Q’loQ

Q’loQ

A Public Key Encryption Algorithm

Based on RSA

- RSA encryption algorithm invented in 1977 by Ron Rivest, Adi Shamir and Leonard Adleman
- RSA is the most popular public key encryption algorithm today
- RSA and Q’loQ encrypt and sign the same way

Q’loQ

- Invented in 2019
- Reference implementation in Python
- Based on the idea of the Klingon cloaking device

Governing Principles

- Base
- Cloak
- Key generation

Cloak

- Establish the cloaking parameters C, K, G, U and V
- C is derived from P modulo Q
- K is derived from Q modulo P
- G is derived from (P modulo Q) + Q

Base

- First, generate 2 primes of equal size and let them be P and Q and let them not be equal
- Second, generate 2 primes of equal size and let them be A and B
- Establish a totient with the product of P -1 * Q – 1 * P * A – 1 * B - 1

Cloak

- We establish the modulus as the product of A and B
- We establish the cloaking modulus as the product of K and G

Cloak

- U is derived from the product of K and G
- V is derived from the following equation

$$((C+K) / K) + (((P/Q) + (Q/P)) / (K+C)$$

Public Key Generation

- Next find a number between 1 and the totient T where the number and T are coprime and call it PK. This becomes the public key.

Private Key Generation

- Find the multiplicative inverse of the public key PK and the totient T and call it SK, the secret key.

Encryption/Decryption

- Encryption is achieved by taking the plain text and raising it to the power of the public key modulo N
- Decryption is achieved by taking the cipher text and raising it to the power of the private key modulo N

Cryptanalysis

- One solves Q'loQ ciphers by finding A and B and the cloaked primes, recontructing the totient and finding the inverse of the totient and the public key
- In RSA one can normally take the modulus modulo some number and when P or Q is encountered a zero should be the result. Q'loQ's cloak defies this and P and Q against the modulus will result in an arbitrary number. One has to use other means to solve the cloaked modulus.