Course: Secure and Dependable Systems
Assignment 5

Fezile Manana

Problem 5.1

a) The sum of my matriculation number (30001426) and 164772993 = 194774419

b) A PGP revocation certificate, once uploaded to a keyserver, revokes the public key to which it is linked. This disallows any user from using the public key for encryption which is useful in the case one loses their private key.

Problem 5.2

a) To generate a public/private key pair with openssl the following instructions should be used:

```
$ openssl genrsa -des3 -out private.pem 2048
```

This will prompt the user to enter a passphrase. This specific command uses a 2048 bit key size.

To extract the public key into a serparate file the follwoing command should be used:

```
$ openssl rsa -in private.pem -outform PEM -pubout -out public.pem
```

The public key should now be in "public.pem".

b) To generate a CSR for the key pair previously generated type the following into terminal:

```
$ openssl req -new -key private.pem -out req.csr
```

To show the content of the request type the following in terminal:

```
$ openssl req -in req.csr -noout -text
```

Below is the content of the genrated CSR:

```
Certificate Request:
    Data:
        Version: 1 (0x0)
        Subject: C = DE, ST = Bremen, L = Bremen, O = Jacobs University Bremen
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (2048 bit)
                Modulus:
                    00:99:d8:d8:cc:ff:3d:06:df:8d:67:4c:3e:89:af:
                    75:06:af:e5:ae:9d:1c:83:6f:32:40:29:17:ac:5d:
                    81:f2:b4:03:66:30:30:10:4f:7e:d8:88:36:bd:ad:
                    46:f6:fa:48:1e:cb:66:5e:e6:99:c3:f2:a7:61:61:
                    be:8d:70:dd:6d:e9:77:1f:f9:0d:65:cb:fc:ca:bd:
                    50:1b:f6:ab:17:2b:1a:f7:f5:17:8a:87:02:e4:3a:
                    ed:ea:bb:c8:38:64:81:8a:b0:c6:8e:bc:90:d3:ca:
                    5b:cc:59:f8:c5:49:ea:6a:b4:5a:2a:58:3f:56:75:
                    93:e2:a4:07:18:9f:77:66:66:90:e8:7e:21:72:80:
                    2e:e8:9f:17:e4:d9:d2:7c:95:6c:d7:e2:be:d0:61:
                    f0:cf:da:b7:93:96:9f:c2:72:ab:eb:2c:1d:74:df:
                    35:5d:ee:18:26:07:2f:51:0b:9b:1e:6e:bd:8e:c1:
                    91:6f:06:a0:51:c2:49:5c:4e:7f:86:8b:1b:4f:d0:
                    8c:18:67:e2:94:31:36:03:e6:32:7b:8e:6a:cc:93:
                    3d:00:be:d8:c9:d9:98:f5:46:bf:1c:b9:09:99:34:
                    61:3e:0d:1d:e9:2f:78:28:7f:ba:26:9d:9e:64:9d:
                    8a:ee:88:e1:9e:c4:e6:0b:24:a4:90:b6:df:84:38:
                    ef:15
                Exponent: 65537 (0x10001)
        Attributes:
            a0:00
    Signature Algorithm: sha256WithRSAEncryption
         51:85:c0:20:b1:94:26:f4:9a:77:a3:fb:43:bf:88:c6:1a:e7:
         47:0c:8a:91:3c:d2:ae:63:8b:cb:6b:07:50:37:8d:e0:a4:db:
         61:61:93:d2:b8:a1:6f:5e:56:ae:3e:42:ac:cd:34:b6:2b:12:
         45:c4:2e:a8:c7:99:ad:f1:0a:eb:36:f2:42:77:6b:21:2c:c6:
         32:ff:50:79:76:23:2a:ed:92:65:87:f4:09:3c:62:dd:d2:0a:
         bd:66:6d:d1:fb:05:00:8a:6e:3a:9e:e3:4c:db:f5:9f:92:56:
         cb:f3:c3:cf:e8:87:2a:ec:0e:64:d8:fa:0d:c5:4c:9c:b5:c9:
         37:48:34:1b:e9:1c:ac:bc:21:8e:69:97:10:41:cc:2d:52:24:
         45:7b:af:2d:49:3a:87:18:35:49:97:c6:bd:78:b6:e6:c7:53:
         89:e8:fd:0f:88:00:a1:42:ad:30:1f:12:83:ef:1d:61:07:cc:
         ab:e0:98:1a:8d:07:bb:32:3f:b0:b1:12:ab:6b:13:8e:6b:a5:
         ed:4e:45:03:7e:07:0a:78:d1:0f:5a:aa:aa:bc:c2:22:cd:13:
         5f:27:4a:26:1d:6d:1f:51:ac:bf:ff:6f:17:fb:f0:9d:f1:7f:
         42:11:af:c3:02:0f:72:95:a2:1a:de:7c:43:b1:a0:b0:84:a3:
         3f:ad:f0:47
```

c) To setup up a Certificate Authority (CA) follow the guide below:

- Create this directory hierachy in a parent directory:

```
total 20
drwxr-xr-x 2 user user 4096 May  9 15:18 certs
drwxr-xr-x 2 user user 4096 May  9 15:18 crl
drwxr-xr-x 2 user user 4096 May  9 15:17 newcerts
drwxr-xr-x 2 user user 4096 May  9 15:18 private
drwxr-xr-x 2 user user 4096 May  9 15:18 requests
```

- Create an file called *index.txt* and a file to store the serial number:

```
$ touch index.txt
$ echo '1234' > serial
```

  where 1234 is our chosen serial number.

- Use our previously generated private key as the input for the root certificate:

```
$ openssl req -new -x509 -key private.pem -out cacert.pem -days 365
```

  where public.pem is our previously generated key cacert.pem is the certificate output and 365 is the perioud of validity of the certificate in days. Follow the prompts from terminal until completed.

d) In order to sign a CSR follow the instructions below:

- Using the 'req.csr' file generated in question b), we can sign the request by typing:

```
$ openssl ca -in req.csr -out cert.crt
```

  After completing the prompts the signed request is now in 'cert.crt'

e) Find uploaded certificate in submission directory under 'certificate/cert.crt'. This certificate was signed by Brian Hanna Nasralla.

Problem 5.3

a) The certificate for `https://cnds.jacobs-university.de` is valid from September 19, 2018 until December 21, 2020 (591 days from today).
The certificates in the chain are valid

b) OCSP Stapling works by appending a time-stamped OCSP response signed by the CA to the initial TLS handshake. This removes the need for clients to contact the certificate authority. The CA queries the the OCSP server at regular intervals and obtains a signed time-stamped OCSP response. When clients attempt to connect, the response is 'stapled' to the TLS handshake.
`https://beadg.de` supports OCSP stapling while `https://cnds.jacobs-university.de` does not.