Collin Gros
04/15/2020

PROG2-1

GENERATING THE CERTIFICATE AND PRIVATE KEY

I had difficulty getting a certificate to generate from the cacert.org website, so I used an openssl command to generate one for me. This command also output the private key into its own .pem file:

```
openssl req -x509 -newkey rsa:4096 -nodes -keyout privkey.pem
-out cert.pem -days 365
```

EXTRACTING THE PUBLIC KEY FROM THE CERTIFICATE USING OPENSSL LIBRARY

To extract the public key from the certificate, I had to install the openssl libraries (libssl-dev), then create a makefile so that I may compile the C program including those libraries.

In my C file, 'extractPK.c', I first create BIO structs to read the certificate file and to write to STDOUT. Then, I attempt reading the cert file with `BIO_read_filename()`. I then read that BIO struct using openssl's X509 struct (since the cert was in PEM format). To do this, I used `PEM_read_bio_X509()`. Finally, to extract the public key, I used openssl's EVP struct (EVP_PKEY) to get the public key from the X509 struct using `X509_get_pubkey()`. Finally, I print the key to STDOUT using `PEM_write_bio_PUBKEY()`. I lastly clean up everything I allocated to avoid any memory leaks.

To save the output to a file, I piped STDOUT to a file, 'pubkey.pem'.