Week 8

Lecture 1

Key Management (Public Key)

Lecture 2

Finite Fields and ElGamal Encryption

Workshop 8: Workshop based on Lectures in Week 7

Quiz 8

Udaya Parampalli

ElGamal Encryption
(Public Key)

COMP90043
Lecture 1

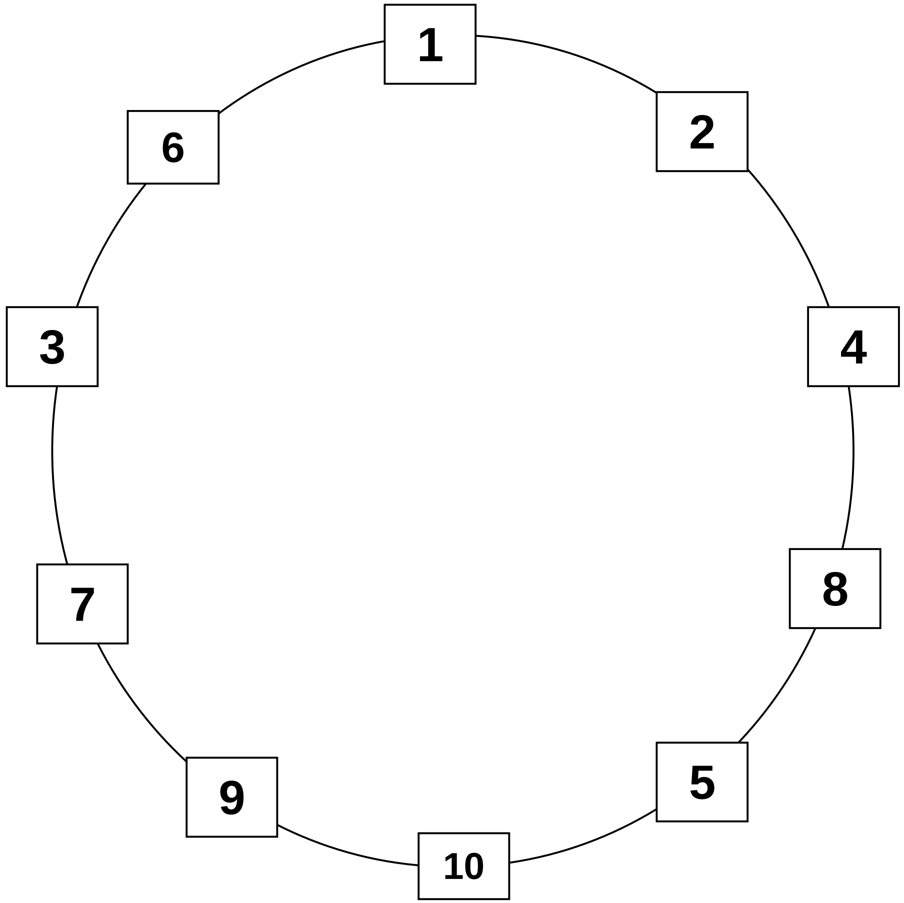
Udaya Parampalli

Lecture 2

1.1 ElGamal Encryption

- DH Protocol to Encryption
- Basic Ideas
- Example
- Security Properties.

Rcap: DLOG: An example



g^i	$g^i \bmod p$	$Dlog(g^i)$
2^1	2	1
2^2	4	2
2^3	8	3
2^4	5	4
2^5	10	5
2^6	9	6
2^7	7	7
2^8	3	8
2^9	6	9
2^{10}	1	10

Example of a Cyclic group modulo $p = 11$

g : generator = 2

Order(size) of G = 10

What power of 2 is 3?

Recap: Example mod 11

X	2^x mod 11	3^x mod 11
0	1	1
1	2	3
2	4	9
3	8	5
4	5	4
5	10 Or -1	1
6	9	3
7	7	9
8	3	5
9	6	4
10	1	1
11	2	3

- 2 is a primitive element.
- 3 is not a primitive element
- Given any power of 2, the exponent can be obtained from reading the corresponding index in the table
- In practice a large modulus is used and hence finding the exponent is difficult. This is one of the important one way functions used in modern cryptography.
- In general finding primitive element is also an interesting problem. We use the groups where we can easily find generating elements.

Recap: Diffie-Hellman Protocol

Public Parameters: g: generator, order of the cyclic group: (p-1), prime: p

- Alice — Bob
- Choose Na=2 — Choose Nb=6
- $g^{Na} = 2^2 = 4 = Ma$
-
- $g^{Nb} = 2^6 = 9 = Mb$
- Compute
- $K_{ab} = Mb^{Na}$
- $= 9^2 = 4$
- Compute
- $K_{ba} = Ma^{Nb} = 4^6 = 4$
- $K_{ab} = K_{ba} = 4$

ElGamal Cryptosystem

- Discovered by ElGamal in 1985.
- It directly uses Diffie-Hellman (DH) Protocol.

- What are the hard problems on which DH Protocol relies?
  - Discrete Logarithm Problem (DLP)
  - Computational DH Problem.
- The goal here is to motivate how ElGamal came up with the scheme, nearly after eight years of the discovery of DH protocol.

- What are the key features of this algorithm?

Key Features

- DH protocol can be formulated over any cyclic group where computing discrete logarithm over the group is hard.

- What is the main objective?
  - Two users connected over insecure channel arrive at a common secret by using only public parameters.
  - In our case, they arrive at $g^{(ab)}$, g is a generator of the group; a, b are random secrets chosen by the participants respectively.

Cyclic Groups

- Z_n: Integers modulo n, n is a positive integer, under multiplication
- Z_p: Integer modulo p, p is a prime number, under multiplication
- Residues of Polynomials over Z_p.
- Elliptic Curves over Z_p.
- Some examples of cyclic groups present inside a bigger groups:
- C_8 : {2,4,8,16,15,13,9,1} of order 8, 2 is the generator, operation mod 17,
- C_{30} : {2,4,8,16,1} of order 5, 2 is the generator, operation mod 31,

Order of Cyclic Groups

- What is the maximum size of cyclic groups obtained from $Z_{p,}$, p, a prime number?
- (p-1)
- What is the maximum size of cyclic groups obtained from Z_n?
- $\phi(n)$ = Numbers of integers < n but relatively prime to n.
- What is the maximum size of cyclic groups obtained from $Z_p[x]$ mod m(x), deg(m(x)) =k?
- P^k-1.
- In fact, we can have groups whose size divides the sizes mentioned above.

A variation of Diffie-Hellman protocol

- I will use the notations as in the textbook, so that it can in your study.
- Let us now assume that one of the users in the DH protocol is fixed in advance. Assume computations mod q, q is a prime. “a”: generator of the group.
- Alice generates the key in advance
  - chooses a secret key (number): $1 < x_A < q-1$
  - compute her **public key**: $y_A = a^{x_A} \bmod q$
-
- Bob knows this public key in advance.

A variation of DH

- Bob
  - Choose a random k and compute $a^k \bmod q$
  - **Send** $a^k \bmod q$ **to Alice**
  - Since y_A is available, compute the DH common
  - key $y_A^k = a^{k\, x_A}$
  - Hide the message in the common key and send it to Alice
  - Bob to Alice: $C = M\, a^{k\, x_A}$
- Alice knows her secret x_A
- Obtain the common key in the cipher $(a^k)^{x_A} = a^{k\, x_A}$
- Recover Message $M = C / a^{k\, x_A}$

The scheme ElGamal Cryptography

- Now we give the actual ElGamal Public-key cryptosystem.
- Main tool is exponentiation in a cyclic group where the DLOG is hard.
- As you will see, the security is directly related to difficulty of computing discrete logarithms.
- As before, each user (eg. Alice) generates their key
  - chooses a secret key (number): $1 < x_A < q-1$
  - compute their **public key**: $y_A = a^{x_A} \bmod q$
- NOTE: a is the generator here.

Encryption and Decryption

- Another user (eg Bob) can encrypt a message to send to A by computing the steps below:
  - Represent message M in range $0 <= M <= q-1$
  - Chose random integer k with $1 <= k <= q-1$
  - Compute one-time key $K = y_A^k \bmod q$
  - Encrypt M as a pair of integers (C_1, C_2) where
    - $C_1 = a^k \bmod q$; $C_2 = KM \bmod q$
- Alice can then perform decryption as follows:
  - Recovering the key K as $K = C_1^{xA} \bmod q$
  - Compute M as $M = C_2 K^{-1} \bmod q$
- Note that k needs to be changed for every new message, they need to be unique, why?
- You will see in next slide, the consequences.

Consequences

- Let (M_1, $C_1 = [C_{11}, C_{12}]$) and
- (M_2, $C_2 = [C_{21}, C_{22}]$) be two message and ciphertext pairs using the same randomization parameter k.

- What does this imply for C_1 and C_2 ?
- $C_{11} = a^k \bmod q = C_{21} = a^k \bmod q$

- Notice that $C_{11} = C_{21}$

- So, if Adversary knows M_1, he can then recover M_2 thus, the scheme is insecure. Hence k should be unique for all encryptions. In general a random source is required to create unique k.

Week 8

Lecture 1

Key Management (Public Key)

Lecture 2

Finite Fields and ElGamal Encryption

Workshop 8: Workshop based on Lectures in Week 7

Quiz 8