RSA is a encryption algorithm that uses public and private keys to encrypt anything.

1 The Main Idea

The main idea of RSA encryption is the use of a trapdoor function; a funciton that is easy to computer in one direction, but hard to compute in the reverse direction.

The RSA algorithm makes use of the discrete logarithm property. This property combined with modular exponentiation gives us our trapdoor funciton.

1.1 Required Prior Information

In general, $\phi(p) = p - 1$ for all primes p.
Fermat's Little Theorem: $M^{p-1} \equiv 1(mod\ p)$ for all primes p.
Chinese Remainder Theorem: For some $p, q \in \mathbb{Z}$ such that p and q are coprime $(gcd(p, q) = 1)$.
If:

$$x \equiv 1(mod\ p)$$

and

$$x \equiv 1(mod\ q)$$

then

$$x \equiv 1(mod\ p \cdot q)$$

2 How it Works

Imagine your friend wants to send you sensitive information. The idea of the RSA algorithm is that you give your friend an open lock, your friend locks the information and sends it back to you. You then unlock this information with a secret key that only you have. Let's see the math that makes this work.

Suppose n is the product of 2 large prime numbers.

$$n = p \times q$$

As the sender, we generate n and some public exponent e such that $gcd(e, \phi(n)) = 1$. As we pick some e, we also compute d.

$$e \times d \equiv 1\ (mod\ \phi(n))$$

Because $\phi()$ is a multiplicative arithmetic function and we generated the 2 prime numbers that make up n (p and q), it is easy for us to compute d once we know $\phi(n)$ using the Extended Euclidian Algorithm.

$$\phi(n) = (p-1)(q-1)$$

Now that we have everything, we make n and e public for anyone who wants to send us messages. Then the sender will now encrypt their message.

$$C \equiv M^e (mod\ n)$$

The cypher text, C, gets sent back. Here's why you need the secret key, d, to get the original message.

$$C^d \equiv M^{e*d}$$

Understand that $e*d = 1+k*\phi(n)$ for some $k \in \mathbb{Z}$ because $e*d \equiv 1(mod\ \phi(n))$ and $\phi(n) = (p-1)(q-1)$.

Then:

$$M^{e*d} = M^{1+k(p-1)(q-1)} = M \cdot M^{k(p-1)(q-1)}$$

Rewritting and using Fermat's Little Theorem gives us:

$$M^{k(p-1)(q-1)} = M^{(p-1)^{k(q-1)}} \equiv 1^{k(q-1)} \equiv 1(mod\ p)$$

and

$$M^{k(p-1)(q-1)} = M^{(q-1)^{k(p-1)}} \equiv 1^{k(p-1)} \equiv 1(mod\ q)$$

Then, by the Chinese Remainder Theorem:

$$M^{k(p-1)(q-1)} \equiv 1(mod\ n)$$

So finally, we see that:

$$M \cdot M^{k(p-1)(q-1)} \equiv M \cdot 1(mod\ n)$$

This gives us the original message, M.