Docker tls Certification Configuration

Step 1: Generate SSL/TLS Certificates

1. **Create a directory for your certificates**:

   ```
   mkdir -p /etc/docker/certs
   cd /etc/docker/certs
   ```

2. **Generate a CA private key**:

   ```
   openssl genrsa -aes256 -out ca-key.pem 4096
   ```

   Note : This command creates a private key for your Certificate Authority (CA) with 4096 bits and encrypts it with AES-256.

3. **Generate a CA certificate**:
   openssl req -new -x509 -days 365 -key ca-key.pem -sha256 -out ca.pem
   Note : You'll be prompted to enter details like country, state, organization, etc. This creates a self-signed certificate valid for 365 days.
4. **Generate a server key**:
   openssl genrsa -out server-key.pem 4096
   Note : This creates a private key for the Docker server with 4096 bits.
5. **Create a certificate signing request (CSR) for the server certificate**:
   openssl req -new -key server-key.pem -out server.csr
   Note : You'll be prompted for details. This CSR will be signed by your CA to create the server certificate.

6. **Create an extension file for the server certificate**:

   ```
   touch extfile.cnf
   ```

   Paste : subjectAltName = IP:192.168.0.90 extendedKeyUsage = serverAuth

7. **Sign the server certificate with the CA certificate**:

   ```
   openssl x509 -req -days 365 -sha256 -in server.csr -CA ca.pem -CAkey ca-key.pem
   -CAcreateserial -out server-cert.pem -extfile extfile.cnf
   ```

   Note : This command creates a server certificate signed by your CA, valid for 365 days.

8. **Secure the keys and certificates**:

   ```
   chmod -v 0400 ca-key.pem server-key.pem

   chmod -v 0444 ca.pem server-cert.pem
   ```

Note : These commands set appropriate permissions on your key and certificate files.

Step 2: Configure Docker to Use SSL/TLS Certificates

1. **Move the certificates to the Docker directory**:

mkdir -p /etc/docker

cp server-cert.pem /etc/docker/

cp server-key.pem /etc/docker/

cp ca.pem /etc/docker/

2. **Edit the Docker service file**:

   Open the Docker service file:

   sudo nano /lib/systemd/system/docker.service

   Find the `ExecStart` line and modify it to include the TLS options:

   ```
   ExecStart=/usr/bin/dockerd --tlsverify
   --tlscacert=/etc/docker/ca.pem
   --tlscert=/etc/docker/server-cert.pem
   --tlskey=/etc/docker/server-key.pem -H=0.0.0.0:2376
   ```

   Note : This tells Docker to use TLS and specifies the paths to your CA, server certificate, and key. Replace `0.0.0.0` with your server's IP address (`192.168.0.90`) if needed.

3. **Reload the systemd daemon and restart Docker**:

   sudo systemctl daemon-reload

   sudo systemctl restart docker

   This reloads the Docker service configuration and restarts Docker to apply the new settings.

Step 3: Configure the Client to Use SSL/TLS

1. **Copy the CA certificate to your client machine**:

   scp user@192.168.0.90:/etc/docker/ca.pem ~/

2. **Generate a client key and certificate**:

   openssl genrsa -out client-key.pem 4096

   openssl req -new -key client-key.pem -out client.csr

   Note : The first command creates a private key for the client, and the second generates a CSR for the client certificate.

3. **Create an extension file for the client certificate**:

   mkdir extfile-client.cnf

4. **Sign the client certificate with the CA**:

   openssl x509 -req -days 365 -sha256 -in client.csr -CA ca.pem -CAkey /etc/docker/certs/ca-key.pem -CAcreateserial -out client-cert.pem -extfile extfile-client.cnf

   This creates a client certificate signed by your CA.

5. ## Step 4: Test the SSL/TLS Setup

   Try running Docker commands from your client machine to ensure it connects properly using TLS. For example:

   docker --tlsverify --tlscacert=ca.pem --tlscert=client-cert.pem --tlskey=client-key.pem -H=192.168.0.90:2376 info