University of London International Programmes
CO3326 Computer security 2016–17

Coursework assignment 2

IMPORTANT: This coursework assignment builds on the first coursework and assumes that you understand Elliptic Curve Cryptography (ECC) and how it works. Furthermore, it reuses the curve you were given in Coursework assignment 1 and the point you had to use for modular multiplication. So, please refer to the email you received that contains the unique set of numbers you used for Coursework assignment 1.

The aim of this coursework assignment is to show evidence of understanding of the Elliptic curve Diffie-Hellman (ECDH) key exchange protocol, which allows two parties, each having an elliptic curve public–private key pair, to establish a shared secret over an insecure channel. This shared secret may be directly used as a key, or to derive another key which can then be used to encrypt subsequent communications using a symmetric key cipher. It is a variant of the Diffie-Hellman protocol using elliptic curve cryptography.

Protocol

The following example will illustrate how a key establishment is made. Suppose Alice wants to establish a shared key with Bob, but the only channel available for them may be eavesdropped by a third party. Initially, the domain parameters, which are *k, a, b,* and *G*, must be agreed upon;

a and *b* determine the curve;

k restricts the curve to a prime field $\mathbb{F}_k$;

G is a generator, a point in the field.

Also, each party must have a key pair suitable for elliptic curve cryptography, consisting of a private key *d* (a randomly selected integer in the interval *(1, n-1]*, where *n* is the order of the field) and a public key *Q* (where $Q = dG$, that is, the result of adding *G* together *d* times). Let Alice's key pair be (d_A, Q_A) and Bob's key pair be (d_B, Q_B). Each party must know the other party's public key prior to execution of the protocol.

Alice computes $(x_k, y_k) = d_AQ_B$. Bob computes $(x_k, y_k) = d_BQ_A$. The shared secret is x_k (the *x* coordinate of the point).

Parameters

For the elliptic curve, please use the same *a* and *b* values that you were given for Coursework assignment 1. Furthermore, please use the *P* point that you used in Coursework assignment 1 for the modular multiplication exercise for the generator (*G*) in Coursework assignment 2.

Report

Please write a report using the following skeleton:

1. Show in detail all the steps of the key exchange protocol, with the calculation expanded using the numbers you were given. For the private keys (d_A and d_B) you may choose any number in the *[11, n-1]*, where *n* is the order of the field.
2. How do Alice and Bob arrive at the same shared secret?
3. If Carol is intercepting the communication and captures Q_A and Q_B, can she compute Alice's and Bob's private keys?
4. A more sophisticated attack by Carol involves generating (d_C, Q_C) for use as a reset value, using the same values of *a*, *b*, *k* and *G* that Alice and Bob are using. Explain how this would work.
5. Write a brief discussion (two paragraphs) on the comparison of ECC (Elliptic Curve Cryptography) and RSA, focusing on the advantages and disadvantages of each.
6. Include key snippets of your code. *NOTE:* as you are doing modulo multiplications with figures greater than 10 on an elliptic curve, your work will most probably involve some programming. You may choose a programming language of your liking, whatever you are most comfortable with. The snippet should be the fragment dealing with modular multiplication on the elliptic curve.

Show all your work. You must include in-text citation and provide a detailed references section at the end of your coursework assignment (see How to avoid plagiarism in Study Support on the VLE.) Please note, that there is no need to copy-paste entire explanations or mathematical proofs.

What to submit

The report should be submitted as a PDF document, following a *strict naming scheme*: *StudentName_{srn}_CO3326_cw2.pdf*. For example, if your name is *Mark Zuckerberg* and your SRN is *000000001*, your report submission will be *MarkZuckerberg_000000001_CO3326_cw2.pdf*. Your report will count as **60 %** of your CW2 mark.

In addition to your report, you will also submit a JSON file, which also follows a *strict format* and *naming scheme* and summarizes the results of your calculations. This will count as **40 %** of your CW2 and will be automatically checked by an algorithm, so please pay particular attention to its format.

The name of the file should be *StudentName_{srn}_CO3326_cw2.json*; for example, if your name is *Mark Zuckerberg* and your SRN is *000000001*, your JSON submission will be *MarkZuckerberg_000000001_CO3326_cw2.json*. You can use the following well-formed JSON and adapt it for your numbers and your calculation results.

A hypothetical student, *Mark Zuckerberg*, with SRN *000000001*, received the following numbers:

SRN	Name	a	b	k	Px	Py	Qx	Qy	n
000000001	MARK ZUCKERBERG	-2	13	103	19	97	27	81	3

He would submit the following JSON, which reflects a correct solution:

```
{
  "name": "MARK ZUCKERBERG",
  "srn": "000000001",
  "ecc": {
    "a": -2,
    "b": 13,
    "k": 103,
    "order": 109,
    "g": {
      "x": 19,
      "y": 97
    },
  },
  "assignment": {
    "key_exchange": {
      "alice" : {
        "da" : 13,
        "qa" : {
          "x": 58,
          "y": 37
        }
      },
      "bob" : {
        "db" : 17,
        "qb" : {
          "x": 22,
          "y": 76
        }
      },
      "key": {
        "x": 63,
        "y": 46
      }
    }
  }
}
```

Using his *a* and *b* values, the generic elliptic curve, defined by $y^2 = x^3 + ax + b$, becomes $y^2 = x^3 - 2x + 13$. This is the same curve he was allocated in Coursework assignment 1.

He uses $k = 103$ to restrict the curve onto the $\mathbb{F}_k = \mathbb{F}_{103}$ prime field. He determines the *order* of the field by working out how many points the discrete curve has (including the point at infinity). In this case it is 109 (observe "order": 109 in the JSON result). Then, using his point, $P(19, 97)$ as a generator $g(x, y) = P(x, y) = (19, 97)$ and choosing randomly Alice's and Bob's private key, $d_a = 13$ and $d_b = 17$, he computes the corresponding public keys $Q_a(x, y)$ and $Q_b(x, y)$, and the shared key between Alice and Bob $key(x, y)$. Note the results in the JSON output.

NOTE: Most programming languages have very good support for JSON parsing and output (including Java, C#, Python and JavaScript), so you can even rely on your programme to produce the required JSON output.

[TOTAL 100 %]

[END OF COURSEWORK ASSIGNMENT 2]