Digital Signatures

To sign message m :

Alice

Bob.

d secret key

N, e public key

$\xleftarrow{m, D(m)}$ $D(m) = m^d \bmod N$

To verify

Alice computes

$$E(D(m)) = m^{de} \pmod N$$

$$\equiv m \pmod N.$$

RSA Public Key Cryptosystem

Alice

Bob.

d = private key

N, e = public key.

Plaintext = m

Ciphertext = $E(m) = m^e \pmod N$ ⟶

to decrypt

$E(m)^d \bmod N$

$\equiv m^{de} \bmod N$

$\equiv m \pmod N$.

RSA (main idea)

$N = P \cdot Q$

$\phi(N) = (P-1)(Q-1)$.

Euler's Thm: $\forall x \in Z_N^* \quad x^{\phi(N)} \equiv 1 \pmod N$

if $\gcd(e, \phi(N)) = 1$ then

$d \equiv \frac{1}{e} \pmod{\phi(N)}$ exists.

To recover x from $x^e \bmod N$:

$y = x^e \pmod N$

$y^d \equiv x^{ed} \bmod N$

$\equiv x^{ed \pmod{\phi(N)}} \bmod N$

$\equiv x \bmod N$.

$N = 15 = 3 \cdot 5$

$x \in \mathbb{Z}_N^*$	$x^3 \pmod N$
1	1
2	8
4	4
7	13
8	2
11	11
13	7

$N = 21 = 3 \cdot 7$

$x \in \mathbb{Z}_N^*$	$x^3 \pmod N$
1	1
2	8
4	1
5	20
8	8
10	13
11	8
⋮	⋮

RSA Cryptosystem

How hard is it to compute cube roots?

If $a = x^3$ for $x \in \mathbb{Z}$ then a is called a perfect cube.

Input : a

Output : x : $a = x^3$

or

"a is not a perfect cube"

Perfect cubes are sparse:

1, 8, 27, 64, 125, ...

Primes are abundant.

Theorem [Euclid]: There are infinitely many primes.

Proof: Suppose finitely many primes

$$p_1, p_2, \ldots, p_k$$

Let $m = p_1 \cdot p_2 \cdot \cdots \cdot p_k + 1$

m not divisible by any p_i. Contradiction!

$\pi(n)$ = number of primes $\leq n$

Theorem [Hadamard] $\lim_{n \to \infty} \pi(n) = \frac{n}{\ln n}$.

$$P[\text{random 100 digit number is prime}] = \frac{\pi(10^{100})}{10^{100}}$$

$$\approx \frac{1}{\ln 10^{100}} = \frac{1}{100 \ln 10} \approx \frac{1}{230}.$$

Carmichael numbers fail Fermat's test.

eg. 561.

Carmichael numbers: Composites N such that

$\forall a$ if $\gcd(a, N) = 1$ then

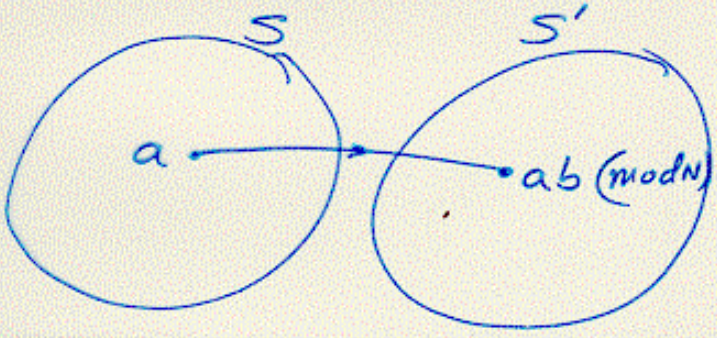
$$a^{N-1} \equiv 1 \pmod{N}.$$

Theorem: If N is a composite and N is not Carmichael then for at least half the choices of $a \pmod N$

$$a^{N-1} \not\equiv 1 \pmod{N}.$$

Proof: Let $S = \{a \pmod N : a^{N-1} \equiv 1 \pmod N\}$

N not Carmichael $\Rightarrow \exists b : b^{N-1} \not\equiv 1 \pmod N\}$



$$(a \cdot b)^{N-1} \equiv a^{N-1} \cdot b^{N-1} \not\equiv 1 \pmod N.$$

Factoring versus Primality

Factoring is hard!!

Trial division: $\sqrt{N}$ steps

$= 2^{\frac{1}{2}\log N}$

bits in N is $\log N$.

Testing primality is easy

Fermat test: To test if N prime:

Pick random $a \pmod N$ $(a \neq 0)$

Test if $a^{N-1} \equiv 1 \pmod N$

Digital Signatures