Diffie-Hellman Key Exchange with Elliptic Curves

Gianluca Dini
Dept. of Ingegneria dell'Informazione
University of Pisa
Email: gianluca.dini@unipi.it
Version: 30/03/25

1

ECDHKE

THE PROTOCOL

2

Browsers implement this for performance reasons

Domain parameters

UNIVERSITÀ DI PISA

- Choose a prime p
- Choose a curve E: $y^2 \equiv x^3 + a \cdot x + b \bmod p$
- Choose a primitive element P
- Domain parameters: p, a, b, P

mar-25 ECDHKE 3

3

The protocol

UNIVERSITÀ DI PISA

Alice		Bob
choose $privK_A = a \in \{2,3,\ldots,\#E-1\}$		choose $privK_B = b \in \{2,3,\ldots,\#E-1\}$
compute $pubK_A = a \cdot P = A$		compute $pubK_B = b \cdot P = B$
	-------------- A ------------ >	
	< ----------- B --------------	
compute $a \cdot B = T_{AB}$		compute $b \cdot A = T_{AB}$

- T_{AB} is a joint secret between Alice and Bob
- $T_{AB} = (x_{AB}, y_{AB})$ can be used to generate the session key
  - (x_{AB}, y_{AB}) are not independent of each other (There exists a relationship due to curve)
  - E.g., session key AES-$K_{AB} = H(x_{AB})|_{128}$

mar-25 ECDHKE 4

4

We typically use x as coordinate for secret and run it through hash.

The protocol

- The correctness of the protocol is easy to prove.
  - Proof.
    - Alice computes $a \cdot B = a \cdot (b \cdot P)$
    - while Bob computes $b \cdot A = b \cdot (a \cdot P)$.
    - Since point addition is associative (remember that associativity is one of the group properties), both parties compute the same result, namely the point $T_{AB} = a \cdot b \cdot P$
    - Q.E.D.

ECDHKE

SECURITY

Security

- Elliptic Curve Diffie Hellman Problem (ECDHP)
  - Given p, a, b, P, A and B determine $T_{AB} = a \cdot b \cdot P$
- It seems there is only one way to solve ECDHP, namely, to solve ECDLP

$$a = \log_P A$$

or

$$b = \log_P B$$

mar-25 ECDHKE 7

7

Security

- IF (big «if») the curve E is chosen accurately (cryptographically strong) the only viable attacks are *generic* DL algorithms
  - Shank's baby-step giant-step
  - Pollard's rho method

  whose running time is $O\left(\sqrt{\#E}\right)$
- Example: $\#E = 2^{160}$ provides 80 bit of security and requires a p roughly 160 bit long (Hasse's bound)

mar-25 ECDHKE 8

8

Security

- A security level of 80 bit provides medium term security
- Normally a security level of 128 bit is required thus we need to use curves #E = 256 → 2^{256}
  whose cardinality is 256.
- Standardised EC
  - NIST: Elliptic Curve Cryptography
    - FIPS 186-4 (July 2013) – 15 different curves
    - FIPS 186-5 (in progress)
  - Should we trust the NIST-recommended ECC parameters?

mar-25 ECDHKE 9

9