Public Key Cryptography

Rushil Saha

Euler Circle

Public key cryptography is a system in which there are two keys: a public key which is available to everyone and a private key which is only known to a select few. This has two functions, authentication to verify that the sender of the message has the private key and encryption, which can be accomplished with the public key. One example of this is the RSA cryptosystem. This system utilizes the same public and private key setup. It was one of the first public key cryptosystems and continues to be used on a large scale for secure data transmission.

Definition 1. *RSA Cryptosystem The RSA cryptosystem is designed as follows. We will let Alice be the sender of the encrypted message and Bob the recepient. Then, Bob will choose two private primes p and q and let $n = pq$ be public. Then let $e \in Z$ be positive so that $\gcd(e, \phi(n)) = 1$ which is the public key. Then he finds the private key $d \in Z$ such that $de \equiv 1 \pmod{\phi(n)}$. This is finding the inverse of e $\pmod{\phi(n)}$. Then, if Alice wants to encrypt some number $m < n$, she can use the function $c = m^e \pmod{n}$. If Bob want to decrypt it, he can find $c^d \pmod{n}$. The proof can be split into two different cases, depending on if n and a are coprime. If they are, then because of Euler's theorem which states that if n and a are coprime positive integers, then $a^{\phi(n)} \equiv 1 \pmod{n}$. This can be applied to the RSA algorithm. Because $ed \equiv 1 \pmod{\phi(n)}$, then $ed = 1 + b\phi(n)$ for some b. Plugging that back in, we have that $m^{ed} \equiv m^{1+b\phi(n)} \equiv m^1 m^{b\phi(n)} \equiv m \pmod{n}$. If n and a are not comprime, then if $k \equiv 1 \pmod{\phi(n)}$, $a^k \equiv a \pmod{n}$, a slight variation of Euler's Theorem. Then because $ed \equiv 1 \pmod{\phi(n)}$, $m^{ed} \equiv m \pmod{n}$ concluding the proof.*

Example We choose two distinct primes $p = 59$ and $q = 53$. Then $n = pq = 59(53) = 3127$. $\phi(n) = (p-1)(q-1) = 3016$. We can select any e such that $\gcd(e, \phi(n)) = 1$ so we let $e = 3$. We find that the inverse of $e \pmod{\phi(n)} = 2011$. Now we have the public and private keys. The public key is $e = 3$ and the private key is $d = 2011$. Now that we have the keys, we can encrypt a phrase. We can encrypt HI. We can convert the letters to numbers based on lexicographical order. This would mean that $H = 8$ and $I = 9$. Then we can add a 0 between the values of the letters so that the phrases remain unique. We are then encrypting 809.Then the crypted data is $c = 809^3 \pmod{3127}$. The encrypted data is then equal to 2108. Now we can assume the role of the decrypter. The decrypted data is equal to $2108^{2011} \pmod{3127}$. This is equal to 809 which is the initial value that we plugged in.

The RSA algorithm can be broken in several ways. If the same encryption key e is chosen for e or more recipients with different p and q, then a system of equations can be set up and because of the Chinese Remainder Theorem, a solution is then guaranteed.

Example One of the common e values in the past has been 3. If the same message is sent to 3 or more recepients, it can be found with the help of the Chinese Remainder Theorem. We can let m be the message and c_n be the corresponding ciphertext for a $modulo_n$. This results in the following system of equations.

$$\begin{cases} c_1 \equiv m^3 \pmod{n_1} \\ c_2 \equiv m^3 \pmod{n_2} \\ c_3 \equiv m^3 \pmod{n_3} \end{cases}$$

With the Chinese Remainder Theorem, we find that $x = m^3 \pmod{n_1 * n_2 * n_3}$. If m is less than n_1,n_2, and n_3, then a cube root can be taken without regards to modulus yielding the contents of the message. However, if this is not true, it becomes more difficult but a solution can still be found. This is why larger e values are used such as $2^{16} + 1$.

A hacker can also use the fact that the RSA cryptosystem is multiplicative.

Definition 2. *An Attack on RSA The process for finding the original message using the fact that the RSA cryptosystem is mutliplicative is as follows. Being multiplicative in this case means that $m_1^e m_2^e \equiv (m_1 m_2)^e \pmod{n}$. This can be used in the following way. If a hacker is given a ciphertext $c \equiv m^e \pmod{n}$, then the hacker could ask the holder of the private key to decrypt another unsuspicious looking key $c' \equiv cr^e \pmod{n}$ for a r that is chosen specifically by the hacker. Then, because of the multiplicative property, c' is the encryption of $mr \pmod{n}$. This attack will result in the hacker knowing $mr \pmod{n}$ from which he can find the original message by multiplying mr by the inverse of $r \pmod{n}$.*

The soundness of the RSA cryptosystem is based on several key factors, the difficulty of factoring very large numbers and finding the ath root modulo n. The Diffie-Hellman key exchange is another example of public key cryptography. Its primary purpose is the exchange of private keys over a public network. Later, these keys can be used to encrypt and decrypt messages. It works in the following way.

Definition 3. *Diffie-Hellman key exchange Both people agree on some modulus which is a prime p and a primitive base in that modulus g. One person chooses a secret number a and sends the other $A = g^a \pmod{p}$. Another person chooses a secret number b and then sends the other $B = g^b \pmod{p}$ in the same exact method. The first person will compute $B^a \pmod{p}$ and the second person will compute $A^b \pmod{p}$. Both of these numbers will be the same. This is true because $A^b = g^{ab} = g^{ba} = B^a \pmod{p}$. However, for smaller modulo, it is easy to find a and b since there are only so many cases. That is why it is necessary to select a large prime to make it secure.*

Example Alice and Bob agree to use a modulus $p = 23$ and base $g = 5$. Then Alice chooses a secret integer $a = 4$ and sends Bob $A = g^a \pmod{p}$ or 4. Then Bob chooses a secret integer $b = 3$ and sends Alice $B = g^b \pmod{p}$ or 10. Alice computes $s = B^a \mod (p)$ or 18 and Bob computes $s = A^b \pmod{p}$ or 18. Now Alice and Bob share the secret 18.

However, for smaller modulo, this is easy since there are only so many cases. That is why it is necessary to select a large prime to make this a secure cryptosystem. Just to put it into perspective, if p is a prime of at least 600 digits, then the best computers today would not be able to find what a is equal to given g, p, and $g^a \pmod{p}$. g does not need to be a large number and small numbers most generally suffice. This process can be extended to more than two parties in the process as follows.

Definition 4. *Extension for RSA to more than two people The people select the parameters p and g together. Then each person will choose their own private keys. In this case, we can follow the process if there are 3 parties. Each person will choose their private keys called a, b, and c. Alice will compute* $g^a \pmod{p}$ *and sends it to Bob. Then Bob computes* $(g^a)^b = g^{ab} \pmod{p}$ *and sends it to Carol. Carol computes* $(g^{ab})^c = g^{abc} \pmod{p}$ *and keeps it secure. Bob will then compute* $g^b \pmod{p}$ *and send it to Carol. Carol computes* $(g^b)^c = g^{bc} \pmod{p}$ *and sends it to Alice. Alice computes* $(g^{bc})^c = g^{abc} \pmod{p}$*. Carol computes* $g^c \pmod{p}$ *and sends it to Alice. Alice computes* $(g^c)^a = g^{ca} \pmod{p}$ *and send it to Bob. Bob computes* $(g^{ca})^b = g^{abc} \pmod{p}$ *and keeps it secure. After all of this has been transferred, Alice, Bob, and Carol will all have the secret number* $g^{abc} \pmod{p}$*. Even if there is someone trying to intercept the messages, if they just have* $g^a \pmod{p}$*,* $g^b \pmod{p}$*,* $g^c \pmod{p}$*,* $g^{ab} \pmod{p}$*,*$g^{ac} \pmod{p}$*, and* $g^{bc} \pmod{p}$*, it is not possible to combine them to find* $g^{abc} \pmod{p}$*.*

There are a variety of different public key cryptosystems and transfer methods but ultimately, many of them rely heavily on number theory, especially modular arithmetic and famous results such as Euler's Theorem.

References

1. DI Management, `https://www.di-mgt.com.au/crt.html`. Last accessed 7 December 2018
2. Brown Mathematics, `https://www.math.brown.edu/~jhs/MathCrypto/Sampleections.pdf`. accessed 7 December 2018
3. Brilliant Homepage, `brilliant.com`. Last accessed 7 December 2018
4. Stanford Applied Crypto Group, `https://crypto.stanford.edu/`. Last accessed 7 December 2018