Secure Server-Client Communication with OpenSSL

Assignment 1

Description

This project implements a **secure client-server communication** using **TLS 1.2** and **OpenSSL in C**. Both server and client authenticate each other using **X.509 certificates** signed by a trusted CA. A rogue client (rclient) attempts to connect using an untrusted certificate and should not have a connection.

Files

- `server.c` — TLS server
- `client.c` — TLS client
- `rclient.c` — Rogue client (untrusted certificate)
- `Makefile` — Builds all executables
- `README.md` — This file

Compilation

```
make
```

Running

1. Start the server:

   ```
   ./server 8082
   ```

   Here **8082** is the listening port.

2. Start the client:

   ```
   ./client 127.0.0.1 8082
   ```

   - `127.0.0.1` is our localhost
   - `8082` = is our server port

3. Start the rogue client (untrusted):

   ```
   ./rclient 127.0.0.1 8082
   ```

OpenSSL Certificate Generation

1. Create Certificate Authority (CA)

We used this command to create a CA

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
  -keyout ca.key -out ca.crt \
  -subj "/C=GR/ST=Crete/L=Chania/O=TUC/OU=ECE/CN=RootCA"
```

Specifically

- `-x509` : generate a self-signed CA certificate
- `-nodes` : we do not use password protection

- `-newkey rsa:2048` : generate new 2048-bit RSA key
- `-days 365` : validity for 1 year
- `-subj` : sets the subject fields directly

---

2. Generate Server Certificate (signed by CA)

```
 openssl req -new -newkey rsa:2048 -nodes \
  -keyout server.key -out server.csr \
  -subj "/C=GR/ST=Crete/L=Chania/O=TUC/OU=ECE/CN=localhost"

openssl x509 -req -in server.csr -CA ca.crt -CAkey ca.key \
  -CAcreateserial -out server.crt -days 365 -sha256
```

---

3. Generate Client Certificate (signed by CA)

```
 openssl req -new -newkey rsa:2048 -nodes \
  -keyout client.key -out client.csr \
  -subj "/C=GR/ST=Crete/L=Chania/O=TUC/OU=ECE/CN=client"

openssl x509 -req -in client.csr -CA ca.crt -CAkey ca.key \
  -CAcreateserial -out client.crt -days 365 -sha256
```

---

4. Generate Rogue Client Certificate (signed by another CA)

```
 openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
  -keyout rca.key -out rca.crt \
  -subj "/C=GR/ST=Crete/L=Chania/O=FakeCA/OU=ECE/CN=FakeRoot"

openssl req -new -newkey rsa:2048 -nodes \
  -keyout rclient.key -out rclient.csr \
  -subj "/C=GR/ST=Crete/L=Chania/O=FakeClient/OU=ECE/CN=rclient"

openssl x509 -req -in rclient.csr -CA rca.crt -CAkey rca.key \
  -CAcreateserial -out rclient.crt -days 365 -sha256
```

Example Interaction

Valid Client:

Request:

```
 <Body>
<UserName>Sousi</UserName>
<Password>123</Password>
</Body>
```

Response:

```
 <Body>
<Name>sousi.com</Name>
<year>1.5</year>
<BlogType>Embedede and c c++</BlogType>
<Author>John Johny</Author>
</Body>
```

Rogue Client:

Response:

```
peer did not return a certificate or returned an invalid one
```

Author

Μυλωνάκης Χαράλαμπος Αγησίλαος Φωτεινάκης ECE — Technical University of Crete
Assignment 1, 2025