

Quantum Computers compromise asymmetric key cryptography because it essentially depends on two large prime number factors forming a large number, with one factor each used to compute the private key and the public key. If an attacker can factorize the large number, they then have the keys.

Peter Shor's algorithm for quantum factorization transforms the factorization problem into a problem of finding the period of a function. Then Quantum Computers are orders of magnitude better at this period finding problem than classical computers.

## So, what is Shor's algorithm?

Shor's algorithm reduces "the factorization of N" problem into one of "finding the period P of the function  $f(x)=R^x \text{ mod } N$ , where R is a random number less than N which is not already a factor of N".

*Example, factorizing 30 using Shor's Algorithm*

**Step 1:** pick a random number less than 30. Say 7

**Step 2:** verify it is not already a factor.  $\text{GCD}(7,30)$  is 1 or  $30 \text{ mod } 7$  is 2 which is  $> 0$

**Step 3:** define a function  $f(x) = 7^x \text{ mod } 30$

X	F(X)	X	F(X)	X	F(X)
1	7	2	19	3	13
4	1	5	7	6	19
7	13	8	1		.. and so on

As we can see, period is 4.

**Step 4:** factors are  $\text{GCD}(R^{(P/2)+1}, N)$  and  $\text{GCD}(R^{(P/2)-1}, N)$

$7^{(4/2)} = 49$ . Plus 1 = 50 and Minus 1 = 48

$\text{GCD}(48,30)=6$  (*Greatest Common Divisor*)

$\text{GCD}(50,30)=10$

So, the **factors are 6 and 10**.

We can now do this iteratively to find prime factors.

## Why are Quantum Computers better at finding period?

In very simplified and not necessarily most accurate terms, we can explain this something like: A Quantum Bit or Qubit is a superposition state – it simultaneously holds all values b/w 0 and 1. So if x is represented in Qubits, then doing  $f(x)$  on this set of qubits representing x gives us all the values of  $f(x)$  at the same time. Doing a Quantum Fourier Transform (QFT) on this results in a probability wavefunction that has an interference pattern whose peaks reveal the period of  $f(x)$ . So step 3 is super easy with a Quantum Computer but super hard with a classical computer.

## Is Cryptography really at risk?

Current quantum computers have on the order of 100–1,000 noisy qubits, with coherence times measured in microseconds to milliseconds. In contrast, breaking RSA-2048 encryption is estimated to require millions of error-corrected qubits that can maintain coherence for many hours while running Shor's algorithm. So while quantum computers theoretically threaten RSA, in practice we are many technological breakthroughs away from such capability.

*Source: <https://github.com/mattvarghese/quantum-computing>*