Scribe: Cryptography and Network Security (Week 10 - Class 5)

Rashil Gandhi

07-Nov-2020

1 Introduction

This class discusses the Discrete Logarithm Problem (DLP) and the ElGamal Cryptosystem, which uses DLP as a defence against brute force attacks and how it follows the principles of semantic security. Some algorithms for breaking the DLP are also discussed. After this, the focus shifts to the two instances of the Diffie-Hellman Key Exchange scheme. Lastly, a brief introduction to Elliptic Curve cryptography is given.

2 The Discrete Logarithm

Consider a finite mathematical group $(G, .)$. For an element $\alpha \in G$ having order n, let

$$\langle\alpha\rangle = \{\alpha^i : 0 \leq i \leq n-1\}$$

The discrete logarithm problem is to find the unique integer i, called the discrete log, such that

$$\alpha^i = \beta$$

It can be seen that the discrete logarithm operation is the inverse of exponentiation operation. While we have efficient algorithms in place for computing exponents, if the group is chosen properly, the discrete logarithm operation is believed to be difficult. This paves way for its applications in cryptography as exponentiation is a potential one-way operation.

3 The ElGamal Cryptosystem

The cryptosystem is defined as follows. Let p be a prime such that DLP in $(Z_p^*, .)$ is hard. Let $\alpha \in Z_p^*$ be a primitive element. Define the plaintext as $P = Z_p^*$ and the ciphertext as $C = Z_p^* \times Z_p^*$. The key is $K = \{(p, \alpha, a, \beta) : \alpha^a = \beta\, mod\, p\}$. For a given key k and for a secret random number $r \in Z_{p-1}$, define

$$e_k(x, r) = (y_1, y_2)$$

where

$$y_1 = \alpha^r \, mod \, p$$

and

$$y_2 = x\beta^r \, mod \, p$$

For $y_1, y_2 \in Z_p^*$, define:

$$d_k(y_1, y_2) = y_2(y_1^a)^{-1}$$

The ciphertext depends on both the plaintext and the randomly chosen value r, making the encryption non-deterministic. This is essential for achieving semantic security, as determinism in cryptosystem can lead to leak of information about the plaintext from the ciphertext. If a system is semantically secure, an adversary, given two plaintexts of equal length and their two respective ciphertexts, cannot determine which ciphertext belongs to which plaintext.

4 Breaking the DLP

4.1 Exhaustive Search

The modular exponentiation operation does not have a monotonic nature; there is no ordering of the powers. As such a linear exahustive search for all values of i might be needed in the worst case. This takes $O(n)$ time and $O(1)$ space.

4.2 Binary Search

If we somehow store all possible values of $\alpha^i \, mod \, p$ as ordered pairs $(i, \alpha^i \, mod \, p)$, we can employ binary search to find i. Good searching algorithms run in $O(logn)$ which can be approximated to $O(1)$ for our use. Thus this method takes $O(n)$ space and $O(1)$ time. This is termed as time-memory trade-off when compared to the previous method.

4.3 Shanks' Algorithm

In a group G, we want to solve for i in the equation $\alpha^i = \beta$ where i is an unknown integer. If g is an element of order $N \geq 2$, let $m = \lceil \sqrt{n} \rceil$. In the two sets:

$$\{e, g, g^2, ..., g^m\}$$

and

$$\{hg, hg^{-m}, hg^{-2m}, ..., hg^{-m^2}\}$$

there will be a common element if the DLP has a solution. If the match is $g^s = hg^{-tm}$, then the solution to the DLP is $i = s + tm$. This algorithm runs in $O(m)$ time with $O(m)$ memory (neglecting logarithmic factors).

5 Diffie-Hellman Key Exchange

Let g and p be two quantities under public domain. The Diffie-Hellman Key Exchange between two entities Alice and Bob happens like this:

Both Alice and Bob have their own private quantities, a and b respectively. Alice sends the value of the expression $g^a \, mod \, p$ to Bob. Bob, upon receiving it, further calculates the expression $(g^a \, mod \, p)^b \, mod \, p = g^{ab} \, mod \, p = k$ using his private quantity b, to arrive at the symmetric key k. Alice will do a similar exercise when Bob sends his expression $g^b \, mod \, p$, i.e., calculate $(g^b \, mod \, p)^a \, mod \, p = g^{ab} \, mod \, p = k$.

This technique, while relying on the hardness of DLP, has an inherent flaw. It is susceptible to man-in-the-middle attack. In this attack, an opponent Carol intercepts Alice's public value and sends her own public value to Bob. When Bob transmits his public value, Carol substitutes it with her own and sends it to Alice. Carol and Alice thus agree on one shared key and Carol and Bob agree on another shared key. After this exchange, Carol simply decrypts any messages sent out by Alice or Bob, and then reads and possibly modifies them before re-encrypting with the appropriate key and transmitting them to the other party. This vulnerability is present because Diffie-Hellman key exchange does not authenticate the participants.

The Diffie-Hellman problem can also be stated in two instances.

Computational Diffie-Hellman Problem (CDH): Given α^b and α^c, find α^{bc}.

Decision Diffie-Hellman Problem (DDH): Given α^b, α^c and α^d, determine if $d \equiv bc \, mod \, n$.

In terms of difficulty, we have DDH << CDH << DLP. Hence, the hardness of DDH is the strongest assumption among the three.

6 Elliptic Curve Cryptography

Elliptic-curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. ECC allows smaller keys compared to non-EC cryptography based on plain Galois fields to provide equivalent security. The logarithm problem in elliptic-curve finite fileds is called Elliptic Curve Discrete Logarithm Problem (ECDLP). ECDLP is considered to be more difficult than DLP. An elliptic curve is a plane curve over a finite field (rather than the real numbers) which consists of the points satisfying the equation

$$y^2 = x^3 + ax + b$$

along with a distinguished point at infinity, denoted ∞. The ECDLP requires solving the following equation for n

$$Q = nP$$

where the scalar multiplication is defined in terms of the elliptic-curve repeated addition operation.

References

blog.jpolak.org/?p=1963

en.wikipedia.org/wiki/Elliptic-curve_cryptography

en.wikipedia.org/wiki/Diffie-Hellman_key_exchange