RSA Notes

Henry Osei

April 12, 2018

RSA: Public Key cryptographic system inverted by Rivest Shamir and Adlemann.

Trapdoor Function: Multiplying numbers (easy) but factoring them hard.

Finding Prime numbers p and q and multiplying them takes logarithmic time but the fastest ways known to factor $q \times p = n$ into p and q require (almost) exponential time.

Steps of RSA:

1)Alice picks two large prime numbers (p and q) randomly multiplies them together $n = p \times q$.

2)Pick exponent e (often 65537) such that

$$gcd(e, (p-1)(q-1)) = 1.$$

These two numbers (n, e) are Alice's public Key.

Bob wants to send Alice a message he encodes it as a number.

He computes

$$C = m^e \equiv \pmod{n}$$

He sends C to Alice.

How does Alice decrypt C? Recall the basic principle of modular exponentation.

$$X^{\varphi(n)} \equiv 1 \pmod{n}$$

So Alice wants a number d such that

$$de \equiv 1 \pmod{\varphi(n)}$$

this means $de = 1 + k(\varphi(n))$
Then Alice Computes

$$C^d \equiv (m^e)^d \equiv m^{ed} \equiv m^{1+k\varphi(n)} \equiv m^1(m^{\varphi(n)})^k \equiv m \pmod{n}$$

How does Alice find d?

$$d = e^{-1} \pmod{\varphi(n)}$$

what is $\varphi(n)$?

$$\varphi(n) = \varphi(p \times q) = p \times q(1 - 1/p)(1 - 1/q) = (p - 1)(q - 1)$$

Alice uses Euclid's Algorithm to find

$$d = e^{-1} \pmod{\varphi(n)}$$

If Eve wants to decrypt c she needs to know d, but this means finding the inverse of e mod $\varphi(n)$ which requires factoring n which is hard (we think).

Ex: Suppose Alice picks

$$p = 5$$
$$q = 11$$

$$n = p * q = (5 * 11) = 55$$
$$e = 7$$

check gcd(7,(5-1)(11-1))=1

She computes

$$d = 7^{-1} \pmod{\varphi(55)}$$
$$d = 7^{-1} \pmod{40}$$

Euclids Algorithm:

$$
\begin{aligned}
40 &= 5(7) + 5 \\
7 &= 1(5) + 2 \\
5 &= 2(2) + 1
\end{aligned}
$$

$$
\begin{aligned}
1 &= 5 - 2(2) \\
&= 5 - 2(7 - 1(5)) \\
&= -2(7) + 3(5) \\
&= 2(7) + 3(40 - 5(7)) \\
1 &= 3(40) - 17(7)
\end{aligned}
$$

$$d = -1 \equiv 23 \pmod{40}$$

d is Alice's Private Key: Suppose Bob wants to send Alice the message

$$m = 2$$

Bob Computes:

$$
\begin{aligned}
C &= 2^7 \pmod{55} \\
7 &= 4 + 2 + 1
\end{aligned}
$$

$$
\begin{aligned}
2^1 &\equiv 2 \pmod{55} \\
2^2 &\equiv 4 \pmod{55} \\
2^4 &\equiv 16 \pmod{55}
\end{aligned}
$$

So

$$
\begin{aligned}
C = 2^7 &= 2^1 * 2^2 * 2^4 \pmod{55} \\
&= 2 * 4 * 16 \pmod{55} \\
&= 2 * 9 \equiv 18 \pmod{55}
\end{aligned}
$$

Bob sends 18 to Alice

Alice wants to decrypt this she computes

$$18^{23} \pmod{55}$$

$$23 = 16 + 4 + 2 + 1$$

$$\begin{aligned} 18^1 &= 18 \pmod{55} \\ 18^2 &= 49 \pmod{55} \\ 18^3 &= 36 \pmod{55} \\ 18^4 &= 31 \pmod{55} \\ 18^8 &= 26 \pmod{55} \\ 18^{16} &= 26 \pmod{55} \end{aligned}$$

$$\begin{aligned} 18^{23} &= 18^1 * 18^2 * 18^4 * 18^{16} \pmod{55} \\ &= 18 * 49 * 36 * 26 \pmod{55} \\ &= 2 \pmod{55} \end{aligned}$$

Suppose someone discovers a way to find the decryption exponent without factoring n.

Magic box that tells you d knowing (n, e)

You can compute

$$\begin{aligned} ed &= 1 + K\varphi(n) \\ ed - 1 &= K\varphi(n) \end{aligned}$$

Try different values of K to find $\varphi(n)$
Knowing d allows you to find $\varphi(n)$.

Note that

$$\begin{aligned}\varphi(n) &= (p-1)(q-1) \\ &= pq - p - q + 1\end{aligned}$$

This means that

$$n - \varphi(n) + 1 = p + q.$$