El Gamel

Public Key Cryptosystem

The Discrete Log Problem

The El Gamel public key cryptosystem is based upon the difficulty of solving the ***discrete logarithm problem*** (DLP) which is as follows:

Given a prime p and values g and y, find x such that

$$y = g^x \bmod p$$

- For a small value of p, it is easy to solve a DLP by trial and error or exhaustive search.

- For example, given $p = 11$, $g = 2$ and $y = 9$, we can try different values of x until we reach the correct solution for $2^x \ mod\ 11 = 9$

- However, for a large value of p, i.e., if p has around 100 decimal digits, then it is not possible to solve a DLP using current technology.

El Gamel

- El Gamel is a public key cryptosystem with security which relies on the difficulty of solving a discrete log problem.

- If you can solve the DLP then you can crack El Gamel.

El Gamel Key Generation

Bob generates public and private keys as follows:

1. He picks a large random prime p
2. He finds a generator $g \ mod \ p$

   (i.e. $g^x \ mod \ p$ gives a different answer for every value of x, which means that

   $g^{p-1} \ mod \ p$ is the **first** time the answer is 1).

3. He picks a random number *a* between *1* and *p-1*.
4. He computes $y = g^a \bmod p$

The **public key** is *(p, g, y)*

The **private key** is *a*

El Gamel Encryption

If Alice wants to send Bob a message, she looks up Bob's public key *(p, g, y)* and breaks the message up into blocks with each block less than *p*. Then for each message block *m* she takes the following steps:

1. She generates a random number *k* between *1* and *p-1*.
2. She computes $r = g^k \bmod p$

   $x = y^k \bmod p$

   $c = (m * x) \bmod p$

3. She sends Bob the values *r* and *c*.

El Gamel Decryption

Bob receives the ciphertext *(r,c)* from Alice. He decrypts it as follows:

1. He computes $r^a \bmod p = x$

   $\{r^a = (g^k)^a = g^{ka} = g^{ak} = (g^a)^k = y^k = x\}$

2. Now he can solve $c = (m * x) \bmod p$ to find the value of *m*.

   Only Bob can do this because only Bob knows the value of the private key *a*.

Comparing RSA and El Gamel

RSA

- Security based on the difficulty of the factorisation problem.
- The ciphertext is just one value *c* which is roughly the same size as the message *m*.

El Gamel

- Security based on the difficulty of the discrete log problem.
- The ciphertext is two values *c* and *r* and so is twice the size of the message *m*.

RSA	El Gamel
• The encryption and decryption algorithms are the same (modular exponentiation).	• The encryption and decryption algorithms are different (although both take about the same time to perform).
• RSA is a patented algorithm.	• El Gamel has no patent. This gives it a financial advantage over RSA.

The Elliptic Curve DLP

- A further advantage of the El Gamel cryptosystem, is that it can be generalised to a cryptosystem based on the ***discrete log problem for elliptic curves***.
- This appears to be even harder to solve which means that the prime used can be smaller and so encryption is faster.

Applications of public key cryptosystems.

Public key cryptosystems are generally less efficient but more secure than symmetric key cryptosystems. In practise public key systems are used for

- Digital signatures
- Key exchange

Digital Signatures

- A digital signature for a message from a particular sender is a cryptographic value which depends upon both the message and the sender.
- A digital signature provides ***data integrity*** (proof that the message hasn't been altered) and ***non-repudiation*** (proof of origin - the sender cannot deny sending the message).

- To digitally sign a message *m* using a public key cryptosystem, Bob simply encrypts the message *m* using his **private key** to get a signature *s*.
- Bob then sends Alice both the message and the signature *(m,s)*.
- Alice decrypts the signature using Bob's **public key**. If the decrypted signature is the same as the message *m* then Alice accepts the message is genuine and from Bob.

Digital Signature using RSA

1. Bob takes a message *m* and uses his private key *d* to compute the signature $s = m^d \bmod n$
2. Bob sends Alice the pair *(m,s)*
3. Alice decrypts the signature *s* using Bob's public key *e* to compute $m_1 = s^e \bmod n$
4. If $m_1 = m$ then Alice accepts the message as genuine because only Bob knows the private key *d* which works in conjunction with the public key *(n,e)*.

DSS

Recall that in the El Gamel public key cryptosystem, the algorithms for encryption and decryption are not the same. This means that using El Gamel for digital signatures is not as straight forward as using RSA. However, the Digital Signature Standard (DSS) is an alternative signature scheme to RSA which is based upon El Gamel.