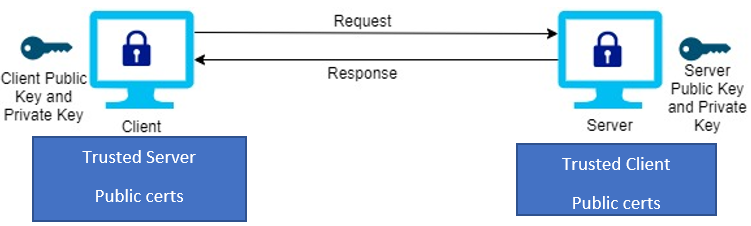
Mule 2Way TLS authentication/Mutual Authentication

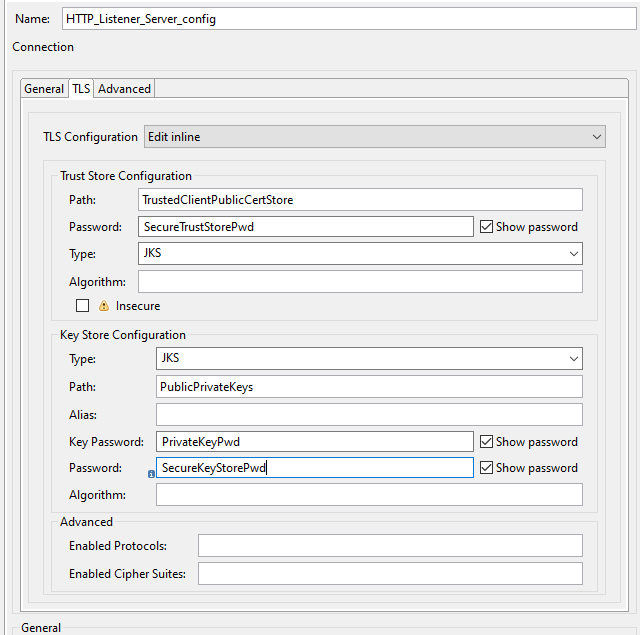
TLS (transport layer security) is security protocol that Mule supports through v1.1 and v1.2. It helps exchange keys between client and server, encrypt data and guaranteed message integrity. 2Way TLS authentication sometimes called as **mutual authentication**, where TLS validation happens on both side (Client and server).

Depending on the business scenario- Mule application can serve as 2way TLS authentication server side (exposing the http listener mule component) or 2 way TLS authenticator client side ( using http requester mule component). Both mule roles are described as follows. Pls refer following diagram



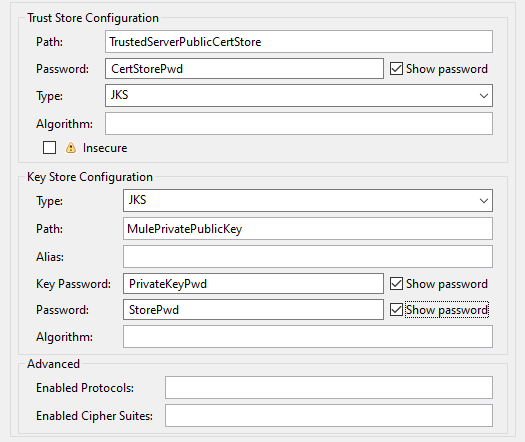
Mule application as the server- with 2way TLS authentication enabled- (i.e. handling client requests)

1. Mule exposes the http listener component accept request on https port
2. Mule gets the CA cert and private key by generating the CSR (Certificate signing request)
3. Mule http listener needs to configure TLS truststore and keystore both
4. TLS truststore contains the trusted client public certs (it can contain the keys but by convention will contain only trusted client public certs)
5. Keystore contains the public/private key of the server ( will contain server private and public key)
6. Server private key never leaves the server
7. Keypassword: is for private key protection password while **password:** is for store password (which contains more than one trusted client certs. Pls refer following screen in detail)
8. Client needs the mule public cert \*.crt to their client truststore (e.g. windows machine certificates)



Mule application as the client- with 2way TLS authentication enabled- (i.e. placing request to provider)

1. Mule uses the http request component to place request over https port
2. Mule gets the CA cert and private key by generating the CSR (Certificate signing request)
3. Mule http request needs to configure TLS truststore and keystore both
4. TLS truststore contains the trusted server public certs (it can contain the keys but by convention will contain only trusted server public certs)
5. Keystore contains the public/private key of the client (In this case mule private and public key)
6. mule private key never leaves the server where mule application is hosted (onprem or cloudhub)
7. Server needs public cert of the mule (clients) in their truststore



External reference Articles:

<https://docs.mulesoft.com/mule-runtime/4.2/tls-configuration>

<https://help.mulesoft.com/s/article/HTTPS-Two-Way-Authentication-Example-in-Mule-4>