Quantum Algorithms for Attacking Public-Key Cryptosystems

Stephen R. Harrigan

Department of Mathematics and Statistics, University of Ottawa, Ontario, Canada

Introduction

Public-key cryptosystems are the basis for virtually all secure communications. The chief of these cryptosystems, RSA, is believed to be intractable to attack using classical computation, making it safe against attacks by adversaries using even the most powerful classical computers today.

Background

To describe the RSA cryptosystem, we must first present a few definitions:

A *trapdoor function* is a function for which computing the output given an input is easy, but to find the input given an output is hard, unless extra (secret) information is given.

e.g. Computing 7×11 is relatively easy, but find the factors of 91 is fairly difficult (Try it!). However, knowing one of the factors is 7, finding the other is easy.

A public-key cryptosystem is one which uses 2 keys, one private and one public, to encrypt information. The *public key* is made available to anyone by the recipient, whereas the *private key* is known only to the recipient.

To encrypt a message, the sender, usually referred to as Bob, must apply the public key to the message. To decrypt the message, the intended recipient, usually referred to as Alice, applies the private key. Thus, only Alice can decrypt the message. An example is given in Figure 1.

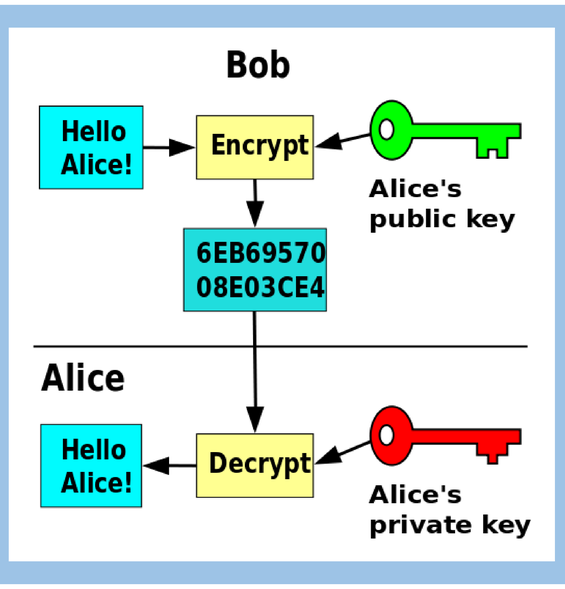

Figure 1: Example of a public key cryptosystem interaction. Bob encrypts a message using Alice's public key. Alice then decrypts the message using her private key.

When a and b have the same remainder upon division by n, we write

$$a \equiv b \pmod{n}.$$

This is known is *modular arithmetic.* Modular arithmetic obeys some nice properties, including Euler's Theorem. If a is coprime to N, then

$$a^{\Phi(N)+1} \equiv a \pmod{N}$$

where $\Phi(N)$ is the number of integers coprime to N. For a number N=pq, for 2 primes p and q,

$$\Phi(N)=(q-1)(p-1).$$

The RSA Cryptosystem

The RSA encryption scheme requires 2 large distinct prime numbers, p and q. Let N=pq. Then, pick 2 numbers e and d such that $ed \equiv 1 \pmod{\Phi(N)}$. Then, the public key is {e,N}. The private key is d.

To encrypt a message m, compute $m^e \pmod{N}$. To decrypt the message, we simply need to raise the result to the power d to get:

$$(m^e)^d \equiv m^{ed} \equiv m \pmod{N}.$$

Thus, the original message is recovered.

Key length\year	2000	2016	2030 (Projected)
512	4.826*10^142	6.409*10^141	3.769*10^141
1024	3.236*10^296	4.299*10^295	2.528*10^295
2048	2.912*10^604	3.868*10^603	2.274*10^603
4096	4.707*10^1220	6.253*10^1219	3.677*10^1219

Table 1: Approximate time, in seconds, for a successful brute force attack on RSA for given key length by average computer for that year. Estimates were based on average floating point operations per seconds (FLOPS) of typical desktop computers for given year. Projections for computer strength based on Moore's law. Improvements in algorithm efficiency were not taken into consideration.

Key length\year	2000	2016	2030 (Projected)
512	5.215*10^138	1.113*10^135	3.769*10^131
1024	3.498*10^292	7.465*10^288	2.528*10^285
2048	3.147*10^600	6.717*10^596	2.274*10^593
4096	5.088*10^1216	1.086*10^1213	3.677*10^1209

Table 2: Approximate time, in seconds, for a successful brute force attack on RSA for given key length by most powerful supercomputers of that given year. Estimates were based on floating point operations per seconds (FLOPS) of most powerful supercomputers for given year. Projections for computer strength based on historical improvements to supercomputing power. Improvements in algorithm efficiency were not taken into consideration.

Since taking powers of a number and taking the modulus is a trapdoor function, one cannot easily deduce the original message from $m^e \pmod{N}$. However, knowing the trapdoor d, Alice can easily recover the message.

How difficult is deducing the message? The problem reduces to finding the primes factors of N, which is known to be exponential in the digits of N. In Tables 1 and 2, we computed approximate times required to find these factors using a brute force attack. (For perspective, the estimated age of the universe is about 10^17 seconds).

Shor's Algorithm

As seen previously, the RSA cryptosystem is intractable with respect to classical attacks. However, in 1994, Shor outlined an algorithm capable of breaking RSA. This algorithm would run on a theoretical quantum computer. Although no quantum computers today are powerful enough to effectively compute the algorithm, many improvements have been made over the last decade to this end. Should powerful quantum computers be built, the RSA cryptosystem would no longer be a safe encryption scheme.

Future Research

A proposed alternative encryption scheme, known as NTRU, is thought to be a resistant to attacks similar to that outlined by Shor. An interesting direction of research is to prove that NTRU is secure and resistant to quantum attacks.

References

1. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems ,R. L. Rivest, et al., *Comm. Assoc. Comput. Mach.*, (1978).
2. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, P. W. Shor, *SIAM J. Comput.*, (1994).

Acknowledgements

I would like to thank Prof. Monica Nevins for her guidance throughout the project. I would also like to thank UROP for opportunity to conduct this research.