Facultad de Informática de Barcelona

Quatrimestre de otoño, 2022/2023

Cuarta práctica de Criptografía

ECC i certificados digitales

Guillem González Valdivia
(guillem.gonzalez.valdivia@Estudiantat.upc.edu)
Daniel Morón Roces
(daniel.moron.roces@Estudiantat.upc.edu)

Índice

1. Conexión con www.wikipedia.org

(a) Comprobad que el número de puntos (orden) de la curva usada en el certificado es primo.

In [6]:
```
#Datos de la curva p-256 (ecdsa_secp256r1_sha256)
p = 115792089210356248762697446949407573530086143415290314195533631308867097853951
n = 115792089210356248762697446949407573529996955224135760342422259061068512044369

#la a es -3
#la b es propia de la curva

a = -3
b = 0x5ac635d8aa3a93e7b3ebbd55769886bc651d06b0cc53b0f63bce3c3e27d2604b


E = EllipticCurve(Zmod(p),[a,b])
E.cardinality()
E.cardinality().is_prime()
```

Out[6]: True

(b) Comprobad que la clave pública P de www.wikipedia.org es realmente un punto de la curva.

Una vez ya hacemos el cálculo del orden del punto estamos verificando que realmente la clave pública pertenece a la curva. El resultado del orden lo mostramos en el siguiente apartado.

(c) Calculad el orden del punto P.

```
#Punto a partir de la clave pública Q = (Qx, Qy)
x = 0x1ba73b45d7d1948351b92073aef3fb77af348815ae9edbe6a29d98d5d7d3de11
y = 0x65dd7b1fb40ee534c0fba27def07cdfa64ae45522ddd4c4338a169f4606cac09
G = E([x,y])
G.order()
```

Out[7]: 115792089210356248762697446949407573529996955224135760342422259061068512044369

(d) Comprobad que la firma ECDSA es correcta.

A continuación mostramos una captura de los cálculos realizados. Igualmente se pueden comprobar todos los cálculos en el fichero Càlculs ECC.ipynb que adjuntamos en el fichero comprimido de la entrega final.

In [2]:
```
#Datos de la curva p-256 (ecdsa_secp256r1_sha256)
p = 115792089210356248762697446949407573530086143415290314195533631308867097853951
n = 115792089210356248762697446949407573529996955224135760342422259061068512044369

#la a es -3
#la b es propia de la curva

a = -3
b = 0x5ac635d8aa3a93e7b3ebbd55769886bc651d06b0cc53b0f63bce3c3e27d2604b


E = EllipticCurve(Zmod(p),[a,b])

#Punto a partir de la clave pública  Q = (Qx, Qy)
x = 0x1ba73b45d7d1948351b92073aef3fb77af348815ae9edbe6a29d98d5d7d3de11
y = 0x65dd7b1fb40ee534c0fba27def07cdfa64ae45522ddd4c4338a169f4606cac09
G = E([x,y])

#Verificacion de la signatura

#Sustituimos f1 y f2 con los bytes que capturamos del wireshark en el campo de signature y di
f1 = 0x0089f4e1ab93b96fe8ab4fd4b0a9e3c2634f53081bd5951e2cc6125824447e2780
f2 = 0x00a5be0c4d696f3e586dda1784f7b67d0dcaf7fde6234fe835ecf152e04b8e8575

#los 256 primeros bits del shaXXX de la concatenacion de los binarios concatenados
m = 0x3eee401d1546fc38cabdf06191242089189c61b051b8ddd95ec4fb3ac7b219e4




#Punto que viene añadido en el documento NIST

x1 = 0x6b17d1f2e12c4247f8bce6e563a440f277037d812deb33a0f4a13945d898c296
y1 = 0x4fe342e2fe1a7f9b8ee7eb4a7c0f9e162bce33576b315ececbb6406837bf51f5
Punto = E([x1,y1])
#Punto.order()

w1 = mod(m*f2^-1,n)
w2 = mod(f1*f2^-1,n)

verificacion = Integer(w1)*Punto+G*Integer(w2)
mod(verificacion[0],n) == f1
```

Out[2]: True

2. Conexión con www.fib.upc.edu

(a) Obtened el periodo de validez del certificado y la clave pública (módulo y exponente, en base 10 del web de la FIB. ¿Cuántos digitos tiene el módulo?

El periodo de validez del certificado es el siguiente:

- Emitido el lunes, 28 de marzo de 2022, 2:00:00
- Vencimiento el miércoles, 29 de marzo de 2023, 1:59:59

El exponente es 65537 y el tamaño de la clave es de 4096 bits. En cuanto al módulo, tiene 1233 digitos en decimal.

Hexadecimal:

```
9D:B2:C7:CA:34:D4:0C:97:E6:70:AC:3A:98:0C:22:EE:51:67:3F:B1:5A:1B:9E:B1:78:DB:CB:21:7D:60:A7:87:D7:98:62:34:5D:
C3:8B:B8:B1:99:A7:AF:7E:D6:BA:8C:6C:31:3B:55:D0:0F:E9:5C:D6:CE:62:04:6A:54:19:03:34:E7:FC:94:7E:7E:49:EE:E1:E1:
0C:10:00:B4:3E:76:A0:1D:25:49:AC:A3:DB:D6:F4:48:CE:09:C8:74:B8:4F:DE:FE:B6:20:3F:FD:7E:69:D7:34:A7:44:B7:49:2F:
E7:08:6A:1A:B8:1E:8E:C2:3C:F9:E9:76:6C:15:A1:ED:80:F7:64:C3:76:1D:6C:1A:52:37:0F:19:87:99:67:FF:4B:44:93:4F:BD:
21:0A:72:8F:89:72:90:A5:8C:7D:3B:8A:B7:04:08:FE:1A:BB:E5:8C:BA:3C:FA:BB:3E:B4:7B:CD:47:59:9B:A6:21:D3:12:86:94:
24:39:9E:2D:89:C6:94:AA:F7:20:89:91:0B:2C:0A:25:69:F1:09:E1:85:BB:03:B1:29:29:A2:F2:93:5C:40:CC:36:45:D2:42:30:
C5:18:52:1F:C3:8C:C6:ED:E3:88:19:1A:C3:15:EE:1C:A4:DA:E4:F8:B3:7A:34:E1:54:5A:B3:EC:61:D2:E6:3D:0D:97:A6:73:CB:
49:B7:0E:D3:F8:D4:DB:35:3A:5F:0D:97:3F:FE:AF:B7:56:CC:2C:DE:FE:DB:1A:FD:9D:5E:67:C6:2B:F5:13:8A:1A:53:21:0E:1B:
8F:1E:D0:F6:83:6A:01:39:1B:B2:03:1C:F8:01:DF:A5:14:13:4A:30:C5:D3:23:AA:DB:2D:07:7B:1A:2E:C9:4E:6C:69:0D:92:F3:
69:1B:45:8E:68:5B:F7:43:42:3B:C8:98:80:9B:AC:59:3D:AB:1C:73:6A:83:A2:20:31:1F:20:02:16:DF:28:2F:84:4A:72:C3:F1:
23:8F:FD:FC:AA:32:92:1E:C2:18:D9:13:29:B6:A2:CC:5F:AC:CD:EB:A5:48:AE:6A:80:8B:47:06:41:D3:6D:BE:80:B6:5A:7C:86:
3E:39:DA:65:FF:78:C8:3C:53:D6:BD:A2:27:AF:94:DD:CE:C2:05:FD:A9:EB:63:DD:ED:4E:A1:17:8E:4F:57:98:4B:B2:F4:2E:B5:
DD:5C:38:AE:39:94:3A:3D:FF:03:00:15:92:B3:5E:7E:BF:25:A1:E8:D0:D3:13:B4:99:79:80:93:60:E5:42:4B:1E:A8:DF:7B:7A:
B5:B0:55:07:7E:E4:C7:6A:7D:4D:01:E9:6E:D9:76:02:71:8A:DD:58:E3:4E:1E:E7:45:C7:EF:C8:18:A3:77
```

Decimal:

```
643353183200092094287771675345221470010077629570715127404791684658292676146258015596958199149307800248646412074
806402713345405555081170237753105542354684208619132518842723761824119200367033102378675040596360890232050015927
575092514853608500408281492709084825821114563373317007098338129291091372712149614623307473988341465977307145752
705448273512348134128938358986609643424139880228535842593559221687695145242873591558628568048836430658763963628
457559544656711786049228438123628688782475462783972073491969055071501707972940833772753016350946584992156170455
681183858334172706286837128565378248759075309162659580574430676771515129978897420613649359259526062256680143079
870493073398084497931098873418020976977974721325500443350148701944644764873813928657096695060301865446376943751
792046183219540156504462769309116980510428926960863497599865711668561576864719822190421309237150056376914612222
922525939922046810299500313041779504422706733308439869413686737049578366809035296945443953798010938544066680985
820077138377974822115049336106352949077227851726631333176196420428097177276681272065534315670606060953264744935
518704733533552192151247631322510337116144318115750922555584769639254837994316871502659521941029235075453497312
888953021303
```

(b) En el certificado encontraréis un enlace a la polıtica de certificados (SCPS) de la autoridad certificadora firmante. ¿Qué tipo de claves públicas y tamaños admite?

- Claves RSA cuyo tamaño de módulo en bits sea divisible por 8, y sea de al menos 2048 bits. Los certificados de firma de código para usuarios finales tendrán al menos 3072 bits.
- Claves ECDSA en las curvas P-256 o P-384.

Todos los certificados que caduquen el 31 de diciembre de 2030 o antes deberán contener Claves Públicas sujetas de almenos 2048 bits para RSA/DSA, de al menos 256 bits para curva elíptica, y estar firmados con la Clave Privada correspondiente.

Todos los certificados que caduquen después del 31 de diciembre de 2030 deberán contener claves públicas de al menos de al menos 3072 bits para RSA/DSA, de al menos 256 bits para curva elíptica, y firmarse con la clave privada correspondiente.

(c) En el certificado encontraréis un enlace un punto de distribución de la CRL de la autoridad certificadora. ¿Cuántos certificados revocados contiene la CRL?

La CRL contiene 13997 certificados revocados.

(d) En el certificado encontraréis la dirección OCSP (Online Certificate Status Protocol) a la que se puede preguntar por el estatus del certifido. ¿Cuál es el estatus del certificado y hasta cuándo es válido dicho estatus?

El estatus del certificado es válido. La siguiente actualización es hasta el 19 de Diciembre (Dec 19 08:34:04 2022) y concretamente el certificado caduca el 29/03/2023.