Seguridad y Protección de Sistemas Informáticos

Práctica 4: Certificados digitales

Francisco Fernández Millán

Seguridad y Protección de Sistemas Informáticos

Práctica 4: Certificados digitales

Descripción de los archivos generados (obtenido del manual de gestión de certificados digitales con OpenSSL):

cakey.pem: clave privada de la autoridad certificadora.

cacert.pem: certificado de la autoridad certificadora.

cliente.csr: solicitud de certificado del servidor.

cliente.crt: certificado del servidor, firmado por la CA.

1. Cread una autoridad certificadora. En este caso se premiará el uso de openssl ca frente a CA.pl , aunque este último comando es admisible.

En primer lugar, para hacer uso de un certificado primero tenemos que tener una Autoridad Certificadora (CA). Ésta autoridad será la que se encargue de validar y confirmar que nuestro certificado es válido.

Creamos la CA con la siguiente orden:

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer1$ openssl req -x5
09 -newkey rsa:2048 -keyout cakey.pem -days 3650 -out cacert.pem
Generating a 2048 bit RSA private key
.....................................+++
.................................................+++
writing new private key to 'cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
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]:ES
State or Province Name (full name) [Some-State]:Granada
Locality Name (eg, city) []:Granada
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:franfermi@correo.ugr.es
```

Generamos una nueva clave de tipo RSA para la CA con una duración de 10 años.

2. Cread una solicitud de certificado que incluya la generación de claves en la misma.

Una vez tengamos nuestra CA podemos realizar la solicitud de un certificado, el siguiente paso es la creación de una petición de certificado para nuestra clave pública, recordemos que una petición de certificado es un fichero que incluye una clave pública junto con los datos de la entidad que posee la clave privada asociada. En la misma solicitud incluimos la generación de las claves de la misma.

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer2$ openssl req -newkey rsa:204
8 -out petic-cert-client.csr
Generating a 2048 bit RSA private key
.........................+++
......+++
writing new private key to 'privkey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
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]:ES
State or Province Name (full name) [Some-State]:Grananda
Locality Name (eg, city) []:Granada
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:franfermi@correo.ugr.es
```

3. Cread un certificado para la solicitud anterior empleando la CA creada en el primer punto.

Por último creamos el certificado a partir de la CA del primer punto y la solicitud del punto anterior.

Le indicamos que será un certificado del tipo **x509** cuya CA está definida en el fichero **cacert.pem** y que usa como clave privada en **-CAkey** el fichero **cakey.pem** y por último, se especifica que el certificado el cual queremos generar tendrán las especificaciones definidas en **-req -in** con el archivo **client-cert.crt**. Podemos observar que la verificación es correcta.

Cuando nos pide que introduzcamos la clave, tenemos que añadir la que elegimos en la generación de la CA.

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer3$ openssl x509 -CA cacert.pem
 -CAkey cakey.pem -req -in petic-cert-client.csr -sha1 -out client-cert.crt
Signature ok
subject=/C=ES/ST=Grananda/L=Granada/O=Internet Widgits Pty Ltd/emailAddress=franfermi@correo
.ugr.es
Getting CA Private Key
Enter pass phrase for cakey.pem:
```

4. Cread una solicitud de certificado para cualquiera de las claves que habéis generado en las prácticas anteriores, excepto las RSA .

En mi caso he escogido la clave generada a partir de los parámetros de una curva elíptica. Con dicha clave generamos la solicitud de certificado a la CA.

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer4$ openssl req -new -sha256 -k
ey SPSI-ECkey.pem -out petic-cert-FranciscoEC.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]:ES
State or Province Name (full name) [Some-State]:Granada
Locality Name (eg, city) []:Granada
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:franfermi@correo.ugr.es

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

5. Cread un certificado para la solicitud anterior utilizando la CA creada.

Una vez enviado la solicitud de certificado a la CA creada, ésta crea el certificado para el solicitante.

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer5$ openssl x509 -CA cacert.pem
 -CAkey cakey.pem -req -in petic-cert-FranciscoEC.csr -sha1 -out cert-FranciscoEC.crt
Signature ok
subject=/C=ES/ST=Granada/L=Granada/O=Internet Widgits Pty Ltd/emailAddress=franfermi@correo.
ugr.es
Getting CA Private Key
Enter pass phrase for cakey.pem:
```

6. Emplead las opciones -text y -noout para mostrar los valores de todos los certificados y solicitudes de los puntos anteriores, incluyendo el certificado raíz que habrá sido creado junto con la CA.

-Punto 1:

Podemos ver los valores del certificado de la CA generada. Entre los datos que se muestran podemos señalar los datos pertenecientes al propietario como son el país, la provincia, correo electrónico…, también podemos observar la fecha de validación, el tamaño de la clave pública que hemos generado de 2048 bit de tipo rsa. La última parte pertenece al certificado digital junto con las claves de identificación.

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer1$ openssl x509 -inform PEM -i
n cacert.pem -text -noout
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 18359676570192384049 (0xfecaac911d611031)
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=ES, ST=Granada, L=Granada, O=Internet Widgits Pty Ltd/emailAddress=franfer
mi@correo.ugr.es
        Validity
            Not Before: Dec  1 07:49:15 2017 GMT
            Not After : Nov 29 07:49:15 2027 GMT
        Subject: C=ES, ST=Granada, L=Granada, O=Internet Widgits Pty Ltd/emailAddress=franfe
rmi@correo.ugr.es
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:ab:f9:50:60:ce:c5:46:8a:3f:92:61:8b:70:db:
                    10:1d:e9:5f:d6:5f:8a:93:79:58:18:10:01:08:52:
                    40:47:13:15:71:70:36:91:fd:f7:6d:6e:f7:e1:d7:
                    af:b6:a4:42:a7:26:d5:bb:fc:0a:84:8c:eb:6f:0d:
                    77:fc:a5:46:38:5b:a5:6d:40:95:70:c0:ae:29:af:
                    29:a2:43:30:d1:88:6c:ab:04:79:d0:53:6b:fe:45:
                    eb:fd:ba:a0:4b:10:ca:c5:f7:56:3e:11:69:90:9b:
                    4d:af:5e:f8:67:65:61:5c:66:cb:ff:20:a3:d5:be:
                    af:e8:4a:4c:c3:12:8a:f7:43:fb:3c:fd:8e:cc:0f:
                    79:af:42:bc:0c:c6:8d:1d:c8:7c:59:47:41:de:29:
                    31:dc:9a:a8:3c:95:eb:e9:02:ad:c8:90:5d:4a:53:
                    9f:a6:51:86:b8:0d:6c:6b:39:0f:d6:88:34:90:e0:
                    bc:0d:eb:b7:2a:8e:c5:eb:89:4b:f7:33:37:50:ca:
                    7a:6b:03:43:2b:38:98:64:e8:fc:aa:d9:72:3c:98:
                    d3:cd:fa:38:26:a0:19:25:ba:a6:f0:a2:49:71:69:
                    51:39:30:4f:0b:f8:34:17:d5:33:ab:4c:c8:47:0e:
                    b6:3c:96:dc:19:c3:fb:90:34:cc:18:36:59:74:61:
                    25:01
                Exponent: 65537 (0x10001)
            X509v3 Subject Key Identifier:
                23:01:1D:3A:15:74:23:C7:9A:A3:64:76:0F:90:C9:EB:C3:A7:6F:FD
            X509v3 Authority Key Identifier:
                keyid:23:01:1D:3A:15:74:23:C7:9A:A3:64:76:0F:90:C9:EB:C3:A7:6F:FD

            X509v3 Basic Constraints:
                CA:TRUE
    Signature Algorithm: sha256WithRSAEncryption
         45:34:4f:16:ba:5e:80:d5:4b:d3:2a:a7:42:89:2f:58:36:5a:
         84:71:22:81:7d:d5:a1:85:df:d1:d2:bc:0d:e9:c1:05:bb:71:
         8a:5b:e9:3c:9f:c6:0e:5b:99:30:58:45:b8:70:00:04:69:a5:
         4a:5f:84:b6:f5:c6:91:60:6d:e4:78:6c:fb:98:e2:f6:ec:64:
         ed:c5:88:2b:46:af:f0:cf:fc:6e:80:63:df:8a:8c:46:4c:63:
         9c:81:3d:43:5e:0e:9d:ae:5f:98:3e:2e:47:c4:44:f7:8d:9e:
         6e:56:0c:bd:dc:a5:81:ea:4f:fe:ae:e8:b9:23:f4:e7:e5:65:
         2e:29:cd:a3:01:b4:15:ce:50:6d:9f:6d:88:dc:95:9a:46:83:
         53:0a:bc:4b:93:71:a4:a4:db:f0:41:02:de:e5:53:a2:19:81:
         2a:03:79:9c:dd:25:a6:d0:41:41:bd:a0:5e:9c:4c:f1:82:30:
         7c:84:d3:d6:5a:4c:7a:df:02:aa:e3:07:e7:5f:c8:00:48:c5:
         68:01:af:e0:18:7d:1c:dd:1e:46:3c:d0:d2:dd:2c:f9:80:ea:
         47:29:10:e0:56:37:ea:12:03:61:8b:3d:d0:25:f9:4a:8e:5b:
         36:00:4c:c6:b2:fa:47:62:0e:1d:a3:70:a2:66:80:a7:f8:f8:
         e0:dd:ab:62
```

-Punto 2:

En los valores mostrados podemos observar los campos como **Subject**, en el que se muestran los campos del propietario así como su localización y correo electrónico, el campo **Public-key**, que muestra la clave pública codificada en hexadecimal. Y al final se muestra la firma digital que ha sido calculada con la clave privada que corresponde a la clave pública que se pretende certificar.

De esta forma, la CA se asegura de que el solicitante de la petición conoce la clave privada.

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer2$ openssl req -in petic-cert-
client.csr -text -noout
Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: C=ES, ST=Grananda, L=Granada, O=Internet Widgits Pty Ltd/emailAddress=franf
ermi@correo.ugr.es
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:e9:89:4f:15:39:ff:ae:f6:66:41:61:cf:9b:77:
                    ab:d2:b0:5c:0a:9b:0f:5b:ac:a0:ac:ae:79:91:c5:
                    d4:56:88:bc:7a:0b:2d:49:76:f4:ec:a3:b8:cd:23:
                    16:fa:16:87:6c:77:81:a7:4c:7e:61:cd:7a:4e:58:
                    69:a6:61:09:ef:3e:1e:00:71:6c:87:fa:a1:a3:79:
                    48:63:69:b6:41:87:b3:52:14:89:29:f0:b0:ba:b2:
                    57:81:20:5b:89:80:92:50:b7:d0:6c:54:2d:dd:d3:
                    9f:aa:72:99:d2:05:0f:20:99:ee:2d:db:b2:5c:55:
                    64:11:0f:26:3b:d7:f3:2b:b4:e0:b4:56:65:69:4e:
                    d8:77:54:e5:55:84:07:f8:31:2e:3b:80:3a:32:0b:
                    07:d5:0c:32:c7:83:94:c6:c6:3c:52:1f:46:1e:c9:
                    9b:cc:bc:ef:1a:49:b7:b4:6a:29:fe:62:86:fb:cc:
                    eb:bd:0b:d3:32:c7:7f:e8:84:5a:91:e7:6e:98:8e:
                    f5:39:20:77:fa:17:28:fd:37:7c:ca:79:cb:95:4e:
                    0e:7f:4d:85:ff:1c:b6:18:d2:51:80:b2:4c:5f:9e:
                    e6:a5:ba:1c:c4:09:cb:f4:31:85:9d:ea:36:f2:bd:
                    04:d6:8b:eb:23:4b:1e:0c:d5:62:18:c9:d5:8f:9d:
                    57:2f
                Exponent: 65537 (0x10001)
        Attributes:
            a0:00
    Signature Algorithm: sha256WithRSAEncryption
         14:b3:7c:00:72:b9:2f:ea:5f:75:b6:a2:44:86:4f:18:e4:6b:
         1f:cd:86:8e:c1:58:f7:9d:bb:e8:27:42:4e:88:32:39:cf:71:
         99:0f:01:c6:b2:70:6c:5c:e1:e1:14:b1:76:0c:3b:a9:4b:ec:
         d6:49:61:e4:be:a2:b6:e2:f5:57:a5:72:c2:a3:27:9e:50:8d:
         62:e4:04:db:7c:c5:00:bf:a6:71:83:e4:b9:51:0e:68:43:00:
         87:0f:6c:2f:9f:62:c7:74:5c:64:92:aa:e5:02:39:85:b3:30:
         13:3e:44:be:3f:c8:4d:2a:55:3f:ec:63:28:5d:f6:9e:30:3f:
         26:73:6b:45:05:1a:6e:52:54:20:5f:72:12:a2:42:d0:38:94:
         a7:ef:ff:2e:d5:b7:7b:da:6d:49:d3:ec:8f:16:2e:09:1a:a7:
         90:1f:a5:8f:57:91:d9:d7:42:c2:b6:bd:f8:16:2e:9b:f0:2a:
         97:36:08:1b:9c:f4:b0:93:9c:be:80:0a:56:67:5a:c8:05:c8:
         5f:40:b0:69:27:10:ea:4d:57:ae:ea:4f:28:6e:eb:97:42:c0:
         f5:ca:b7:fc:8e:86:0a:33:1f:20:d0:8c:43:84:62:c8:82:79:
         3b:84:0a:72:12:6a:3b:bf:bc:5f:ad:e3:3a:76:38:00:2e:0f:
         83:e5:32:53
```

-**Punto 3:** En esta captura se muestran los valores obtenidos del certificado generado a partir de la solicitud. Podemos observar que los datos del propietario de la CA coinciden con el certificado generado.

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer3$ openssl x509 -in client-cer
t.crt -text -noout
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number: 13221353228152655977 (0xb77baeb61600d069)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=ES, ST=Granada, L=Granada, O=Internet Widgits Pty Ltd/emailAddress=franfer
mi@correo.ugr.es
        Validity
            Not Before: Dec  1 07:58:59 2017 GMT
            Not After : Dec 31 07:58:59 2017 GMT
        Subject: C=ES, ST=Granada, L=Granada, O=Internet Widgits Pty Ltd/emailAddress=franfe
rmi@correo.ugr.es
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:eb:aa:05:00:51:87:9d:6e:33:7a:69:4d:93:8b:
                    e6:98:60:6d:37:1c:3a:8e:14:fa:57:2d:45:e1:7f:
                    d9:f3:6f:5f:67:c5:1f:d6:be:09:73:50:22:7f:be:
                    90:ab:40:3b:7d:73:31:e6:25:46:ca:67:2d:31:67:
                    3f:07:63:86:2b:b8:29:ca:18:3b:8c:32:93:1f:d4:
                    d0:d4:a0:23:03:fb:b0:de:8e:8a:fc:3b:cd:ea:3a:
                    49:84:ee:bb:eb:9e:fb:d1:5b:c1:13:eb:e4:10:01:
                    4b:55:fd:ba:27:38:ba:7b:e1:8e:9d:11:8b:4c:7f:
                    7d:aa:c1:b4:27:44:a1:ff:bb:c7:91:5a:38:3c:4c:
                    f8:73:73:81:25:90:d5:9b:b2:ca:7a:87:fe:b7:93:
                    c6:3a:5b:88:19:76:bf:2f:1d:10:76:37:e9:02:00:
                    37:0d:2e:b4:ee:c5:f0:eb:9e:63:63:91:4d:86:6e:
                    d5:70:44:9f:fa:cf:0c:50:86:86:30:e6:84:77:0f:
                    70:86:8d:b9:0a:af:78:3a:6f:90:e8:95:06:99:bb:
                    ad:14:55:a0:9a:a0:33:8f:b4:ad:b3:01:e9:a4:74:
                    0e:eb:06:bb:d2:14:c3:54:36:57:c7:49:87:36:2d:
                    ac:bf:68:a7:c4:3f:91:b7:e1:88:55:4d:d0:f5:e1:
                    0c:35
                Exponent: 65537 (0x10001)
    Signature Algorithm: sha1WithRSAEncryption
         2d:91:52:a4:bf:83:ff:ad:bc:1f:f5:85:d5:ed:32:9b:14:ba:
         28:b9:fb:57:de:01:c8:92:59:b3:98:62:5d:e4:6a:06:39:ff:
         dd:2f:d4:fa:d0:01:16:a8:ca:f9:e2:1d:f1:20:6a:5e:7d:73:
         69:3b:ae:03:30:21:d7:2f:3d:b3:e3:6a:a2:f0:e7:3b:26:82:
         2c:7b:9b:e0:2f:79:c5:69:f4:d7:bb:6a:16:8e:62:5c:10:fd:
         ab:2d:e8:e1:88:ee:bc:f4:b4:98:00:55:59:03:b6:5d:73:6e:
         ac:22:cd:09:6b:02:fe:4e:84:ea:f6:a7:f6:c5:eb:97:b5:8a:
         f5:f4:9e:df:32:95:c2:90:2e:82:7c:57:11:f1:ae:91:c2:93:
         fb:b1:81:aa:fa:4e:21:8e:94:93:56:99:cc:53:b3:6d:d1:8f:
         cd:76:b6:1f:a5:b8:d3:88:55:0b:1b:45:3f:28:a2:6a:78:3d:
         ef:54:f0:69:43:e6:35:97:b4:0f:59:c8:05:7f:0f:b7:10:8b:
         db:00:41:2f:ce:60:1a:c4:05:91:d0:19:6f:af:9d:dc:ba:2f:
         5a:48:fc:ea:1a:2a:6b:75:dd:32:8f:f6:b8:ff:16:04:e8:ea:
         91:55:8b:19:f4:1b:30:8e:c4:29:0e:90:9b:19:ef:59:fe:75:
         3f:dd:f9:97
```

-Punto 4:

Al igual que en la solicitud del punto 2, podemos observar campos interesantes como **Subject**, **Public-key**, a diferencia del anterior, en esta solicitud podemos observar el nombre de la curva elíptica de la cual pertenece el archivo que hemos utilizado en el campo **ASN1 OID.**

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer4$ openssl req -in petic-cert-
FranciscoEC.csr -text -noout
Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: C=ES, ST=Granada, L=Granada, O=Internet Widgits Pty Ltd/emailAddress=franfe
rmi@correo.ugr.es
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (192 bit)
                pub:
                    04:2e:5f:a1:37:5b:c0:a6:27:01:18:f7:cf:11:1f:
                    3c:54:05:9e:2f:03:25:2f:20:ac:70:69:93:fe:64:
                    3e:7b:7a:56:fa:52:bf:18:32:70:c1:96:00:bc:18:
                    0a:fc:17:f6
                ASN1 OID: secp192k1
        Attributes:
            a0:00
    Signature Algorithm: ecdsa-with-SHA256
         30:35:02:18:72:a8:07:95:1a:5c:7c:17:0f:a9:70:c3:19:22:
         c5:ef:83:e3:68:c0:86:7f:0b:10:02:19:00:82:af:a3:4a:6b:
         08:1a:37:0a:da:98:ec:b0:6f:b0:98:b7:77:4f:43:4e:1a:6d:
         c1
```

-**Punto 5:** Por último, se muestran los valores del certificado generado a partir de la solicitud. Tenemos los mismos datos del propietario, como dato interesante que se observa puede ser en **Public Key Algorithm** que tenemos **id-ecPublicKey** ya que estamos utilizando los parámetros de una curva elíptica y al igual que en el punto anterior en **ASN1 OID** se especifica el nombre de la curva seleccionada.

```
francisco@Fernandez-Ubuntu:~/Escritorio/SPSI/Prácticas/P4/Ejer5$ openssl x509 -in cert-Franc
iscoEC.crt -text -noout
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number: 13221353228152655976 (0xb77baeb61600d068)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=ES, ST=Granada, L=Granada, O=Internet Widgits Pty Ltd/emailAddress=franfer
mi@correo.ugr.es
        Validity
            Not Before: Dec  7 09:22:32 2017 GMT
            Not After : Jan  6 09:22:32 2018 GMT
        Subject: C=ES, ST=Granada, L=Granada, O=Internet Widgits Pty Ltd/emailAddress=franfe
rmi@correo.ugr.es
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (192 bit)
                pub:
                    04:2e:5f:a1:37:5b:c0:a6:27:01:18:f7:cf:11:1f:
                    3c:54:05:9e:2f:03:25:2f:20:ac:70:69:93:fe:64:
                    3e:7b:7a:56:fa:52:bf:18:32:70:c1:96:00:bc:18:
                    0a:fc:17:f6
                ASN1 OID: secp192k1
    Signature Algorithm: sha1WithRSAEncryption
         94:21:d5:7f:53:b1:7a:57:45:16:8d:34:fb:c8:2e:05:ad:4c:
         b1:14:46:ff:b6:cd:c4:4d:50:06:cb:80:58:a0:b1:6c:e1:f0:
         cc:0c:43:de:cf:cb:70:67:07:47:a4:ad:4e:38:41:d8:aa:83:
         5b:35:5a:a0:3e:83:d1:79:94:be:7f:11:6c:43:98:3c:2e:a5:
         54:e1:c0:19:f6:71:dc:50:54:82:b9:48:4d:ba:8c:fa:35:8f:
         9e:bd:44:26:ff:4f:86:fb:0f:db:22:22:4a:ad:2a:60:4f:e2:
         01:74:22:64:9d:20:5a:f6:f8:f4:be:d4:fb:e7:6f:00:eb:41:
         6f:53:f8:9c:93:c7:02:58:cc:63:ca:48:cf:19:1a:b0:8e:c7:
         dc:fe:4f:6a:93:c3:22:9b:95:b2:5b:69:e9:13:9d:b9:01:23:
         6a:e4:4b:d6:ce:67:e2:5f:6d:ec:6e:4f:15:82:8a:11:a7:94:
         1e:8e:f4:24:f4:35:3f:12:6d:f0:b8:01:03:3c:d6:92:05:23:
         2e:b2:56:11:a2:8d:a9:b3:04:dc:7d:66:3e:77:71:21:0d:ff:
         bf:cd:c8:79:49:1a:6b:4f:e2:46:38:92:30:8c:bf:94:93:36:
         b1:c8:bf:0c:b6:be:ae:f3:9d:8b:18:44:b3:d5:74:67:d6:6b:
         49:81:ed:cb
```