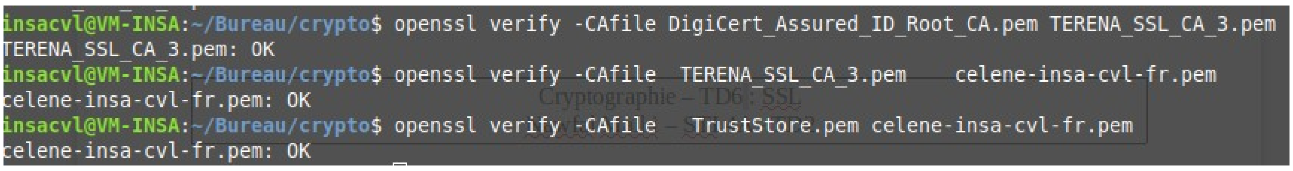
Cryptographie – TD6 : SSL
Nawfal Malki – STI 4A - TD2

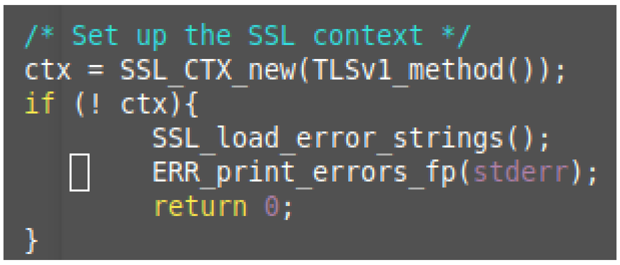
Partie 1 (client withssl.c, serveur celene)

0. On vérifie la chaîne de certification.



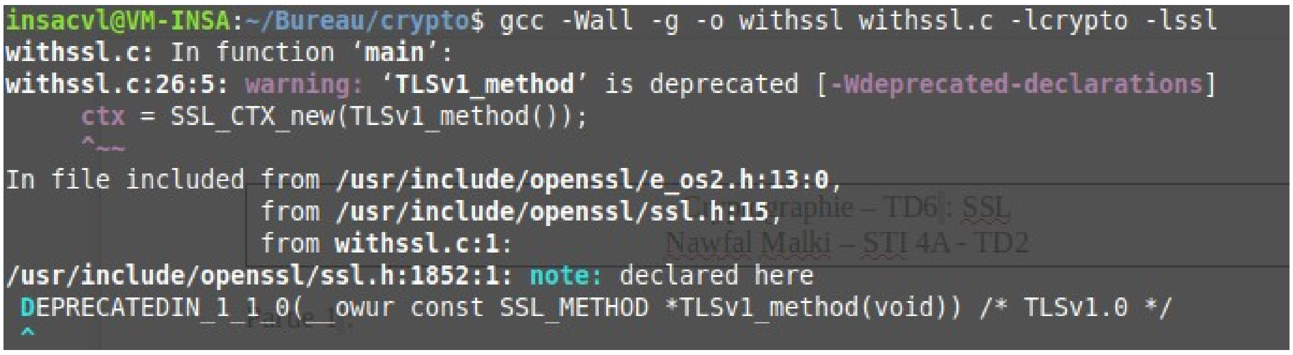
```
insacvl@VM-INSA:~/Bureau/crypto$ openssl verify -CAfile DigiCert_Assured_ID_Root_CA.pem TERENA_SSL_CA_3.pem
TERENA_SSL_CA_3.pem: OK
insacvl@VM-INSA:~/Bureau/crypto$ openssl verify -CAfile  TERENA_SSL_CA_3.pem    celene-insa-cvl-fr.pem
celene-insa-cvl-fr.pem: OK
insacvl@VM-INSA:~/Bureau/crypto$ openssl verify -CAfile   TrustStore.pem celene-insa-cvl-fr.pem
celene-insa-cvl-fr.pem: OK
```

1. Dans le code source withssl.c, on change le type de connexion pour établir une connexion TLSv1. On utilise pour cela la méthode TLSv1_method() pour initialiser le contexte SSL.



```
/* Set up the SSL context */
ctx = SSL_CTX_new(TLSv1_method());
if (! ctx){
        SSL_load_error_strings();
        ERR_print_errors_fp(stderr);
        return 0;
}
```

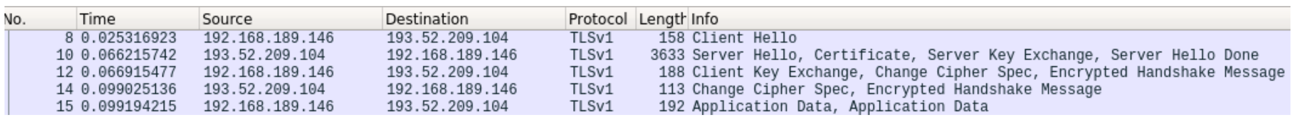
On compile. On obtient un warning qui nous indique que la méthode TLSV1_method est dépréciée.



```
insacvl@VM-INSA:~/Bureau/crypto$ gcc -Wall -g -o withssl withssl.c -lcrypto -lssl
withssl.c: In function 'main':
withssl.c:26:5: warning: 'TLSv1_method' is deprecated [-Wdeprecated-declarations]
     ctx = SSL_CTX_new(TLSv1_method());
     ^~~
In file included from /usr/include/openssl/e_os2.h:13:0,
                 from /usr/include/openssl/ssl.h:15,
                 from withssl.c:1:
/usr/include/openssl/ssl.h:1852:1: note: declared here
 DEPRECATEDIN_1_1_0(__owur const SSL_METHOD *TLSv1_method(void)) /* TLSv1.0 */
 ^
```

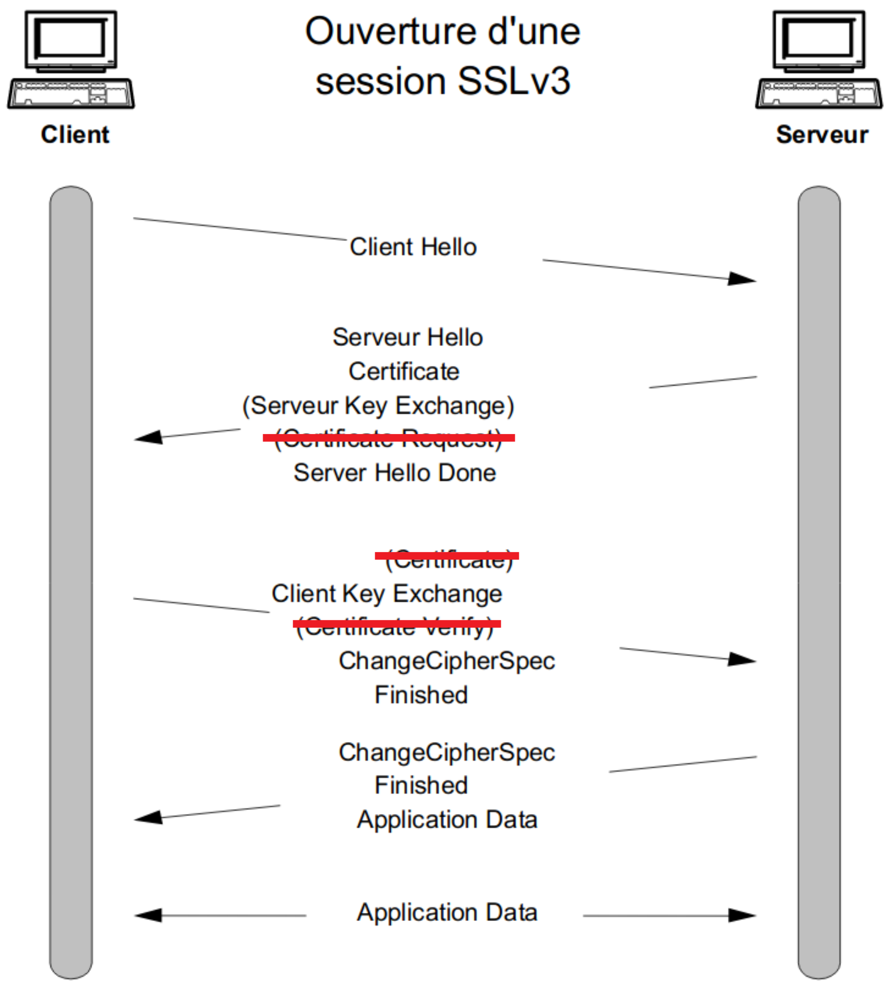
On lance withssl. Le serveur de celene.insa-cvl.fr établie la connexion SSL avec notre client et répond à la requête.

2. On lance wireshark et on lance une capture avec le filtre ssl.



```
No.  Time         Source           Destination      Protocol Length Info
   8 0.025316923  192.168.189.146  193.52.209.104   TLSv1      158 Client Hello
  10 0.066215742  193.52.209.104   192.168.189.146  TLSv1     3633 Server Hello, Certificate, Server Key Exchange, Server Hello Done
  12 0.066915477  192.168.189.146  193.52.209.104   TLSv1      188 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
  14 0.099025136  193.52.209.104   192.168.189.146  TLSv1      113 Change Cipher Spec, Encrypted Handshake Message
  15 0.099194215  192.168.189.146  193.52.209.104   TLSv1      192 Application Data, Application Data
```

Il manque l’étape de vérification du certificat du client Certificate Verify ainsi que l’étape d’envoi du certificat client. Ici, le serveur ne le réclame pas, et nous n’en n’avons pas.

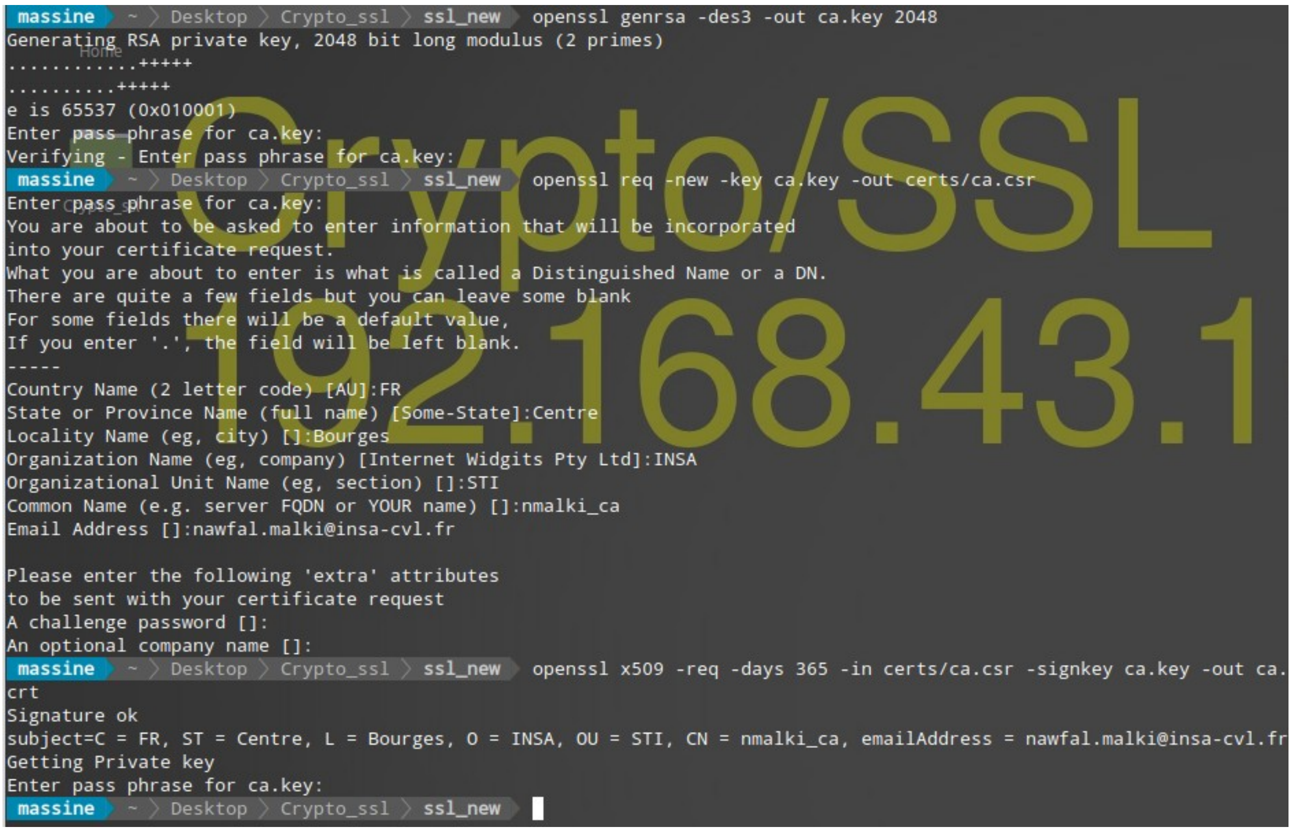


3. Le client SSLHandshake vérifie si le Common Name qui caractérise le certificat correspond bien au nom de domaine avec lequel nous souhaitons établir une connexion SSL. Cela permet à notre client de s’assurer qu’il communique bien avec le bon serveur, celui dont le CN est indiqué dans le certificat qu’il a délivré, et que ce n’est pas un autre serveur qui essaye de nous leurrer avec un certificat qui ne lui appartient pas.

Partie 2 (serveur/client)

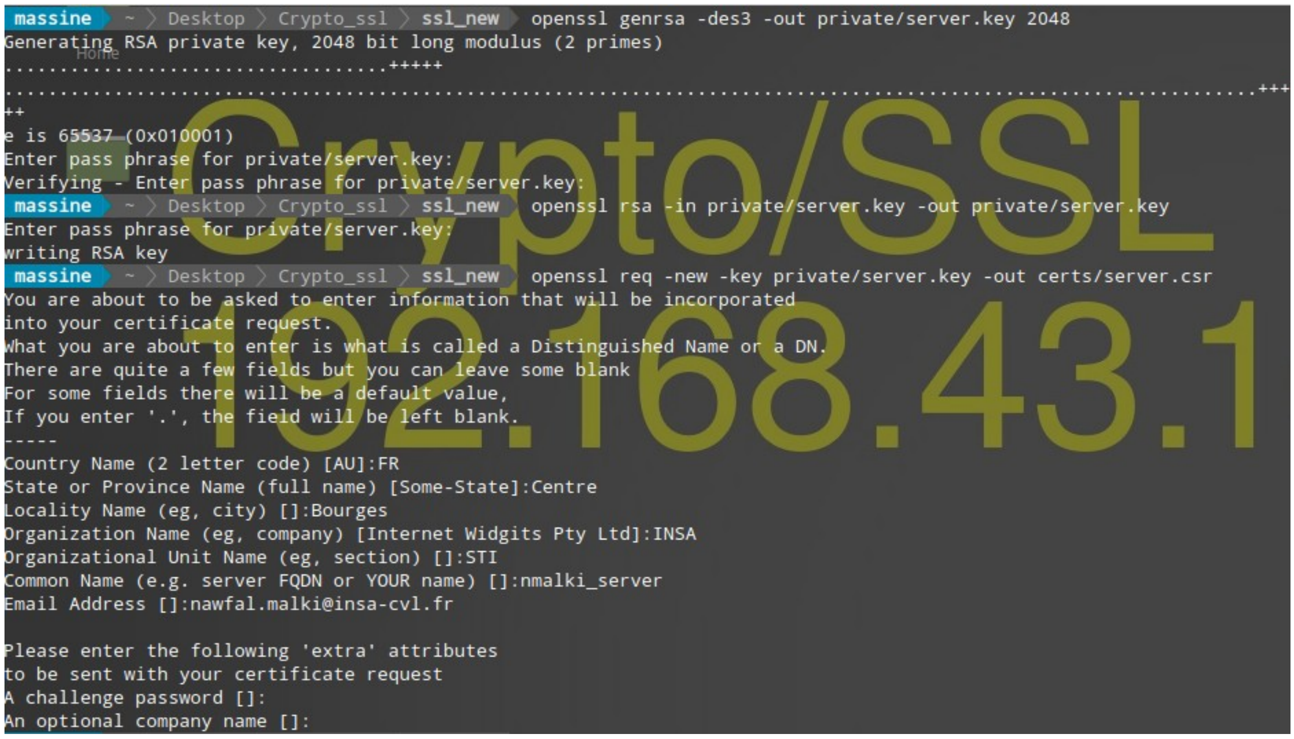
0. Préparation de l'Autorité de Certification et des certificats serveur/client

Création d'un centre d'authentification

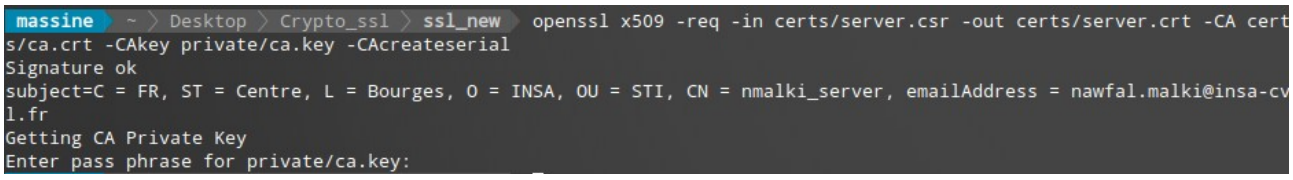


```
massine  ~ 〉 Desktop 〉 Crypto_ssl 〉 ssl_new  openssl genrsa -des3 -out ca.key 2048
Generating RSA private key, 2048 bit long modulus (2 primes)
...........+++++
.........+++++
e is 65537 (0x010001)
Enter pass phrase for ca.key:
Verifying - Enter pass phrase for ca.key:
massine  ~ 〉 Desktop 〉 Crypto_ssl 〉 ssl_new  openssl req -new -key ca.key -out certs/ca.csr
Enter pass phrase for ca.key:
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]:FR
State or Province Name (full name) [Some-State]:Centre
Locality Name (eg, city) []:Bourges
Organization Name (eg, company) [Internet Widgits Pty Ltd]:INSA
Organizational Unit Name (eg, section) []:STI
Common Name (e.g. server FQDN or YOUR name) []:nmalki_ca
Email Address []:nawfal.malki@insa-cvl.fr

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
massine  ~ 〉 Desktop 〉 Crypto_ssl 〉 ssl_new  openssl x509 -req -days 365 -in certs/ca.csr -signkey ca.key -out ca.
crt
Signature ok
subject=C = FR, ST = Centre, L = Bourges, O = INSA, OU = STI, CN = nmalki_ca, emailAddress = nawfal.malki@insa-cvl.fr
Getting Private key
Enter pass phrase for ca.key:
massine  ~ 〉 Desktop 〉 Crypto_ssl 〉 ssl_new  
```

Création et signature du certificat du serveur par le CA

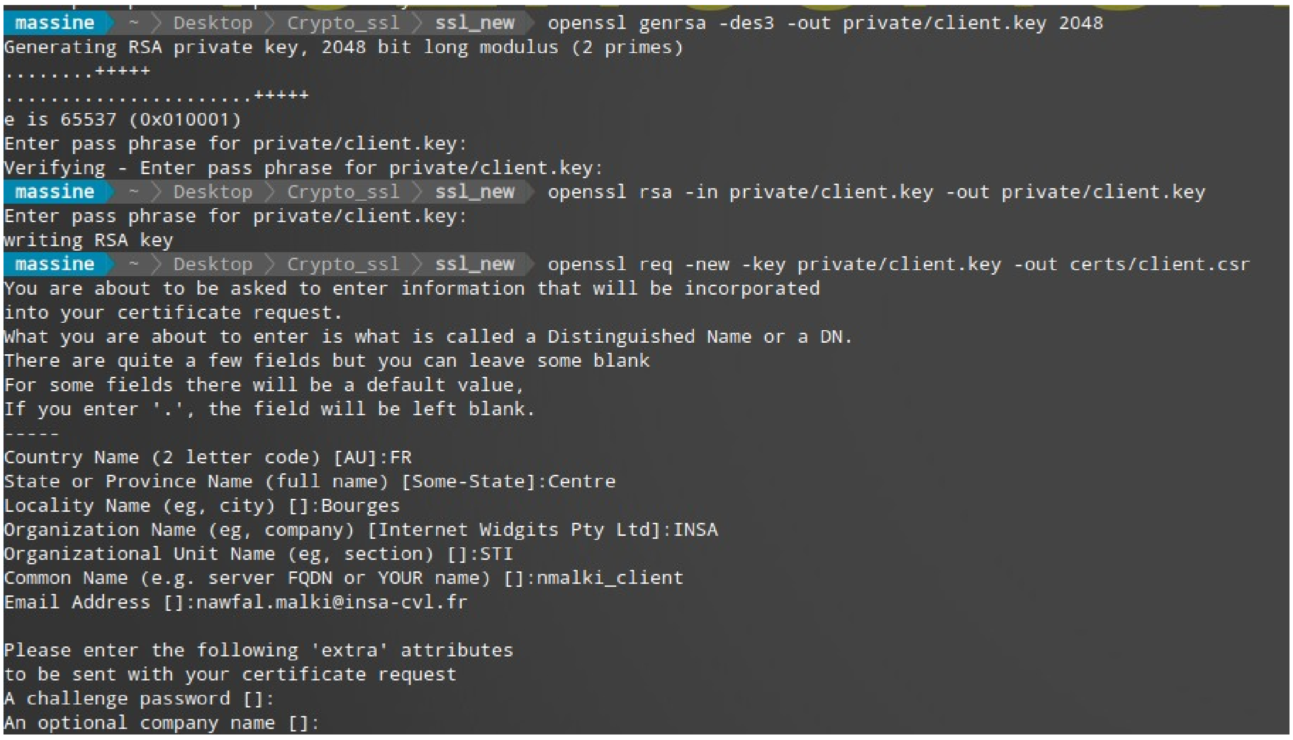


```
massine  ~ 〉 Desktop 〉 Crypto_ssl 〉 ssl_new  openssl genrsa -des3 -out private/server.key 2048
Generating RSA private key, 2048 bit long modulus (2 primes)
.................................+++++
.............................................................................................................+++
++
e is 65537 (0x010001)
Enter pass phrase for private/server.key:
Verifying - Enter pass phrase for private/server.key:
massine  ~ 〉 Desktop 〉 Crypto_ssl 〉 ssl_new  openssl rsa -in private/server.key -out private/server.key
Enter pass phrase for private/server.key:
writing RSA key
massine  ~ 〉 Desktop 〉 Crypto_ssl 〉 ssl_new  openssl req -new -key private/server.key -out certs/server.csr
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]:FR
State or Province Name (full name) [Some-State]:Centre
Locality Name (eg, city) []:Bourges
Organization Name (eg, company) [Internet Widgits Pty Ltd]:INSA
Organizational Unit Name (eg, section) []:STI
Common Name (e.g. server FQDN or YOUR name) []:nmalki_server
Email Address []:nawfal.malki@insa-cvl.fr

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

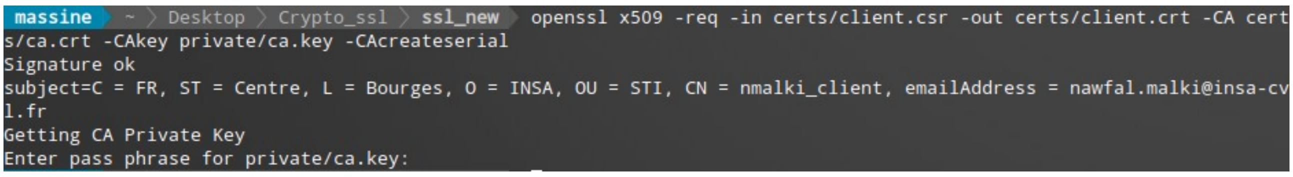


```
massine  ~  Desktop  Crypto_ssl  ssl_new  openssl x509 -req -in certs/server.csr -out certs/server.crt -CA cert
s/ca.crt -CAkey private/ca.key -CAcreateserial
Signature ok
subject=C = FR, ST = Centre, L = Bourges, O = INSA, OU = STI, CN = nmalki_server, emailAddress = nawfal.malki@insa-cv
l.fr
Getting CA Private Key
Enter pass phrase for private/ca.key:
```

On fait de même pour le client

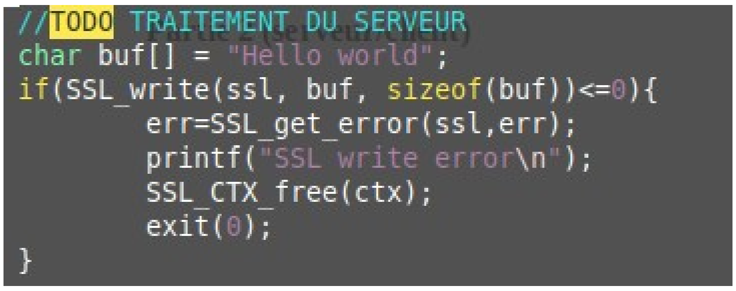


```
massine  ~  Desktop  Crypto_ssl  ssl_new  openssl genrsa -des3 -out private/client.key 2048
Generating RSA private key, 2048 bit long modulus (2 primes)
........+++++
.....................+++++
e is 65537 (0x010001)
Enter pass phrase for private/client.key:
Verifying - Enter pass phrase for private/client.key:
massine  ~  Desktop  Crypto_ssl  ssl_new  openssl rsa -in private/client.key -out private/client.key
Enter pass phrase for private/client.key:
writing RSA key
massine  ~  Desktop  Crypto_ssl  ssl_new  openssl req -new -key private/client.key -out certs/client.csr
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]:FR
State or Province Name (full name) [Some-State]:Centre
Locality Name (eg, city) []:Bourges
Organization Name (eg, company) [Internet Widgits Pty Ltd]:INSA
Organizational Unit Name (eg, section) []:STI
Common Name (e.g. server FQDN or YOUR name) []:nmalki_client
Email Address []:nawfal.malki@insa-cvl.fr

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```



```
massine  ~  Desktop  Crypto_ssl  ssl_new  openssl x509 -req -in certs/client.csr -out certs/client.crt -CA cert
s/ca.crt -CAkey private/ca.key -CAcreateserial
Signature ok
subject=C = FR, ST = Centre, L = Bourges, O = INSA, OU = STI, CN = nmalki_client, emailAddress = nawfal.malki@insa-cv
l.fr
Getting CA Private Key
Enter pass phrase for private/ca.key:
```

1. Implémentation du traitement de serveur



```c
//TODO TRAITEMENT DU SERVEUR
char buf[] = "Hello world";
if(SSL_write(ssl, buf, sizeof(buf))<=0){
        err=SSL_get_error(ssl,err);
        printf("SSL write error\n");
        SSL_CTX_free(ctx);
        exit(0);
}
```

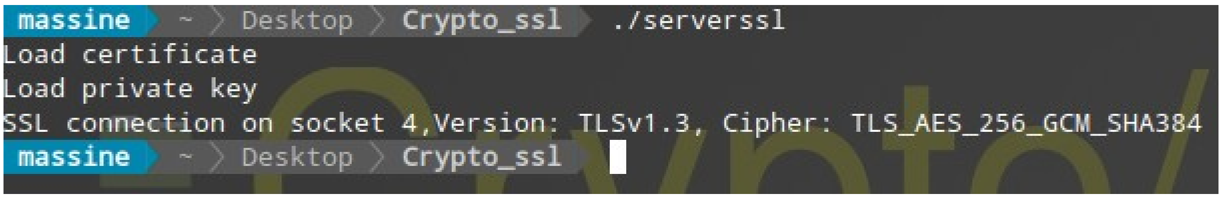
Le serveur envoie Helloworld. Libssl fournit une fonction qui permet d'écrire sur la socket par-dessus le chiffrement SSL : SSL_write().

On fait attention à bien corriger les paths des certificats.

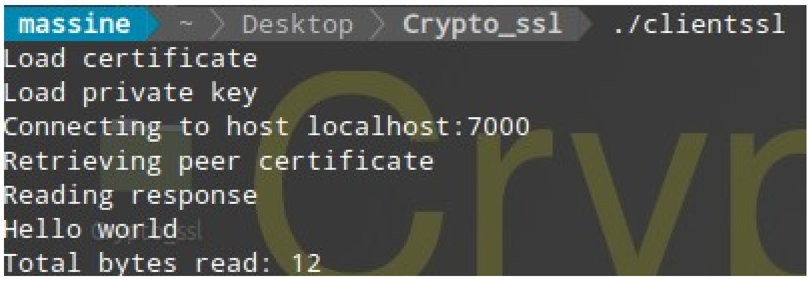
2. Implémentation du client

On se sert du modèle fourni dans sslhandshake.c pour implémenter le client :
- on met en place les librairies SSL et BIO
- on indique les path du certificat du CA, du certificat client et de la clé privée du client.

- on vérifie que la clé privée ‘’ correspond bien au certificat du client.
- on initialise la connexion pour écouter sur localhost:7000
- une fois la connexion établie, on récupère le certificat du serveur et on vérifie qu’il correspond bien au CN ‘nmalki_server’
- si la correspondance est correcte, on lit ce que nous envoie le serveur
- on ferme la connexion



```
massine  ~  Desktop  Crypto_ssl  ./serverssl
Load certificate
Load private key
SSL connection on socket 4,Version: TLSv1.3, Cipher: TLS_AES_256_GCM_SHA384
massine  ~  Desktop  Crypto_ssl
```



```
massine  ~  Desktop  Crypto_ssl  ./clientssl
Load certificate
Load private key
Connecting to host localhost:7000
Retrieving peer certificate
Reading response
Hello world
Total bytes read: 12
```