Master 2 **Protocoles des services Internet**

TP4 **TLS**

Transport Layer Security

Exercice 1.

Créez un certificat TLS à l'aide de la commande suivante :

```
openssl req -newkey rsa:2048 -nodes -keyout key.pem -x509 -days 365 -out cert.pem
```

Quels fichiers ont été créés ? Que contiennent-ils ?

(Indication : la commande **file** est votre amie.)

Exercice 2(1).

Créez un certificat TLS à l'aide de la commande suivante :

openssl req -newkey rsa:2048 -nodes -keyout key.pem -x509 -days 365 -out cert.pem

Quels fichiers ont été créés ? Que contiennent-ils ?

(Indication : la commande **file** est votre amie.)

```
lenny@DESKTOP-DMJ749K:/mnt/c/Users/lenny/OneDrive/Documents/Master 2/Premiere_periode/Protocoles
 des services Internet$ openssl req -newkey rsa:2048 -nodes -keyout key.pem -x509 -days 365 -out cert.pem
Generating a RSA private key
.....................................................................+++++
.............+++++
writing new private key to 'key.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]:FR
State or Province Name (full name) [Some-State]:Ile-de-France
Locality Name (eg, city) []:Paris
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Universite Paris Cite
Organizational Unit Name (eg, section) []:UFR Informatique
Common Name (e.g. server FQDN or YOUR name) []:Lenny
Email Address []:leonard.namolaru@etu.u-paris.fr
```

```
$ ls
 InternetTP4.docx   TP   'TP 2022'   cert.pem   key.pem   tp4.go
```

```
lenny@DESKTOP-DMJ749K:/mnt/c/Users/lenny/
$ file cert.pem
cert.pem: PEM certificate
lenny@DESKTOP-DMJ749K:/mnt/c/Users/lenny/
$ file key.pem
key.pem: ASCII text
lenny@DESKTOP-DMJ749K:/mnt/c/Users/lenny/
```

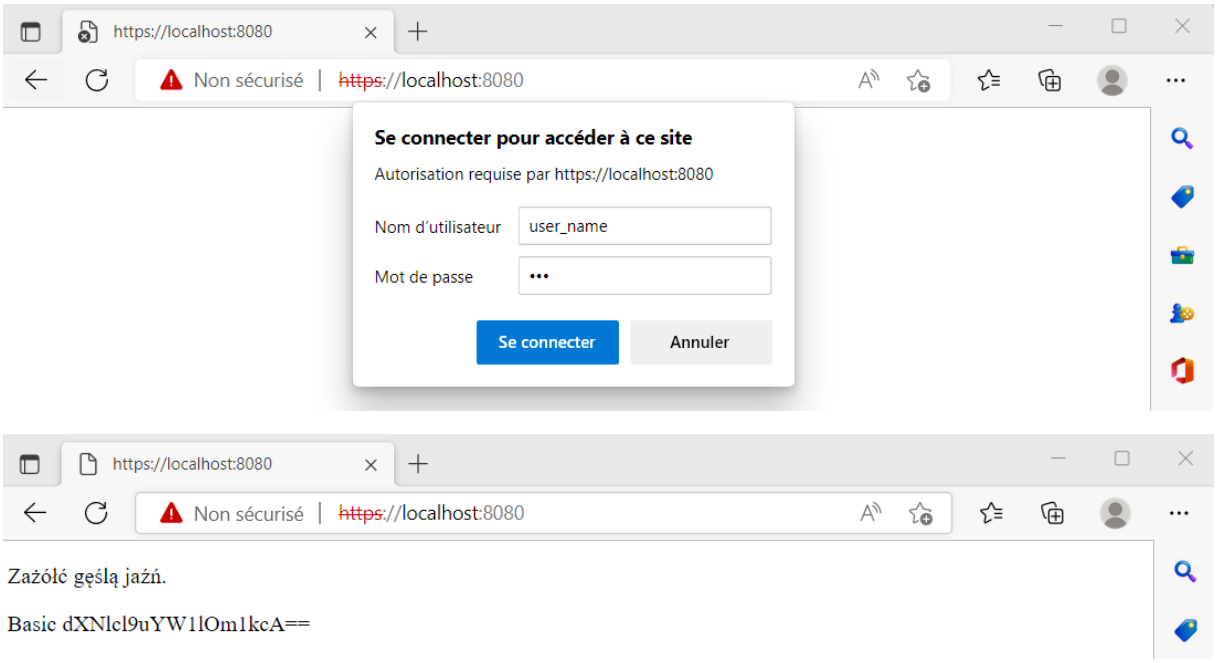
```
lenny@DESKTOP-DMJ749K:/mnt/c/Users/lenny$ gnutls-cli https://localhost:8080
Processed 129 CA certificate(s).
Resolving 'https://localhost:8080'...
Cannot resolve https://localhost:8080: Servname not supported for ai_socktype
```

```
lenny@DESKTOP-DMJ749K:/mnt/c/Users/lenny$ sudo gnutls-cli https://galene.org/
Processed 129 CA certificate(s).
Resolving 'https://galene.org/'...
Cannot resolve https://galene.org/: Servname not supported for ai_socktype
```

```
lenny@DESKTOP-DMJ749K:/mnt/c/Users/lenny$ gnutls-cli --port 8080 localhost
Processed 129 CA certificate(s).
Resolving 'localhost:8080'...
Connecting to '127.0.0.1:8080'...
- Certificate type: X.509
- Got a certificate list of 1 certificates.
- Certificate[0] info:
 - subject `EMAIL=leonard.namolaru@etu.u-paris.fr,CN=Lenny,OU=UFR Informatique,O=Universite Paris Cite,L=Paris,ST=Ile-de-France,C=FR', issuer `EMAIL=leonard.namolaru@etu.u-paris.fr,CN=Lenny,OU=UFR Informatique,O=Universite Paris Cite,L=Paris,ST=Ile-de-France,C=FR', serial 0x76c71588a9d3114f7cac12d516a75510c9cc860a, RSA key 2048 bits, signed using RSA-SHA256, activated `2022-10-15 10:20:18 UTC', expires `2023-10-15 10:20:18 UTC', pin-sha256="BxXuMV5+62tZeTQttxWTI5WZay3DUZBUn838wD2lqkE="
        Public Key ID:
                sha1:eb0ea86b9d77d3a0070fd218e059a3473635b03e
                sha256:0715ee315e7eeb6b5979342db715932395996b2dc35190549fcdfcc03da5aa41
        Public Key PIN:
                pin-sha256:BxXuMV5+62tZeTQttxWTI5WZay3DUZBUn838wD2lqkE=

- Status: The certificate is NOT trusted. The certificate issuer is unknown. The name in the certificate does not match the expected.
*** PKI verification of server certificate failed...
*** Fatal error: Error in the certificate.
```

```
lenny@DESKTOP-DMJ749K:/mnt/c/Users/lenny$ gnutls-cli galene.org
Processed 129 CA certificate(s).
Resolving 'galene.org:443'...
Connecting to '51.210.14.2:443'...
- Certificate type: X.509
- Got a certificate list of 3 certificates.
- Certificate[0] info:
 - subject `CN=galene.org', issuer `CN=R3,O=Let's Encrypt,C=US', serial 0x041457df226083251630943932 4ce5e41f23, RSA key 2048 bits, signed using RSA-SHA256, activated `2022-08-19 12:42:07 UTC', expires `2022-11-17 12:42:06 UTC', pin-sha256="884KqcQtRXujJoonuT1HWS1I65DNlzEijoOrP/H9Vrc="
        Public Key ID:
                sha1:459aee70c807e920af2cae68b31d1875effd1622
                sha256:f3ce0aa9c42d457ba3268a27b93d47592d48eb90cd9731228e83ab3ff1fd56b7
        Public Key PIN:
                pin-sha256:884KqcQtRXujJoonuT1HWS1I65DNlzEijoOrP/H9Vrc=

- Certificate[1] info:
 - subject `CN=R3,O=Let's Encrypt,C=US', issuer `CN=ISRG Root X1,O=Internet Security Research Group,C=US', serial 0x00912b084acf0c18a753f6d62e25a75f5a, RSA key 2048 bits, signed using RSA-SHA256, activated `2020-09-04 00:00:00 UTC', expires `2025-09-15 16:00:00 UTC', pin-sha256="jQJTbIh0grw0/1TkHSumWb+Fs0Ggogr621gT3PvPKG0="
- Certificate[2] info:
 - subject `CN=ISRG Root X1,O=Internet Security Research Group,C=US', issuer `CN=DST Root CA X3,O=Digital Signature Trust Co.', serial 0x4001772137d4e942b8ee76aa3c640ab7, RSA key 4096 bits, signed using RSA-SHA256, activated `2021-01-20 19:14:03 UTC', expires `2024-09-30 18:14:03 UTC', pin-sha256="C5+lpZ7tcVwmwQIMcRtPbsQtWLABXhQzejna0wHFr8M="
- Status: The certificate is NOT trusted. The certificate chain uses expired certificate.
*** PKI verification of server certificate failed...
*** Fatal error: Error in the certificate.
```

```
Lenny@DESKTOP-DMJ749K:/mnt/c/Users/Lenny$ gnutls-cli www.irif.fr
Processed 129 CA certificate(s).
Resolving 'www.irif.fr:443'...
Connecting to '81.194.27.171:443'...
- Certificate type: X.509
- Got a certificate list of 4 certificates.
- Certificate[0] info:
 - subject `CN=www.math.univ-paris-diderot.fr,O=Université de Paris,ST=Île-de-France,C=FR', issuer `CN=GEANT OV RSA CA 4,O=GEANT Vereniging,C=
NL', serial 0x00c9ee732d8009efc503bc0c0a1ea7e83a, RSA key 2048 bits, signed using RSA-SHA384, activated `2022-03-03 00:00:00 UTC', expires `20
23-03-03 23:59:59 UTC', pin-sha256="4bexhU3V0bp81/aHHk+UStPeueYpd5USOeHcxln+6Qw="
        Public Key ID:
                sha1:66868af3cbf7899cf260500392bd8f84da4d824f
                sha256:e1b7b1854dd5d1ba7cd7f6871e4f944ad3deb9e62977951239e1dcc659fee90c
        Public Key PIN:
                pin-sha256:4bexhU3V0bp81/aHHk+UStPeueYpd5USOeHcxln+6Qw=

- Certificate[1] info:
 - subject `CN=GEANT OV RSA CA 4,O=GEANT Vereniging,C=NL', issuer `CN=USERTrust RSA Certification Authority,O=The USERTRUST Network,L=Jersey C
ity,ST=New Jersey,C=US', serial 0x00da43bd139bd258bb4dd61cacc4f3dbe0, RSA key 4096 bits, signed using RSA-SHA384, activated `2020-02-18 00:00:
00 UTC', expires `2033-05-01 23:59:59 UTC', pin-sha256="j0qRK9S0oUba9b4ttZdKp42Q4T2J8S4FFKPNG5FTFVA="
- Certificate[2] info:
 - subject `CN=USERTrust RSA Certification Authority,O=The USERTRUST Network,L=Jersey City,ST=New Jersey,C=US', issuer `CN=AAA Certificate Ser
vices,O=Comodo CA Limited,L=Salford,ST=Greater Manchester,C=GB', serial 0x3972443af922b751d7d36c10dd313595, RSA key 4096 bits, signed using RS
A-SHA384, activated `2019-03-12 00:00:00 UTC', expires `2028-12-31 23:59:59 UTC', pin-sha256="x4QzPSC810K5/cMjb05Qm4k3Bw5zBn4lTdO/nEW/Td4="
- Certificate[3] info:
 - subject `CN=AAA Certificate Services,O=Comodo CA Limited,L=Salford,ST=Greater Manchester,C=GB', issuer `CN=AAA Certificate Services,O=Comod
o CA Limited,L=Salford,ST=Greater Manchester,C=GB', serial 0x01, RSA key 2048 bits, signed using RSA-SHA1 (broken!), activated `2004-01-01 00:
00:00 UTC', expires `2028-12-31 23:59:59 UTC', pin-sha256="vRU+17BDT2iGsXvOi76E7TQMcTLXAqj0+jGPdW7L1vM="
- Status: The certificate is trusted.
- Description: (TLS1.2-X.509)-(ECDHE-SECP256R1)-(RSA-SHA256)-(AES-256-GCM)
- Session ID: FC:62:F2:21:79:D6:86:B4:CE:CF:09:34:2B:34:B5:F6:B8:9F:2A:96:12:25:49:89:09:3F:4B:34:C3:77:9B:D2
- Options: safe renegotiation,
- Handshake was completed

- Simple Client Mode:

- Peer has closed the GnuTLS connection
```



Le sans état : L'URI (*Uniform Resource Identifier*), qui identifie une ressource, et ses éventuels paramètres sont suffisants pour afficher une ressource ou effectuer une action sur une ressource. Il ne doit pas y avoir de notion de session, et les cookies ne doivent pas servir non plus à stocker un état.

Source : *Développer des services REST en Java : Échanger des données au format JSON / Sobrero Aurélie*