oneM2M - SSL/TLS Enablement

Yungi Park (yun0429@keti.re.kr)

Korea Electronics Technology Institute

2020.11.13

The project "International Digital Cooperation - ICT Standardisation" is funded by the European Union

© 2020 oneM2M

Motivation

- oneM2M platform is expanded, and high confidentiality is required from users while communicating with various devices
- oneM2M can provide confidentiality of data transmitted based on SSL/TLS
- In this tutorial, we will show you how data is encrypted via SSL/TLS

© 2020 oneM2M

Outline

- Prerequisites
- oneM2M – SSL/TLS Enablement
  - Mobius SSL/TLS Enabling Overview
  - openSSL Installation
  - Certificate Creation
  - Mosquitto SSL/TLS Enabling
  - Mobius SSL/TLS Enabling

© 2020 oneM2M

Prerequisites

- S/W
  - Node.js
  - Visual Studio Code
    - https://code.visualstudio.com/download
  - cmder(Terminal)
    - https://cmder.net/
  - Postman
    - https://www.postman.com/downloads/
  - oneM2M-API-Testing Source
    - https://github.com/IoTKETI/oneM2M-API-Testing.git

© 2020 oneM2M

oneM2M – SSL/TLS Enablement

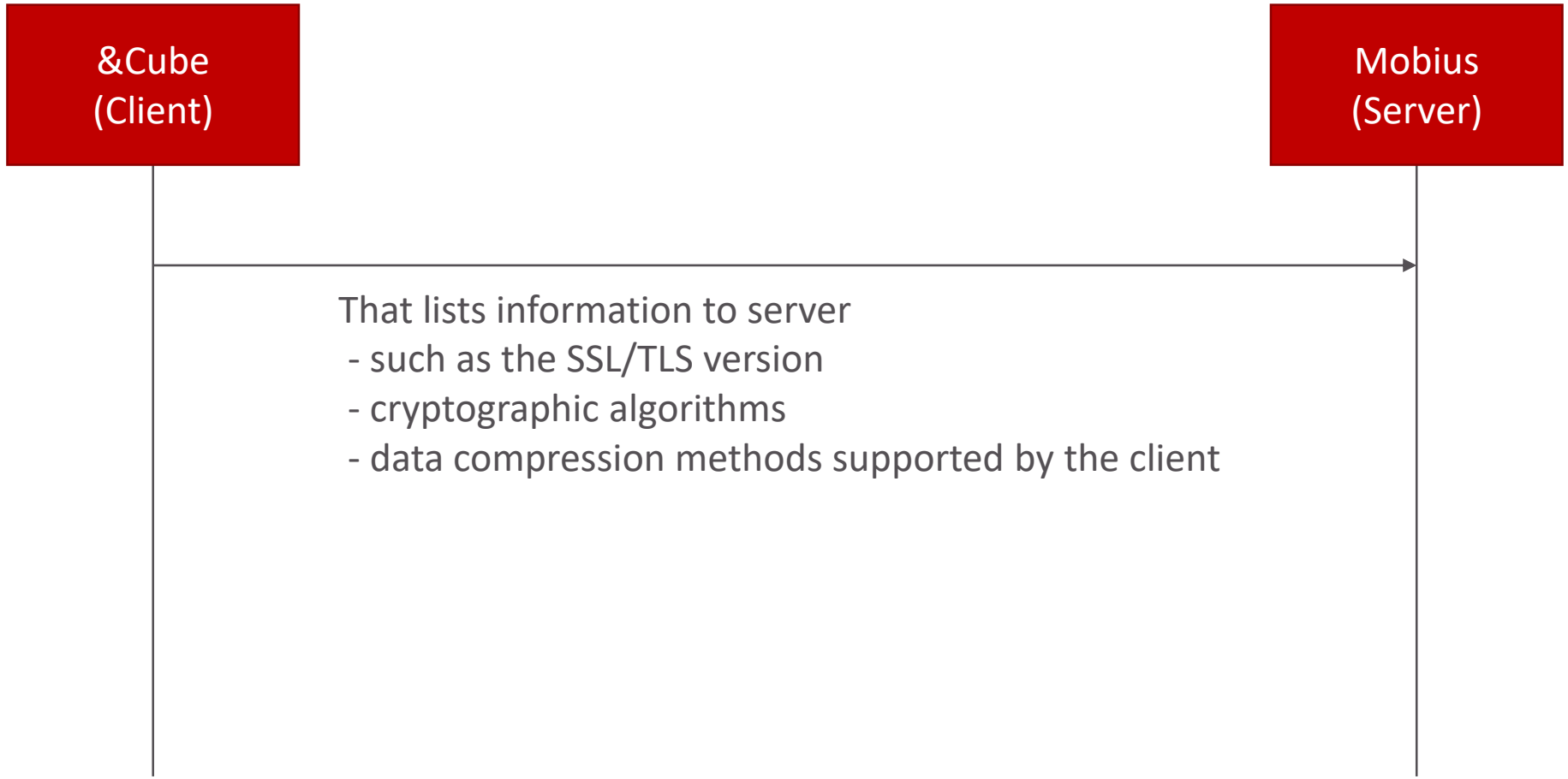
Mobius SSL/TLS Overview

- Mobius supports confidential and intact data transmission using HTTPs and MQTTs protocol based on Secure Sockets Layer / Transport Layer Security (SSL/TLS)
- Commercial servers use certificates that are issued by a validated Certificate Authority (CA) to use SSL/TLS, but for non-commercial purposes, the openSSL can be used which is free
- This tutorial provides the guide on generating certificates using openSSL and applying to the oneM2M communications

© 2020 oneM2M

SSL/TLS flow

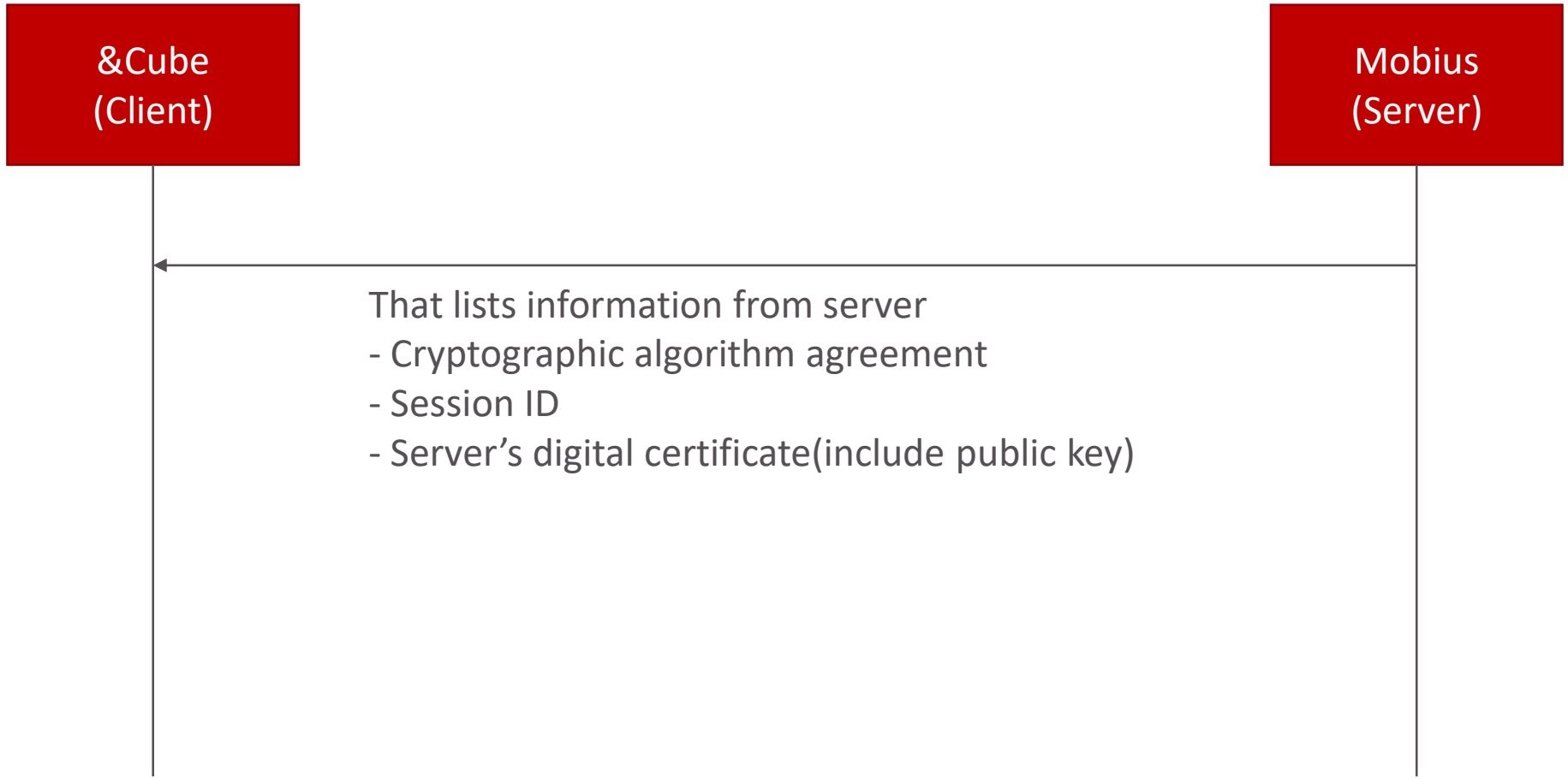
- The &Cube sends a “Client Hello” message



© 2020 oneM2M

Data exchange flow

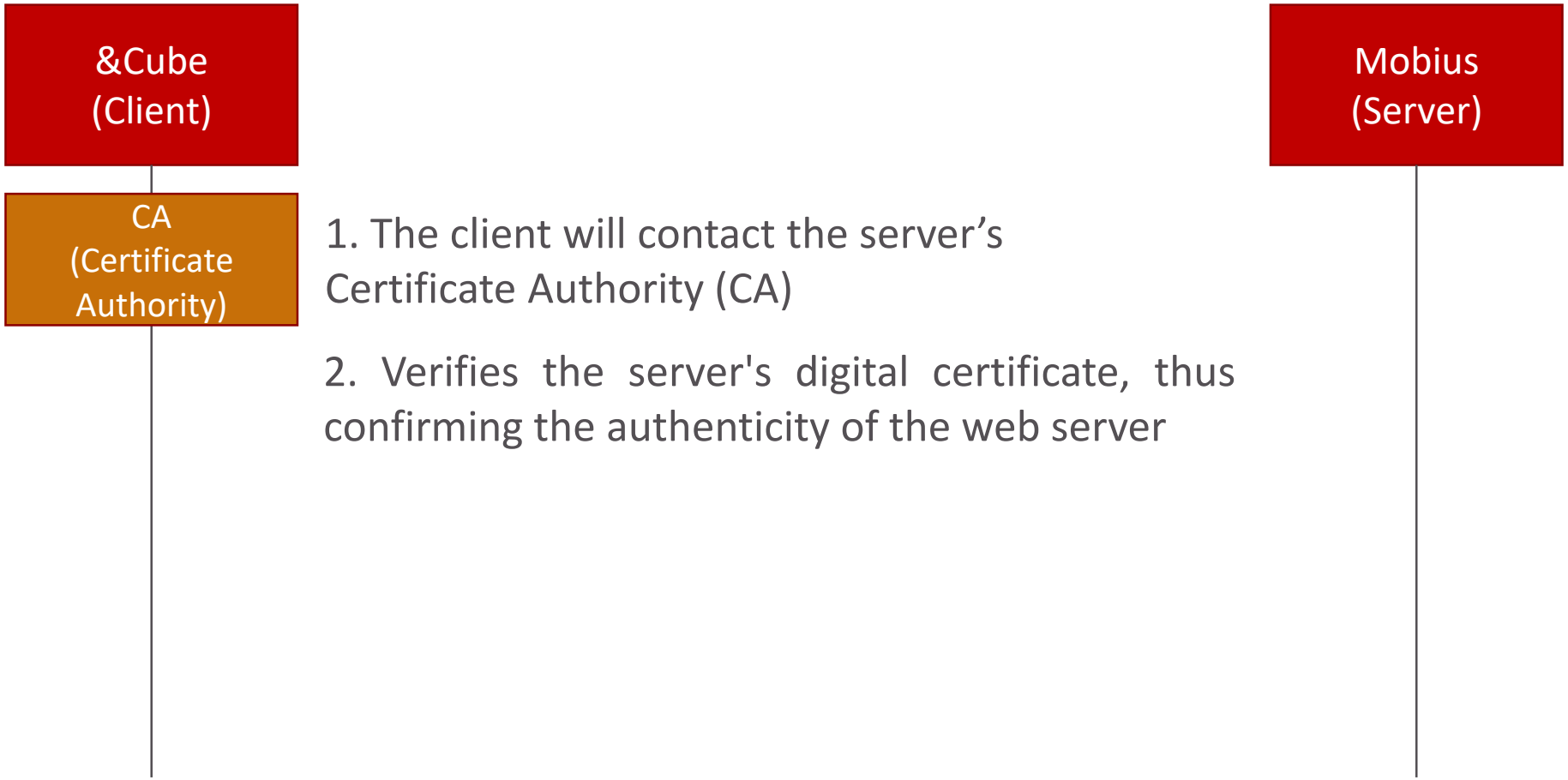
- The Mobius responds a message



© 2020 oneM2M

Data exchange flow

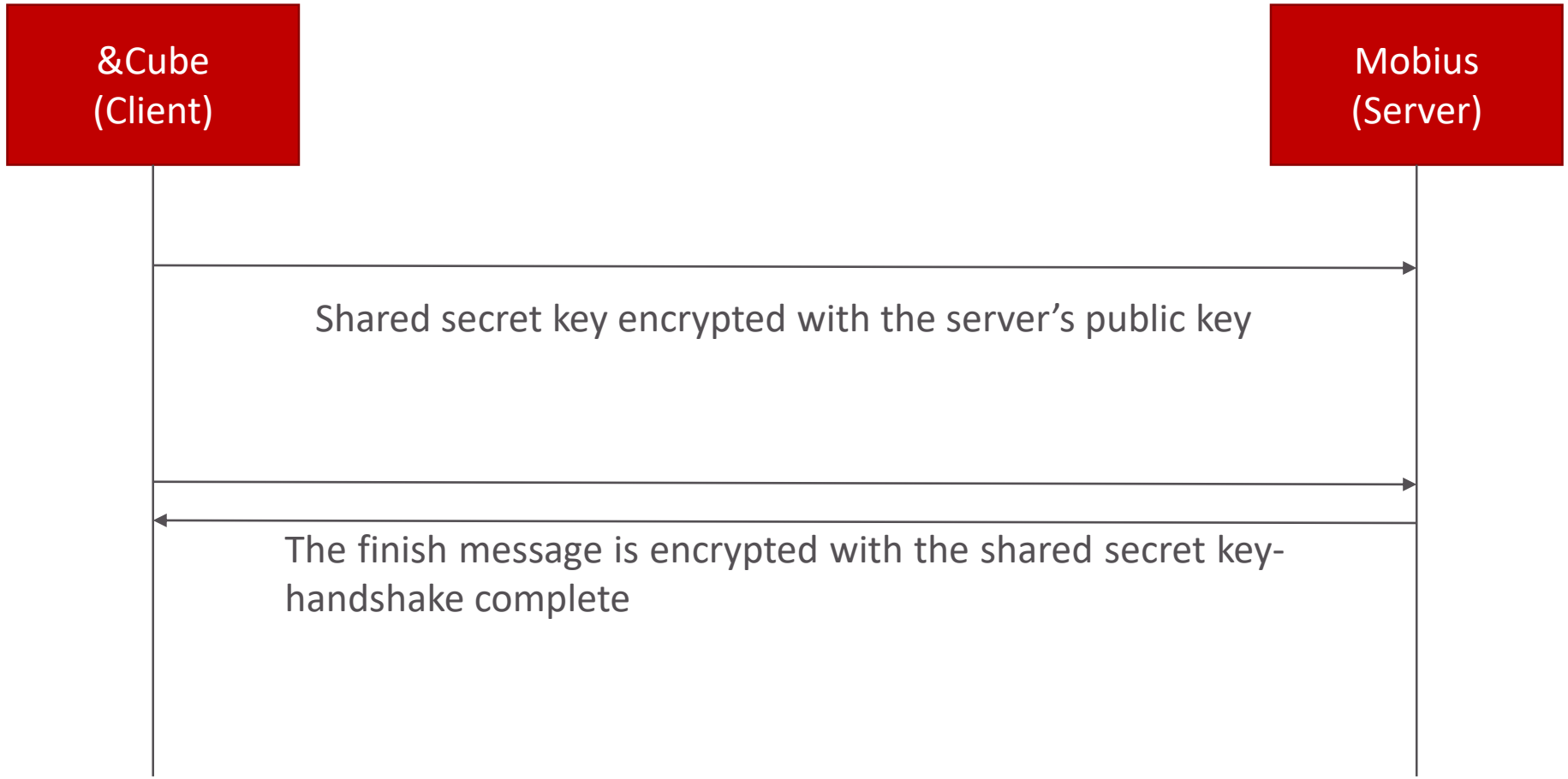
- Authenticity check



© 2020 oneM2M

Data exchange flow

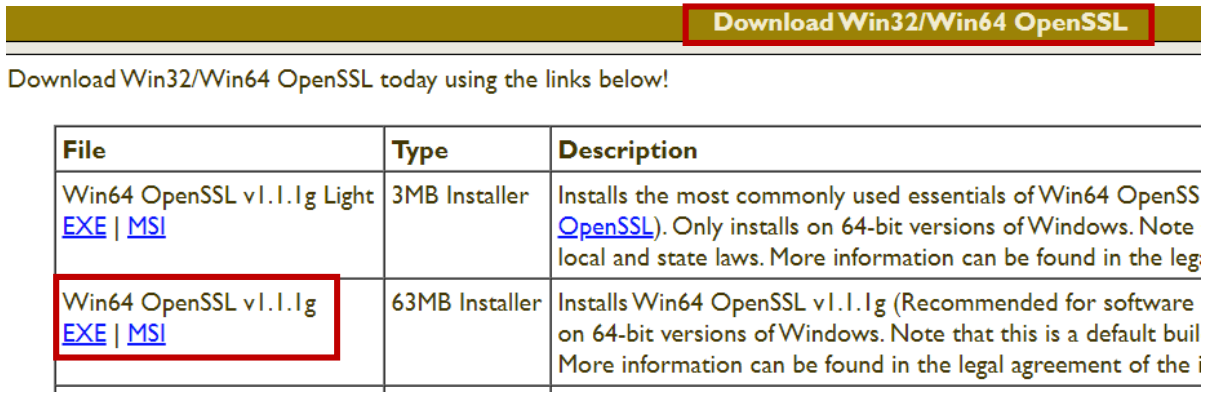
- Client key exchange and finished

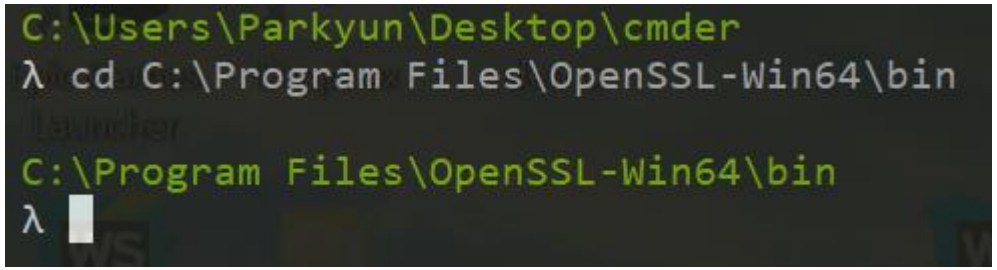


© 2020 oneM2M

openSSL Installation(Windows)

- Download opsnSSL for Windows exe file
  - https://slproweb.com/products/Win32OpenSSL.html
  - Install the openSSL
  - Open CMD and move to the installed directory
    - > cd C:\Program Files\OpenSSL-Win64\bin





© 2020 oneM2M

openSSL Installation(Linux,macOS)

- Linux(Ubuntu)
  - Open terminal and enter command below.
    - > sudo apt-get install libssl-dev openssl
- MacOS
  - To download and install openSSL, brew is recommended
  - About brew installation, please refer to the brew official website
  - After brew installation, open terminal and enter command below
    - > sudo apt-get install libssl-dev openssl

© 2020 oneM2M

Certificate Creation

- CA key pair and certificate Creation(1)
  - Set path in the openssl.cnf file before Certificate Creation
    - > set OPENSSL_CONF=C:\[PATH TO YOUR OPENSSL DIRECTORY]\bin\openssl.cfg

```
C:\Program Files\OpenSSL-Win64\bin
λ set OPENSSL_CONF=C:\Program Files\OpenSSL-Win64\bin\openssl.cfg
```

© 2020 oneM2M

Certificate Creation

- CA key pair and certificate Creation(1)
  - On CMD or terminal, enter below command to create CA key pair
    - > openssl genrsa -out ca-key.pem 4096

```
C:\Program Files\OpenSSL-Win64\bin
λ openssl genrsa -out ca-key.pem 4096
Generating RSA private key, 4096 bit long modulus (2 primes)
.....................................................................
.....................................................++++
.....................................................................
e is 65537 (0x010001)
```

© 2020 oneM2M

Certificate Creation

- CA key pair and certificate Creation(1)
  - Enter below command to create CA certificate
    - > openssl req -new -x509 -days 9999 -key ca-key.pem -out ca.crt
  - To create CA certificate, some information are required

```
C:\Program Files\OpenSSL-Win64\bin
λ openssl req -new -x509 -days 9999 -key ca-key.pem -out ca.crt
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]:KR
State or Province Name (full name) [Some-State]:Gyeonggi-do
Locality Name (eg, city) []:Seongnam-si
Organization Name (eg, company) [Internet Widgits Pty Ltd]:KETI
Organizational Unit Name (eg, section) []:IoT Platform Research Center
Common Name (e.g. server FQDN or YOUR name) []:localhost
Email Address []:yun0429@keti.re.kr
```

© 2020 oneM2M

Certificate Creation

- Server key pair, csr and Certificate Creation(2)
  - To create server key pair, enter command below
    - > openssl genrsa -out server-key.pem 4096

```
C:\Program Files\OpenSSL-Win64\bin
λ openssl genrsa -out server-key.pem 4096
Generating RSA private key, 4096 bit long modulus (2 primes)
.....................................................................++++
.....................................................................
.....................++++
e is 65537 (0x010001)
```

© 2020 oneM2M

Certificate Creation

- Server key pair, csr and Certificate Creation(2)
  - Enter below command to create csr which will be needed to create server certificate
    - > openssl req -new -out server.csr -key server-key.pem
  - To create csr file, some information are required. Input your information
  - In this section, What you have to note is that you have to enter your server's domain name on Common Name field

```
C:\Program Files\OpenSSL-Win64\bin
λ openssl req -new -out server.csr -key server-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]:KR
State or Province Name (full name) [Some-State]:Gyeonggi-do
Locality Name (eg, city) []:Seongnam-si
Organization Name (eg, company) [Internet Widgits Pty Ltd]:KETI
Organizational Unit Name (eg, section) []:IoT Platform Research Center
Common Name (e.g. server FQDN or YOUR name) []:localhost
Email Address []:yun0429@keti.re.kr

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

© 2020 oneM2M

Certificate Creation

- Server key pair, csr and Certificate Creation(2)
  - Create server certificate using server-ser.pem and ca-crt.pem
    - > openssl x509 -req -in server.csr -CA ca.crt -CAkey ca-key.pem -CAcreateserial -out server.crt -days 9999

```
C:\Program Files\OpenSSL-Win64\bin
λ openssl x509 -req -in server-csr.pem -CA ca-crt.pem -CAkey ca-key.pem -CAcreateserial -out server-crt.pem -days 9999
Signature ok
subject=C = KR, ST = Gyeonggi-do, L = Seongnam-si, O = KETI, OU = IoT Platform Research Center, CN = localhost, emailAddress = yun0429@keti.re.kr
Getting CA Private Key
```

© 2020 oneM2M

Certificate Creation

- Server key pair, csr and Certificate Creation(2)
  - Use the following command to check the contents of the server.crt file
    - > openssl x509 –in server.crt –text -noout

```
C:\Program Files\OpenSSL-Win64\bin
λ openssl x509 -in server.crt -text -noout
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number:
            5c:ff:88:07:8c:0f:91:75:0a:cb:c0:a5:6a:8d:85:72:4b:e5:88:86
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = KR, ST = Gyeonggi-do, L = Seongnam-si, O = KETI, OU = IoT Platfor
m Research Center, CN = localhost, emailAddress = yun0429@keti.re.kr
        Validity
            Not Before: Oct  7 12:20:01 2020 GMT
            Not After : Feb 22 12:20:01 2048 GMT
        Subject: C = KR, ST = Gyeonggi-do, L = Seongnam-si, O = KETI, OU = IoT Platfo
rm Research Center, CN = localhost, emailAddress = yun0429@keti.re.kr
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (4096 bit)
                Modulus:
                    00:ba:c4:65:eb:c9:2b:57:de:6e:11:6b:49:17:e8:
                    9c:d0:43:e7:f7:a3:e8:de:8c:8c:4b:70:90:fb:a8:
                    67:46:e8:2a:4b:da:2a:eb:0d:c0:80:b8:2a:e1:e2:
                    04:a6:54:09:ab:4c:db:6a:0d:6a:b7:cc:63:78:16:
                    dd:c8:92:36:53:ba:ff:bd:6f:3f:47:df:bf:cb:52:
                    cc:a5:ef:c8:a8:14:e3:fa:e9:79:af:e4:a0:b6:b1:
```

© 2020 oneM2M

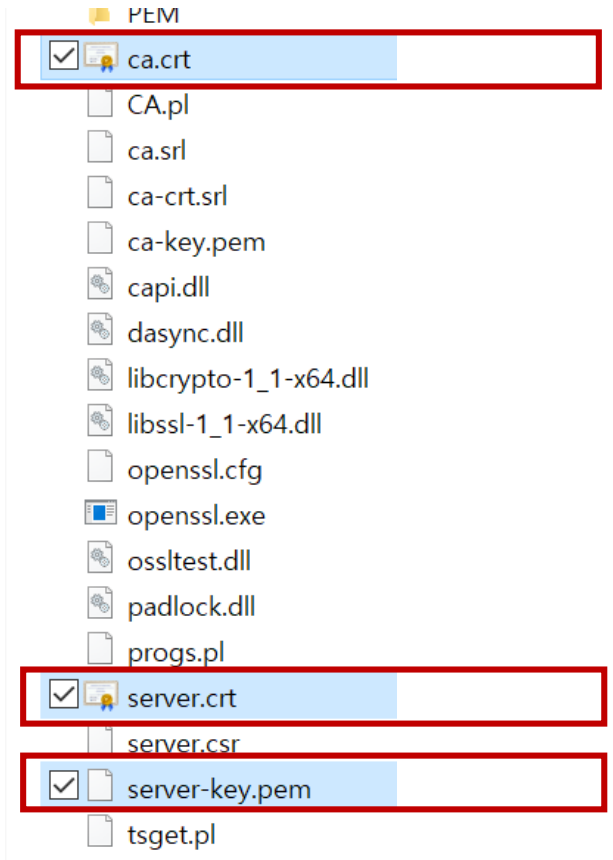
Mosquitto SSL/TLS Enablement

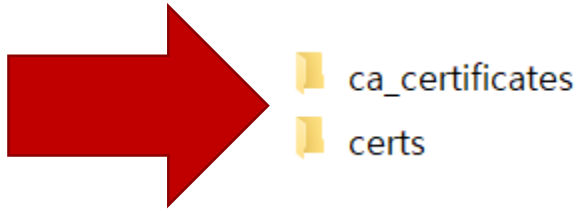
- To enable SSL/TLS on mosquitto, some steps are required
  - Create cert,ca_certificates folder in your mosquitto directory
  - Copy and paste the ca.crt file into the ca_certificates directory under in C:\Program Files (x86)\mosquitto
  - Copy and paste the server-key.pem, server.crt files into the certs directory under C:\Program Files (x86)\mosquitto
    - (For linux, mosquitto.conf file exist in /etc/mosquitto)
      (For macOS and 1.4.14_2 version of mosquitto, mosquitto.conf file exist in /usr/local/Cellar/mosquitto/1.4.14_2/etc/mosquitto)

© 2020 oneM2M

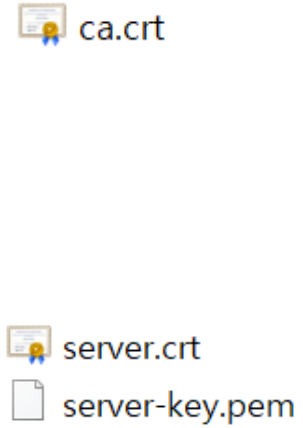
Mosquitto SSL/TLS Enablement

- C:\Program Files\OpenSSL-Win64\bin





- C:\Program Files\mosquitto\



© 2020 oneM2M

Mosquitto SSL/TLS Enablement

- Edit mosquitto.conf to append SSL/TLS setting scripts
  (cafile, certfile, keyfile should be exact path where each files placed. Below example is for Window)
  - Conf path : C:\Program Files\mosquitto\mosquitto.conf

```
listener 8883
cafile C:\Program Files\mosquitto\ca_certificates\ca.crt
certfile C:\Program Files\mosquitto\certs\server.crt
keyfile C:\Program Files\mosquitto\certs\server-key.pem
```

© 2020 oneM2M

Mosquitto SSL/TLS Enablement

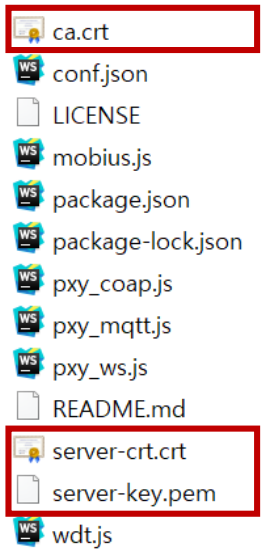
- Open CMD or terminal and enter into the directory where mosquitto.conf placed
  - Run mosquitto with command below
    - > net stop mosquitto
    - > mosquitto -c mosquitto.conf -v

```
C:\Program Files\mosquitto
λ mosquitto -c ./mosquitto.conf -v
1600849520: mosquitto version 1.6.9 starting
1600849520: Config loaded from ./mosquitto.conf.
1600849520: Opening ipv6 listen socket on port 8883.
1600849520: Opening ipv4 listen socket on port 8883.
```

© 2020 oneM2M

Mobius SSL/TLS Enablement

- To enable SSL/TLS on Mobius, some steps are required
  - Delete server-key.pem, server.crt, ca.crt files which are downloaded with Mobius
  - Copy and paste server-key.pem, server.crt, ca.crt that you created using openssl to Mobius root directory
  - Edit the code in Mobius.js like below



© 2020 oneM2M

Mobius SSL/TLS Enablement

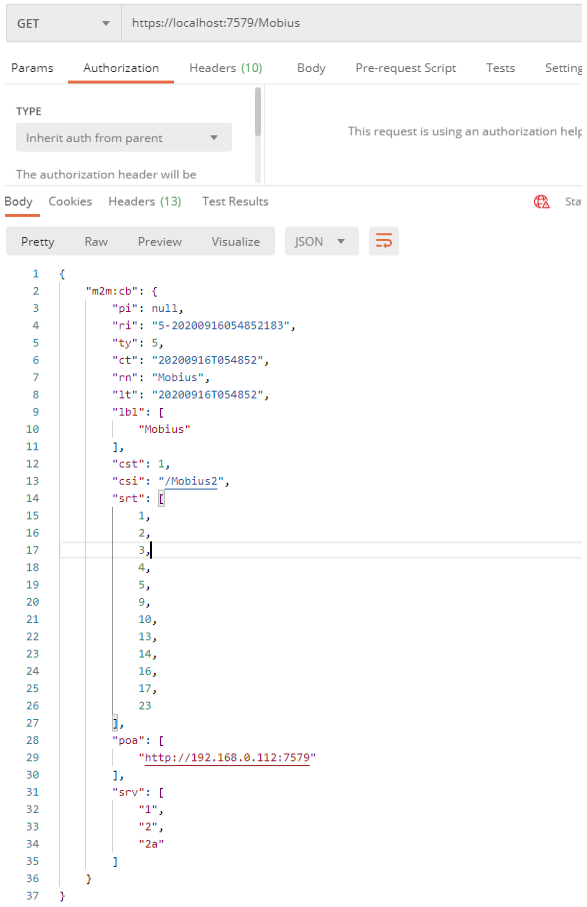
- Run Mobius with command below
  - > node mobius.js

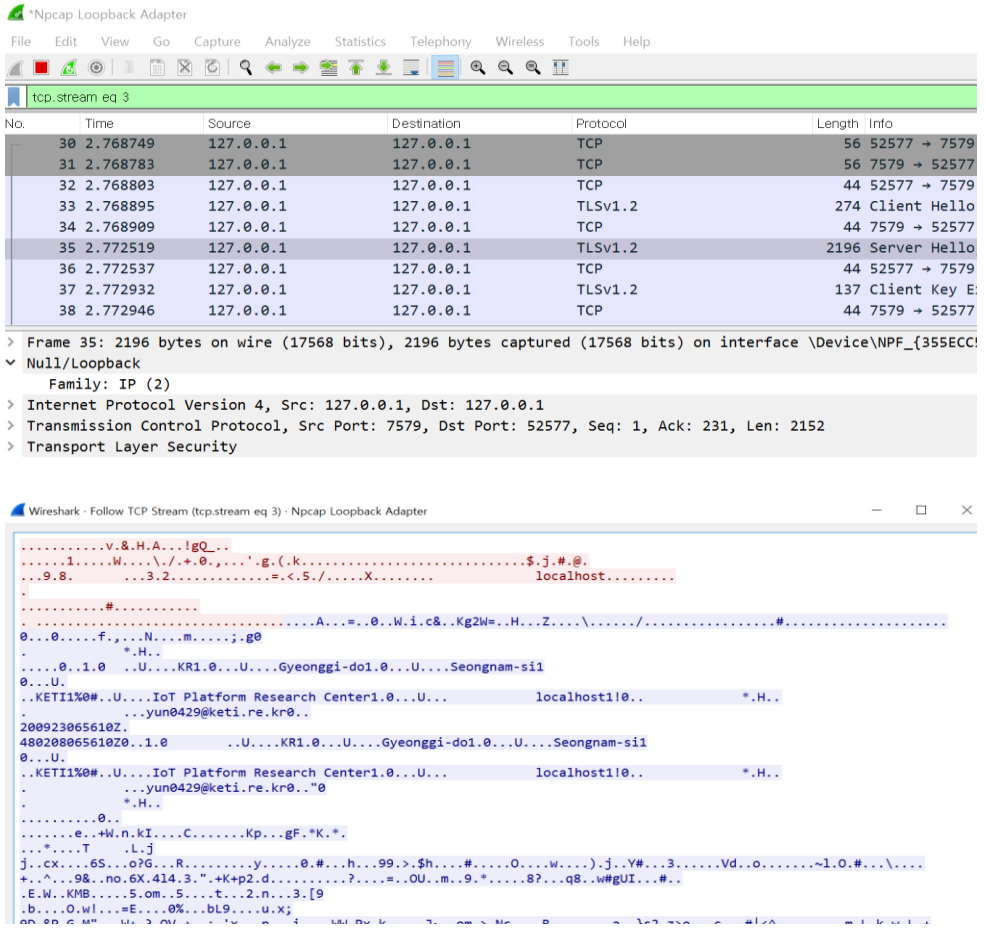
```
Production Mode
Garbage collection is not exposed
CPU Count: 12
select_ri_lookup /Mobius: 2.609ms
update_cb_poa_csi /Mobius: 2.995ms
""
sgn_mqtt_client is connected
pxymqtt server (192.168.0.112) running at 7578 port
pxyws server (192.168.0.112) running at 7577 port
```

© 2020 oneM2M

Mobius SSL/TLS Enablement

- Also you can see that all of request and response packets are encrypted

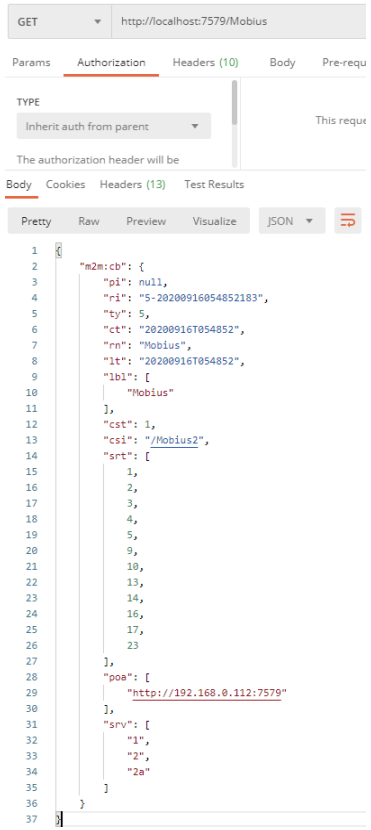




© 2020 oneM2M

Mobius SSL/TLS Enablement

- You can see the result like the picture when you check the packet of http, not https



© 2020 oneM2M