CSIT 495/595 - Introduction to Cryptography
ElGamal Encryption

Bharath K. Samanthula
Department of Computer Science
Montclair State University

ElGamal Encryption: Introduction

- Introduced by *Taher El Gamal* based on the Diffie-Hellman key-exchange protocol (1985)
- Security is based on discrete logarithm and Decisional Diffie-Hellman (DDH) assumptions
- The ciphertext size is twice that of original message (Note: this is not the case in RSA)
- Uses different randomization in each encryption - each message has many different possible ciphertexts

Recap: Diffie-Hellman Key-Exchange

Fix a finite cyclic group G (e.g $G = (Z_p)^*$) of order n

Fix a generator g in G (i.e. $G = \{1, g, g^2, g^3, \ldots, g^{n-1}\}$)

Alice | **Bob**

choose random **a** in $\{1,\ldots,n\}$ | choose random **b** in $\{1,\ldots,n\}$

$$\xrightarrow{A = g^a}$$

$$\xleftarrow{B = g^b}$$

$$\mathbf{B^a} = (g^b)^a = \boxed{\mathbf{k_{AB} = g^{ab}}} = (g^a)^b = \mathbf{A^b}$$

Diffie-Hellman $\rightarrow$ Public-key encryption

- Imagine that Bob uses the shared (key) value to encrypt a message m
- That is, Bob sends $k \cdot m$ to Alice
- Alice recovers m from $k \cdot m$ using her knowledge of k

ElGamal: Key Generation

- Suppose Alice wants to generate public and private keys based on ElGamal
- Public key: (p, g, A), where $\mathbb{Z}_p$ represents a cyclic group of order q and g is the generator
  - $A = g^a \bmod p$ and a is randomly chosen in $\mathbb{Z}_q$ by Alice
- Private Key: a

ElGamal: Encryption + Decryption

- Suppose Bob wants to send a message m to Alice
- Note that Bob knows Alice public key (p, g, A)
- Encryption by Bob:
  - Choose a uniform $b \in \mathbb{Z}_q$
  - Compute $c_1 = g^b \bmod p$ and $c_2 = A^b \cdot m \bmod p$
  - Send ciphertext $\langle c_1, c_2 \rangle$ to Alice
- Decryption by Alice:
  - Using private key a, compute $\hat{m} = \frac{c_2}{c_1^a}$

Why ElGamal Encryption Scheme Works

Expand decryption process:

- $$\begin{aligned} \hat{m} &= \frac{c_2}{c_1^a} \\ &= \frac{A^b \cdot m}{(g^b)^a} \\ &= \frac{(g^a)^b \cdot m}{(g^b)^a} \\ &= \frac{g^{ab} \cdot m}{g^{ab}} \\ &= m \end{aligned}$$

ElGamal Encryption Scheme: Example (1)

- Let Alice choose G with prime $p = 107$, $g = 2$ and $a = 67$
- Alice compute $A = g^a = 2^{67} \bmod 107 = 94$
- Alice Public key: $(107, 2, 94)$
- Alice private key: 67

ElGamal Encryption Scheme: Example (2)

- Suppose Bob wants to send message "66" to Alice
- Say Bob chooses a random integer $b = 45$
- Encryption by Bob:
  $c_1 = g^b \bmod 107 = 2^{45} \bmod 107 = 28$
  $c_2 = A^b \bmod 107 = 94^{45} \bmod 107 = 9$
- Decryption by Alice:
  Compute $\left(c_1^a\right)^{-1} \cdot c_2 \bmod 107 = \left(28^{67}\right)^{-1} \cdot 9 \bmod 107 = 66$

Analysis of ElGamal (1)

- a (chosen at random) must be kept secret by Alice
- b is a random integer:
  - $c_1 = g^b \bmod p$ remains a random integer
  - A^b is also a random integer mod p
  - Therefore, $c_2 = A^b \cdot m \bmod p$ is the message m multiplied by a random integer
- What happens if b is know to the attacker?

Analysis of ElGamal (2)

Sender must use different b values while encrypting each message (even when encrypting the same message at different times)

- Suppose Bob uses same b for encrypting two messages m_1 and m_2
- In this case, Bob sends $\langle g^b, A^b \cdot m_1 \rangle$ and $\langle g^b, A^b \cdot m_2 \rangle$ for m_1 and m_2, resp.
- This reveals lot to information to the attacker listening on the communication channel. For example,
  - $\frac{m_1}{m_2}$ is known to the attacker
  - Further, if the attacker finds out m_1, he can also determine m_2

Overhead of ElGamal

- Encryption: Two exponentiations; preprocessing possible
- Decryption: one exponentiation
- Message expansion: ciphertext is twice the length of plaintext

Semantic Security of ElGamal

- Note that the generic ElGamal encryption scheme is not semantically secure.
- We can infer whether a ciphertext is quadratic residue or not.
- We can use the above fact to come up with two message where one of them is a quadratic residue and the other one is a quadratic non-residue so that attacker has high advantage in distinguishing encryptions.

- The above issue can be addressed if every plaintext is qudratic residue and $p = 2q + 1$ where q is prime
  - It can be shown that this version is semantically secure if DL is infeasible

Useful References

- Chapter 11, Introduction to Modern Cryptography by Jonathan Katz and Yehuda Lindell, 2nd Edition, CRC Press, 2015.
- http://cacr.uwaterloo.ca/hac/about/chap8.pdf
- http://caislab.kaist.ac.kr/lecture/2010/spring/cs548/basic/B02.pdf