REPUBLIQUE TUNISIENNE

MINISTERE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

FACULTÉ DES SCIENCES DE TUNIS

TP 3 Open SSL

Réalisé par

Houcem Hbiri - Oussema Hamouda - Brahem Salim

Étudiants en 4ème année

Spécialité : Ingénierie en génie logiciel

I. Installation et configuration de OpenSSL

1- Installation de OpenSSL

2- Vérification de la version en utilisant

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl version
OpenSSL 3.0.13 30 Jan 2024 (Library: OpenSSL 3.0.13 30 Jan 2024)
```

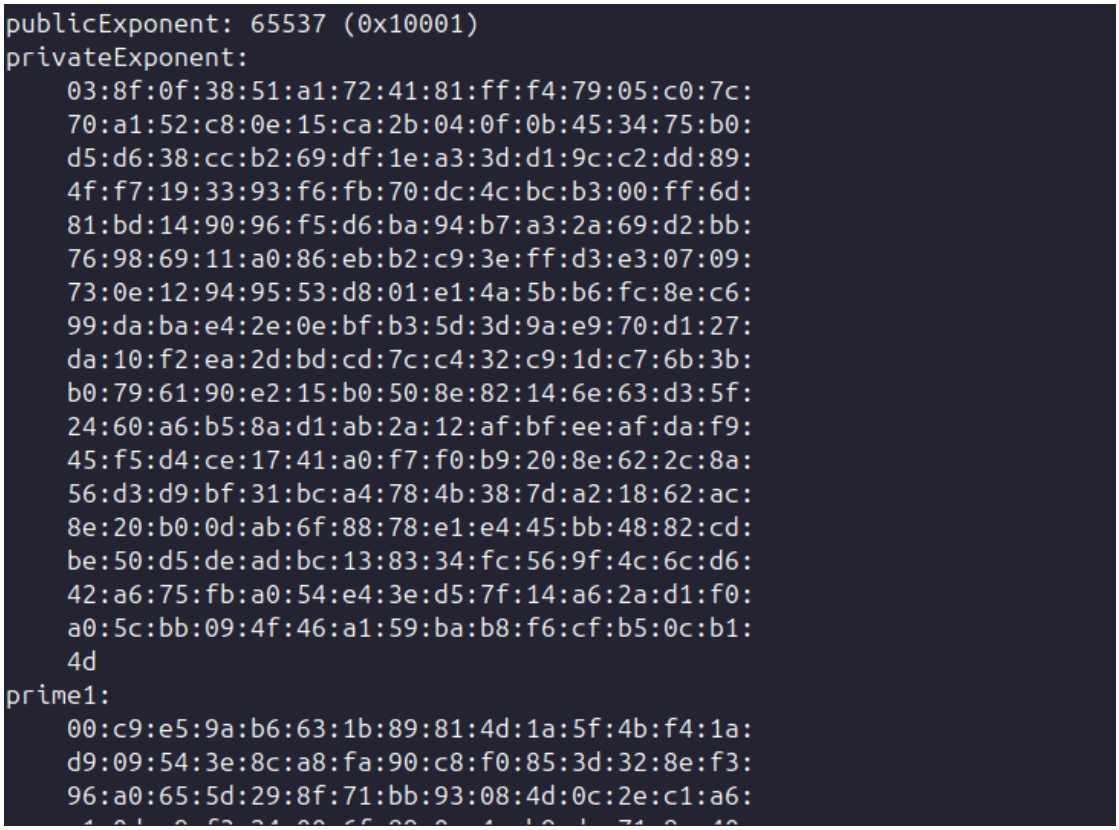
II. Génération de clés RSA

1-Générer une paire de clé dans un fichier nommé key.pem.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl genrsa -out key.pem 2048
oussema@oussema-hp:~/ttp_OpenSSL$
```

2-Visualiser la clé générée et donner l'exposant publique

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl rsa -in key.pem -text -noout
Private-Key: (2048 bit, 2 primes)
modulus:
    00:96:ef:c4:eb:39:a7:c3:31:1f:2e:57:62:fb:a7:
    11:4a:4e:03:f4:b6:60:36:79:0f:0b:5c:86:a4:e7:
    fc:46:ea:71:b7:e2:ef:ba:80:de:12:72:a7:fc:12:
    c0:02:b8:00:c6:55:54:59:ee:a5:05:c1:bf:96:5f:
    ec:50:1d:8a:12:64:b7:03:42:37:9e:3c:65:0e:65:
    5a:a2:40:78:f1:cf:4d:1d:66:2a:ca:36:12:72:b6:
    90:3a:e6:22:fb:63:2e:a6:5e:65:d7:d3:7d:fe:b7:
    d6:06:44:76:45:d2:2e:43:b1:cb:65:1f:f0:b6:aa:
    80:58:ee:64:ff:12:c2:48:e9:6b:b7:9e:ff:77:04:
    ea:77:c1:4c:cb:67:53:1e:cf:6b:98:ad:5a:c4:33:
    0a:e2:a6:bc:2d:d3:61:26:9b:a7:58:e4:b8:5f:86:
    33:65:07:9a:38:f4:17:dc:0e:3c:65:69:72:77:e4:
    00:03:fa:d8:a6:1b:65:0b:fc:26:bd:46:1e:a4:07:
    fb:e3:dd:27:cb:21:3c:86:20:ea:5a:30:e8:db:f6:
    6b:84:98:fc:dd:0c:53:f5:b2:73:9d:23:59:92:e4:
    d9:27:89:50:9c:37:cb:82:f9:52:76:59:4f:a6:5f:
    93:7b:41:56:71:b0:0e:c4:69:36:13:50:a9:82:86:
    1d:71
publicExponent: 65537 (0x10001)
privateExponent:
    03:8f:0f:38:51:a1:72:41:81:ff:f4:79:05:c0:7c:
    70:a1:52:c8:0e:15:ca:2b:04:0f:0b:45:34:75:b0:
    d5:d6:38:cc:b2:69:df:1e:a3:3d:d1:9c:c2:dd:89:
```



3-Chiffrer dans le fichier key_s.pem la clé key.pem avec le chiffrement des3.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl rsa -des3 -in key.pem -out key_s.pem
writing RSA key
Enter pass phrase:
Verifying - Enter pass phrase:
oussema@oussema-hp:~/ttp_OpenSSL$ cat key_s.pem
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFHDBOBgkqhkiG9w0BBQ0wQTApBgkqhkiG9w0BBQwwHAQIzWi6J0zO4LgCAggA
MAwGCCqGSIb3DQIJBQAwFAYIKoZIhvcNAwcECLIU4u54llthBIIEyA2eQ6QNv6XU
jfHkWHYEHIh2Mq5wYOfT5+otAmkAzo+CrmWbs9lZeQKBhaj+/ib648GAu2tA9S7O
jQ1/qBxm4xkTx78vFMaRNR9UweR6Eernaet7hdEbMlVKwc4dYUjbSjGB3UaeAL85
BNYzz4gyv2dPxUNAraDRvR+xha1KcqphjbgvGmzvJqxbX1P84E431yMIjAwYVhba
N6sZuYLHEGuoSJNaPy2ZdrjkN19Df94Y6ikIvAMR+gaJ0ZkE4kRrs6EIYQ3Pjw0e
EO8+dfXF5H1lzGcvwROup67Mzo5rKuuBVQQdHyFYJ7l3tj6TZQul5QIiaqEdUHVh
KmPmDsZUXGhu1ZqfXWacakaYaQkXdHvvdF0Pik8dJ3JY3YgPeL99GzR/5trkZKWg
Oxwik0T5mnk6tVUbCMOnU5tDSC1KKqMY8fsNtt4CXnJ+fAwg3Eru+ORuWTo8jQ1G
ZysPqlMQW94LSoFXr1jkST5Pjt0lEGycD3Y1m/oZpA/vIst5OZAv5pOoAQu+LoLI
qqttYk7hCuSHvG4bpE2H3wgGlh6s7glrboUJN6rW/z69uw/lQZqAEMGqWjlu2N65
Vua4dSR/v3ftlv3/CePWrinF4oC4U6pZUHtyh+FWuiW6fLcVsrUY/xYc67fp3Qm9
Oqw3nxw543xrMHLwmtmSVEO6l8j2jgX78Oi0Bmq5UPVwVL/lTXn9hQAxDHErBqQz
u9F/pxKcg4O5W8h2dyZb2XXHSMHeIsgh/qTPeFbc9LBESzeU2KXdCQNsac7bAOWh
iPc6K0YPzCBhZN42kF++A87qldBfmvT8Nu7TuOrJcdBd8hJattRnhb9CTsSllLuo
khGAvGDtrknMZP/LABRZZLxYIhxe4NUyuNeJ2EZ222AYixF0Q3vjwA/HS31ncbpt
wd7V4aWsX+4f+y+ZUUJPK2/7kM89lLLq2fgskmj4NG8VGJd+r9c4p6GzlkuS0CAp
0qq5SiOfXwPxWXta6BZgtli2VcuIC8UFMW8Mq4Nbi2xYC412fNL+E62JYhfz0KNv
MxYpfuMuhAf2EbMCkFFIYdH9k8+crlvCTR+Z7mxG8GOsCcWS0DgiYD7Q+/3PwbMF
93UBi32zQKfyJkYBk0Awz+EWw4pE4Auu7/5zJ/tGEm2v2xuN42/Z9ToNexv5RG67
Nla/Nc9lWK5Ujy9OumziW5ueE3a0Q9qEE4gzItr+oDZ+//ODAdlJCE7h+4usHAfR
U9wBhxPbddCR6ZmilVRJj84/TZeDMIgSv5q3gPRGV89ARz3dk5HdM/z+XyP1BjEG
whNr7lhk9X4o9r+RECIgYgbQrUgERXPOxx/j/1xEkrCdgs3wSUS2e35zz3KIt3EO
KnrbzeRz31L1HR13eOrXKw69JzHd4JLEm2SBhL63LdLSCEf529JcmTnyf3lC1y2X
S4nUM2qiUQys27dwKqcRRDIpugILEy/F4WPQKf0L5452ovLyvHUn/gsC/keknNal
sgaAB8ggZWmy9/W8RJ979rvW0v7WviXmqM6NYXvirRcrhsFbjpQY2SyOP0nleqKc
5zH2hUR7Jif+01oAjOZEVsxgcvdgl/MOVL570569bPOpKACMPxELDwS3q/2pN9dr
Txa0PO6/4Lz3YvzUunTJ7Q==
-----END ENCRYPTED PRIVATE KEY-----
```

4. Exporter la clé publique dans le fichier key_pub.pem.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl rsa -pubout -in key.pem -out key_pub.pem
writing RSA key
oussema@oussema-hp:~/ttp_OpenSSL$ cat key_pub.pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAlu/E6zmnwzEfLldi+6cR
Sk4D9LZgNnkPC1yGpOf8Rupxt+LvuoDeEnKn/BLAArgAxlVUWe6lBcG/ll/sUB2K
EmS3A0I3njxlDmVaokB48c9NHWYqyjYScraQOuYi+2Mupl5l19N9/rfWBkR2RdIu
Q7HLZR/wtqqAWO5k/xLCSOlrt57/dwTqd8FMy2dTHs9rmK1axDMK4qa8LdNhJpun
WOS4X4YzZQeaOPQX3A48ZWlyd+QAA/rYphtlC/wmvUYepAf7490nyyE8hiDqWjDo
2/ZrhJj83QxT9bJznSNZkuTZJ4lQnDfLgvlSdllPpl+Te0FWcbAOxGk2E1CpgoYd
cQIDAQAB
-----END PUBLIC KEY-----
oussema@oussema-hp:~/ttp_OpenSSL$
```

5. Visualiser la clé publique.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl rsa -pubin -in key_pub.pem -text -noout
Public-Key: (2048 bit)
Modulus:
    00:96:ef:c4:eb:39:a7:c3:31:1f:2e:57:62:fb:a7:
    11:4a:4e:03:f4:b6:60:36:79:0f:0b:5c:86:a4:e7:
    fc:46:ea:71:b7:e2:ef:ba:80:de:12:72:a7:fc:12:
    c0:02:b8:00:c6:55:54:59:ee:a5:05:c1:bf:96:5f:
    ec:50:1d:8a:12:64:b7:03:42:37:9e:3c:65:0e:65:
    5a:a2:40:78:f1:cf:4d:1d:66:2a:ca:36:12:72:b6:
    90:3a:e6:22:fb:63:2e:a6:5e:65:d7:d3:7d:fe:b7:
    d6:06:44:76:45:d2:2e:43:b1:cb:65:1f:f0:b6:aa:
    80:58:ee:64:ff:12:c2:48:e9:6b:b7:9e:ff:77:04:
    ea:77:c1:4c:cb:67:53:1e:cf:6b:98:ad:5a:c4:33:
    0a:e2:a6:bc:2d:d3:61:26:9b:a7:58:e4:b8:5f:86:
    33:65:07:9a:38:f4:17:dc:0e:3c:65:69:72:77:e4:
    00:03:fa:d8:a6:1b:65:0b:fc:26:bd:46:1e:a4:07:
    fb:e3:dd:27:cb:21:3c:86:20:ea:5a:30:e8:db:f6:
    6b:84:98:fc:dd:0c:53:f5:b2:73:9d:23:59:92:e4:
    d9:27:89:50:9c:37:cb:82:f9:52:76:59:4f:a6:5f:
    93:7b:41:56:71:b0:0e:c4:69:36:13:50:a9:82:86:
    1d:71
Exponent: 65537 (0x10001)
oussema@oussema-hp:~/ttp_OpenSSL$
```

6. Chiffrer le ficher /etc/hostname dans le fichier hostname_encrypted en utilisant la Clé key.pem.

→ chiffrement avec une clé commune

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl pkeyutl -encrypt -in /etc/hostname -inkey key.pem -out hostname_encrypted
oussema@oussema-hp:~/ttp_OpenSSL$ cat hostname_encrypted
d5◆◆R◆◆◆◆◆(J◆6◆◆◆◆I◆◆◆◆◆*◆◆◆h◆◆2◆I
{◆◆6◆◆%◆◆◆G◆◆n◆◆◆◆◆WB%◆◆:x◆at◆◆◆◆◆◆◆`;8[illegible]◆◆◆◆{G◆t◆◆l◆◆{S◆t
◆E◆◆I◆:?R¦x>◆◆z◆◆v(◆◆   a                                    I◆◆骞[◆:◆◆z<◆◆◆3◆?◆◆◆gmr5d◆◆c)
                        i◆;◆/◆V;◆◆◆◆ZS◆◆
|◆◆W◆f◆◆f7◆◆◆??5◆k◆%1◆(◆v◆◆◆NA◆q^w!◆◆◆◆oussema@oussema-hp:~/ttp_OpenSSL$
```

7. Chiffrer le ficher /etc/hostname dans le fichier hostname_encrypted_s en utilisant la clé key_s.pem.

→ chiffrement avec une clé privée

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl pkeyutl -encrypt -in /etc/hostname -inkey key_s.pem -out hostname_encrypted_s
Enter pass phrase for key_s.pem:
oussema@oussema-hp:~/ttp_OpenSSL$
```

8. Chiffrer le ficher /etc/hostname dans le fichier hostname_encrypted_pub en utilisant la clé key_pub.pem.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl pkeyutl -encrypt -in /etc/hostname -inkey key_pub.pem -pubin -out hostname_encrypted_pub
oussema@oussema-hp:~/ttp_OpenSSL$
```

9. Quelle est la différence entre l'exécution de la commande 6 et 7?

6. Chiffrement avec une clé commun.
7. Chiffrement avec une clé privée.

10. Quelle est la différence entre la commande 6 et 8 ?

6. Chiffrement avec une clé commun.
8. Chiffrement avec une clé publique.

11. Déchiffrer le fichier hostname_encrypted.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl pkeyutl -decrypt -in hostname_encrypted -inkey key.pem -out  hostname_decrypted
oussema@oussema-hp:~/ttp_OpenSSL$ cat hostname_decrypted
oussema-hp
oussema@oussema-hp:~/ttp_OpenSSL$ cat hostname_encrypted
d5◆◆R◆◆◆◆◆(J◆6◆◆◆◆◆I◆◆◆◆◆*◆◆◆h◆◆2◆I
[◆◆6◆◆%◆◆◆G◆◆n◆◆◆◆◆wB%◆◆:x◆a[◆◆◆◆◆◆◆`;8▒◆◆◆◆{G◆t◆◆l◆◆{S◆t
>E◆◆I◆:?R¦x>◆◆Z◆◆V(◆◆   a                                  I◆◆騫[◆:◆◆z<◆◆◆3◆?◆◆◆gmr5d◆◆c)
                      i◆;◆/◆V;◆◆◆◆ZS◆◆
◆◆◆w◆f◆◆f7◆◆◆??5◆k◆%1◆(◆v◆◆◆NA◆q^w!◆◆◆◆oussema@oussema-hp:~/ttp_OpenSSL$
```

12. Déchiffrer le fichier hostname_encrypted_s.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl pkeyutl -decrypt -in hostname_encrypted_s -inkey key_s.pem -out  hostname_decrypted_s
Enter pass phrase for key_s.pem:
oussema@oussema-hp:~/ttp_OpenSSL$ cat hostname_decrypted_s
oussema-hp
oussema@oussema-hp:~/ttp_OpenSSL$
```

13. Déchiffrer le fichier hostname_encrypted_pub. Quelle clé nous avons utilisé avec ce fichier ?

On a utilisé private key pour déchiffrer encrypted_pub.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl pkeyutl -decrypt -in hostname_encrypted_pub -inkey key_s.pem -out  hostname_decrypted_pub
Enter pass phrase for key_s.pem:
oussema@oussema-hp:~/ttp_OpenSSL$ cat hostname_decrypted_pub
oussema-hp
oussema@oussema-hp:~/ttp_OpenSSL$
```

14. Générer le fichier passwd_h qui contient l'empreinte du fichier /etc/passwd.

```
oussema@oussema-hp:~/ttp_OpenSSL$ sudo openssl dgst -md5 -out passwd-h /etc/passwd
oussema@oussema-hp:~/ttp_OpenSSL$ cat passwd-h
MD5(/etc/passwd)= 8afabd2ca8c8aeabb7e5087ee751f579
oussema@oussema-hp:~/ttp_OpenSSL$
```

15. Générer la signature passwd_sign qui contient la signature du fichier /etc/passwd.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl pkeyutl -sign -in passwd-h -inkey key.pem -out passwd_sign
oussema@oussema-hp:~/ttp_OpenSSL$
```

16. Vérifier la signature.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl pkeyutl -verify -in passwd-h -sigfile passwd_sign -pubin -inkey key_pub.pem
Signature Verified Successfully
```

III. Certificats

1. Générer le fichier «req.pem» qui contiendra votre requête de certificat.

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl req -new -key key.pem -out req.pem
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:tn
State or Province Name (full name) [Some-State]:tunis
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
oussema@oussema-hp:~/ttp_OpenSSL$
```

2. Visualiser la requête de certificat. Quel est le marqueur de début de ce fichier?

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl req -in req.pem -text -noout
Certificate Request:
    Data:
        Version: 1 (0x0)
        Subject: C = tn, ST = tunis, O = Internet Widgits Pty Ltd
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:96:ef:c4:eb:39:a7:c3:31:1f:2e:57:62:fb:a7:
                    11:4a:4e:03:f4:b6:60:36:79:0f:0b:5c:86:a4:e7:
                    fc:46:ea:71:b7:e2:ef:ba:80:de:12:72:a7:fc:12:
                    c0:02:b8:00:c6:55:54:59:ee:a5:05:c1:bf:96:5f:
                    ec:50:1d:8a:12:64:b7:03:42:37:9e:3c:65:0e:65:
                    5a:a2:40:78:f1:cf:4d:1d:66:2a:ca:36:12:72:b6:
                    90:3a:e6:22:fb:63:2e:a6:5e:65:d7:d3:7d:fe:b7:
                    d6:06:44:76:45:d2:2e:43:b1:cb:65:1f:f0:b6:aa:
                    80:58:ee:64:ff:12:c2:48:e9:6b:b7:9e:ff:77:04:
                    ea:77:c1:4c:cb:67:53:1e:cf:6b:98:ad:5a:c4:33:
                    0a:e2:a6:bc:2d:d3:61:26:9b:a7:58:e4:b8:5f:86:
                    33:65:07:9a:38:f4:17:dc:0e:3c:65:69:72:77:e4:
                    00:03:fa:d8:a6:1b:65:0b:fc:26:bd:46:1e:a4:07:
                    fb:e3:dd:27:cb:21:3c:86:20:ea:5a:30:e8:db:f6:
                    6b:84:98:fc:dd:0c:53:f5:b2:73:9d:23:59:92:e4:
                    d9:27:89:50:9c:37:cb:82:f9:52:76:59:4f:a6:5f:
                    93:7b:41:56:71:b0:0e:c4:69:36:13:50:a9:82:86:
                    1d:71
                Exponent: 65537 (0x10001)
        Attributes:
            (none)
            Requested Extensions:
    Signature Algorithm: sha256WithRSAEncryption
    Signature Value:
        58:86:a6:b2:6d:02:d1:57:7f:b6:6a:55:ab:a1:67:18:6a:fd:
        fe:a2:d2:c1:c7:f5:de:08:33:b1:94:68:11:aa:7c:d2:05:67:
        13:7c:c5:d9:cc:27:bd:82:4e:05:7b:80:96:68:78:59:65:a4:
        13:f6:51:58:4c:98:2e:f2:7f:91:6f:7a:15:af:05:17:0b:43:
        7e:c2:87:33:a4:d1:3e:93:65:42:02:ce:d1:64:cc:ed:67:f1:
        f5:86:58:82:f2:00:1a:58:a9:46:3f:2e:78:f7:94:4e:09:b5:
```

3. Quelles sont les informations, si elles existent, de la clé privé qu'on peut avoir?

Aucune information sur la clé privée

4. Générer un certificat d'autorité auto-signé « CA_cert.pem »

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl req -new -x509 -days 365 -key key.pem -out CA_cert.perm
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:tn
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
oussema@oussema-hp:~/ttp_OpenSSL$
```

5. Générer le un certificat «cert.crt» de votre clé

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl x509 -req -days 365 -CA CA_cert.perm -CAkey key.pem -set_serial 5 -in req.pem -out cert.crt
Certificate request self-signature ok
subject=C = AU, ST = Some-State, O = Internet Widgits Pty Ltd
oussema@oussema-hp:~/ttp_OpenSSL$
```

6. Vérifier le certificat généré.

```
        64:1d:b8:61
oussema@oussema-hp:~/ttp_OpenSSL$ openssl verify -CAfile CA_cert.perm cert.crt
cert.crt: OK
oussema@oussema-hp:~/ttp_OpenSSL$
```

7. Chiffrer le mail «~/.bashrc» et enregistrer le dans le fichier «mailc»

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl smime -encrypt -in ~/.bashrc -text -from "em@em.com" -to "am@am.com" -subject "mon Courrier" -des3 -out mailc cert.crt
oussema@oussema-hp:~/ttp_OpenSSL$
```

8. Déchiffrer le mail «mailc» et enregistrer le dans le fichier «maile»

```
oussema@oussema-hp:~/ttp_OpenSSL$ openssl smime -decrypt -in mailc -recip cert.crt -inkey key.pem  -out maile
oussema@oussema-hp:~/ttp_OpenSSL$ cat mail
cat: mail: No such file or directory
oussema@oussema-hp:~/ttp_OpenSSL$ cat maile
Content-Type: text/plain

# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
      *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
shopt -s checkwinsize
```