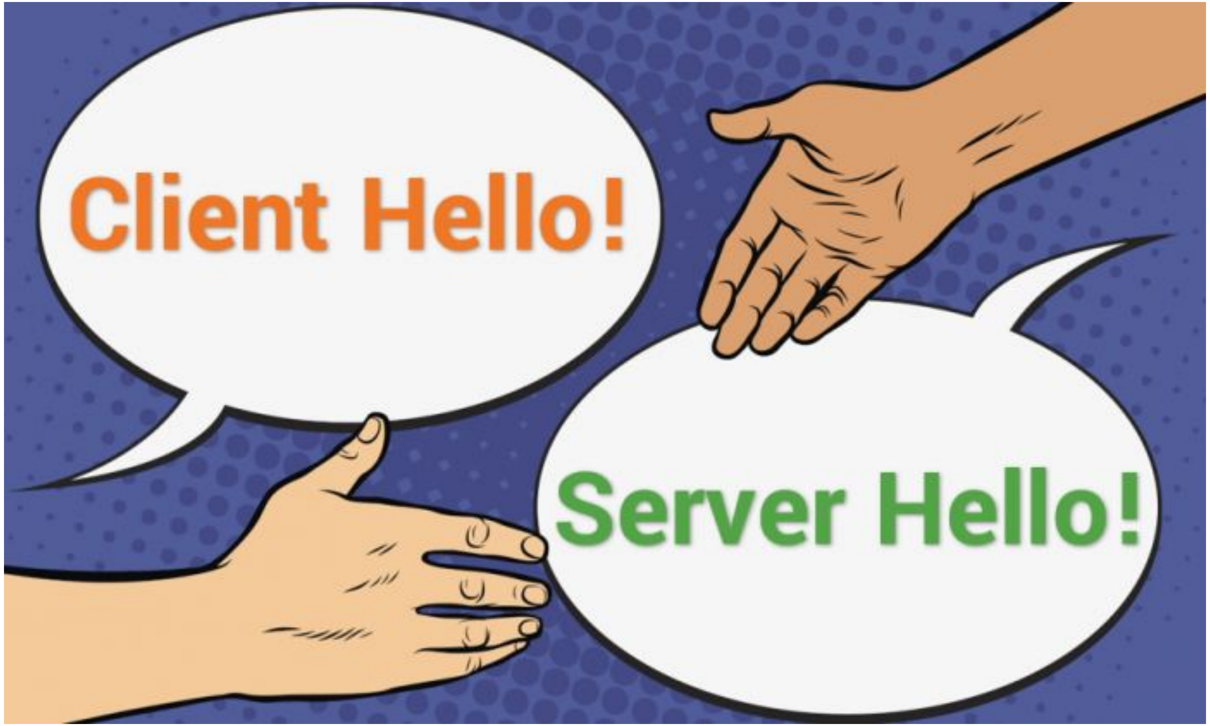
What is TLS ?

- TLS = Transport Layer Security
- Encryption protocol in wide use on the Internet
- Authenticates the server in a client-server connection and encrypts communications between client and server

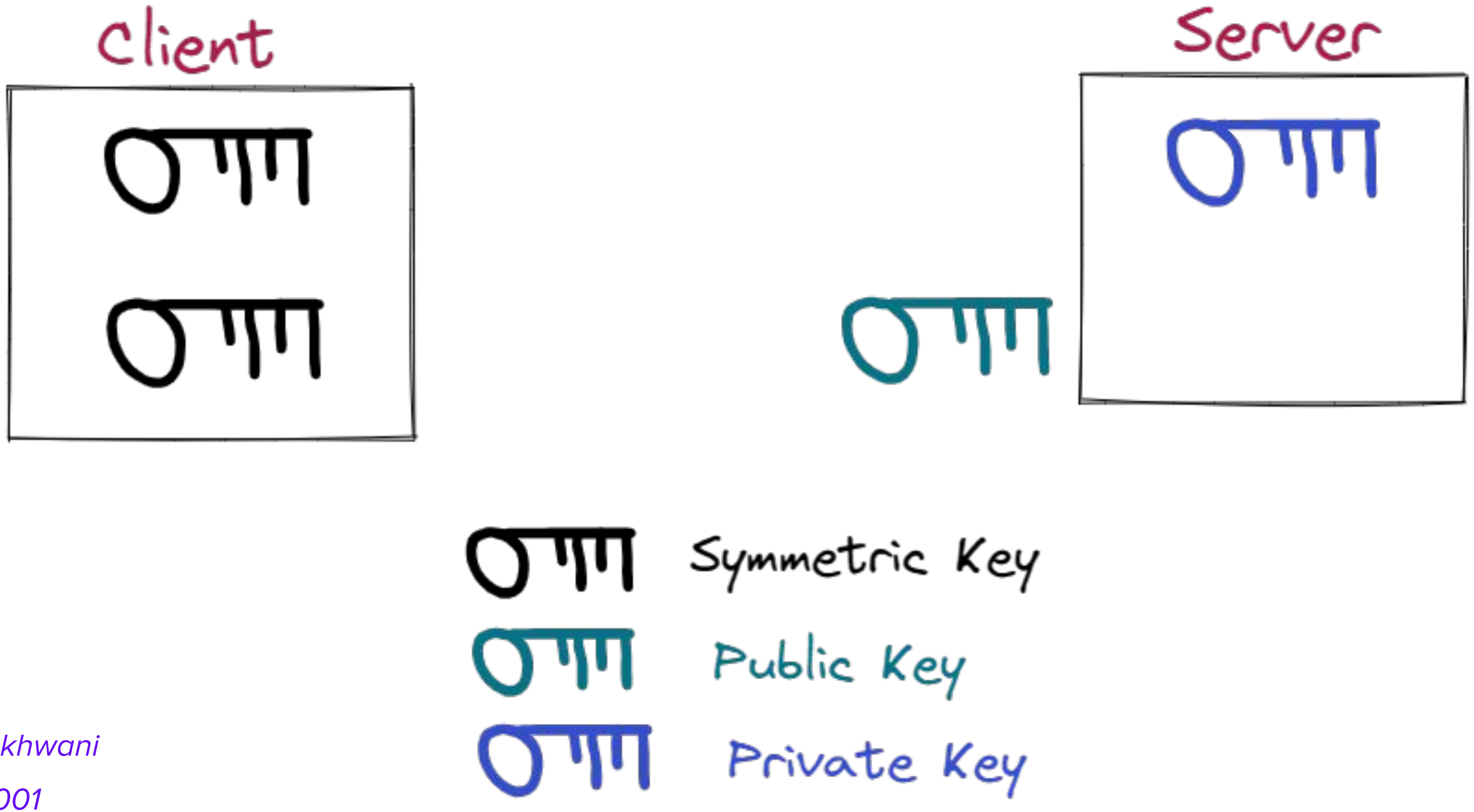
How TLS works ?

1. **Public key and Private key**
2. **TLS certificate** : data file that contains important information
3. **TLS handshake** : responsible for the authentication and key exchange necessary to establish secure sessions

TLS Handshake



TLS Handshake



NamanLakhwani

Namanl2001

What is mutual TLS (mTLS) ?

Method for mutual authentication.

“mTLS ensures that the parties at each end of a network connection are who they claim to be by verifying that they both have the correct private key.”

How does mTLS works ?

Additional steps :

- **Client presents its TLS certificate**
- **Server verifies the client’s certificate**
- **Server grants access**
- Client and server exchange information over encrypted TLS connection

Certificate Authority

“All trusted TLS certificates are issued by a Certificate Authority (CA), which is a company that has been approved to issue digital certificates.”

However, in the case of mutual TLS, the organization implementing mTLS can either use these CA’s or can act as their own certificate authority.

The root certificate is self-signed, meaning that the organization creates it themselves.

Why use mTLS ?

mTLS helps ensure that traffic is secure and trusted in both directions between a client and server.

This provides an additional layer of security for the users.

mTLS prevents various kinds of attacks, including:

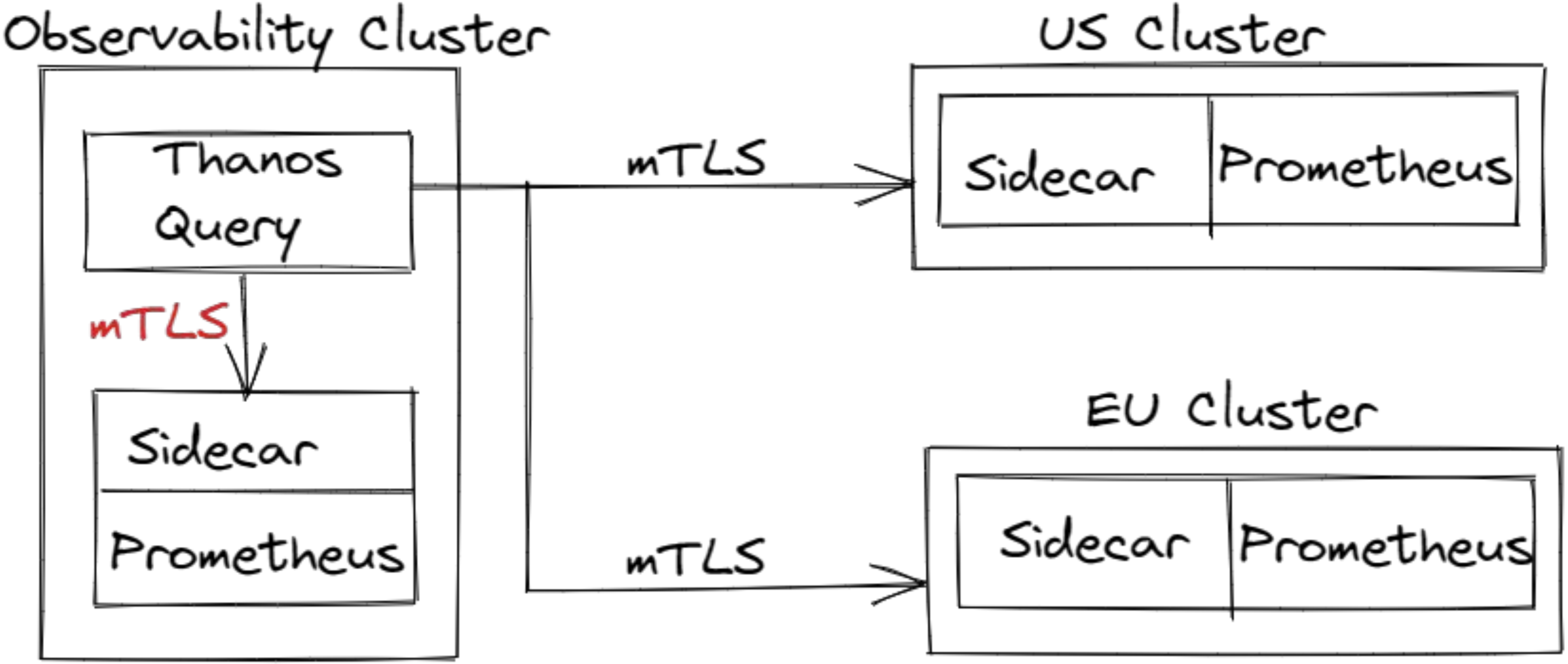
On-path attacks, Credential stuffing, Brute force attacks, Phishing attacks, etc.

Websites already use TLS, so why is mTLS not used on the entire Internet?

- The goals of TLS on the public Internet are
  - to ensure that people do not visit fake websites
  - to keep private data secure as it crosses the various networks
  - to make sure that data is **not** altered in transit

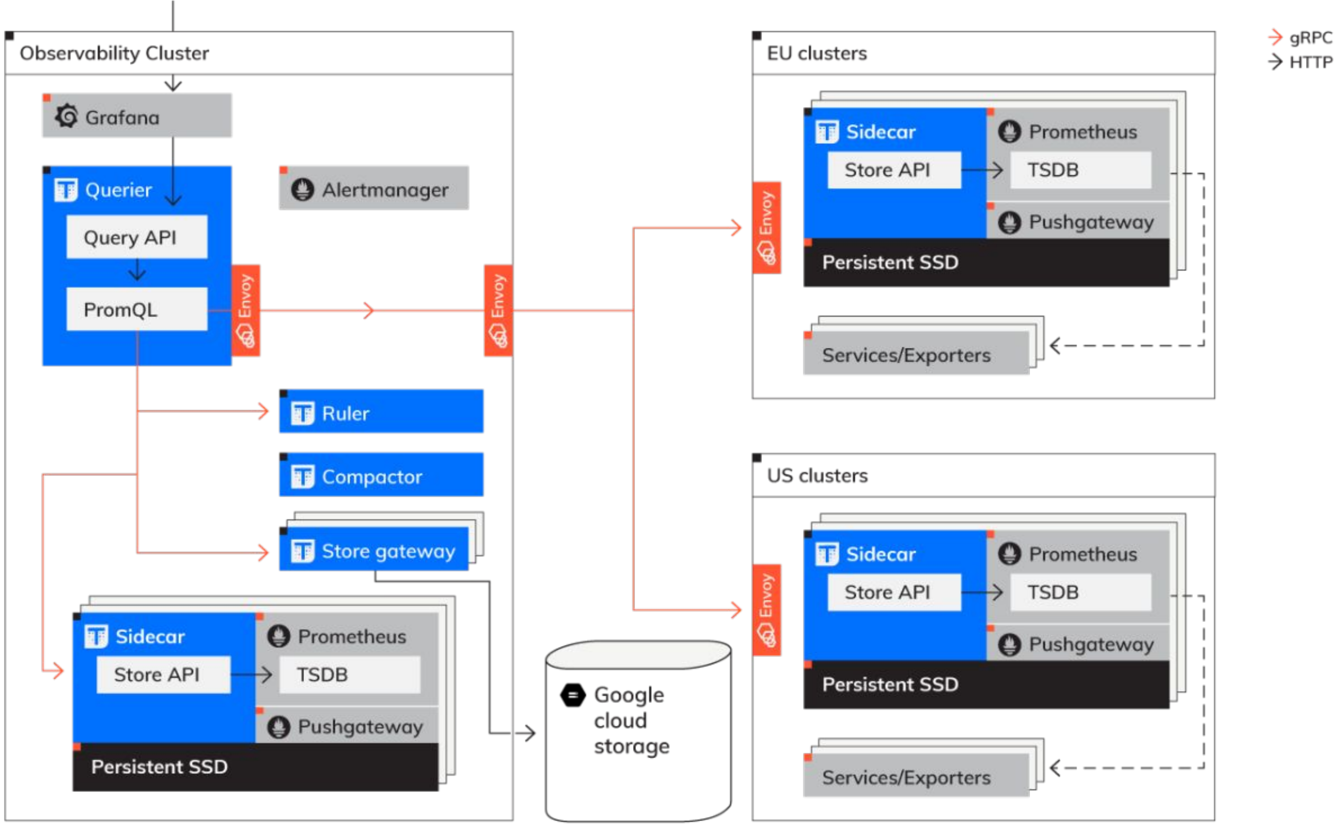
- Besides, distributing TLS certificates to all end user devices would be extremely difficult.

mTLS in Thanos



NamanLakhwani

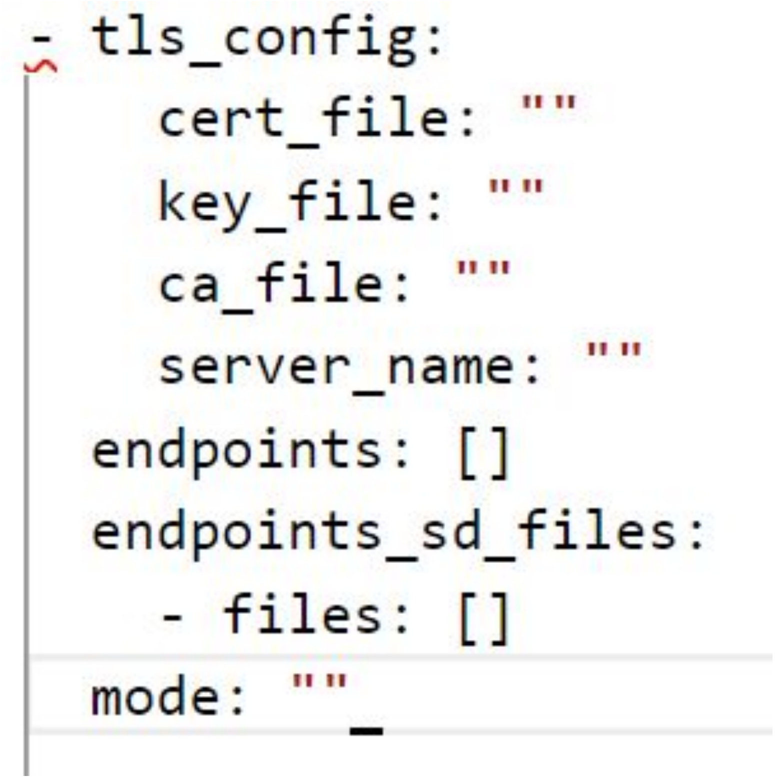
Namanl2001



NamanLakhwani

Namanl2001

--endpoint.config



```
- tls_config:
    cert_file: ""
    key_file: ""
    ca_file: ""
    server_name: ""
  endpoints: []
  endpoints_sd_files:
    - files: []
  mode: ""
```

Certificate Rotation

Certificate Rotation: means the replacement of existing certificates with new ones (renewing of certificate)

It is needed when:

- Certificate is expired
- Private key is leaked
- Wants to change the CA.
- Etc....

Curious to read and learn more about TLS ?

Read my blog: https://namanlakhwani.tech/posts/202108-tls-in-brief/

Head over to this awesome article on TLS:
https://www.thesslstore.com/blog/explaining-ssl-handshake/

Resources:

- https://www.cloudflare.com/en-gb/learning/ssl/transport-layer-security-tls/
- https://www.cloudflare.com/en-gb/learning/ssl/what-happens-in-a-tls-handshake/
- https://www.cloudflare.com/en-gb/learning/access-management/what-is-mutual-tls/

NamanLakhwani

Namanl2001

Thanks for your attention!

NamanLakhwani

Namanl2001

TLS Handshake

- Cipher suites : negotiate and agree on the exact encryption method (based on their capabilities).
- Client sends the “clientHello” message to the server.
- Server responds back with the “serverHello ” message and the TLS certificate
- Client verifies the certificate, encrypts the session key with the server public key and sends it to the server.
- Server decrypts the received message using it’s private key.
- Voila! Now both client and server can communicate securely using the shared symmetric session keys .