Lecture 5.3: Why RSA works

Matthew Macauley

Department of Mathematical Sciences
Clemson University
http://www.math.clemson.edu/~macaule/

Math 4190, Discrete Mathematical Structures

Generating large prime numbers

To implement RSA, we need to be able to generate large prime numbers.

In practice, this is basically done by "*guess and check*." To see both why and how this works, we'll need a little bit of number theory.

Prime number theorem

The probability that a random number n is prime is approximately $1/(\ln n)$, i.e.,

$$\lim_{n\to\infty}\Big(\text{proportion of numbers} \leq n \text{ that are prime}\Big) = \frac{1}{\ln n}.$$

The chances of a random 9-digit number being prime is approx. 4% (i.e., 1 in 25). For a 200-digit number, this is approx. 0.2% (i.e., 1 in 500).

Heuristic for finding a large prime

```
while (true) {
    let n be a random 200-digit number;
    if (n is prime)                          \\ How to check this??
        return n;
}
```

Checking whether a large number is prime

The Fermat primality test is a probabilistic method to determine whether a number is ("probably") prime. It relies on the following result.

Fermat's little theorem

For any prime p and integer a,

$$a^p \equiv a \pmod{p}.$$

Without loss of generality, assume that $a \in \{0, 1, \ldots, p-1\}$. If $a = 0$, this trivially holds.

Otherwise, $\gcd(a, p) = 1$. This means that a has a multiplicative inverse, modulo p.

Multiplying both sides by this inverse a^{-1} yields

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

We now have the following heuristic for testing for primes:

Fermat primality test

Given a number $n \in \mathbb{N}$, compute $a^{n-1} \pmod{n}$ for _many_ random values of $a < n$.

- If $a^n \not\equiv 1 \pmod{n}$ for some a, then n must be composite.
- If $a^n \equiv 1 \pmod{n}$ for every a that we try, then n is "probably prime."

Proof of Fermat's little theorem

Fermat's little theorem (restated)

For any prime p and integer a with $\gcd(a,p)=1$,

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

Proof

Assume without loss of generality that $a \in \{1, 2, \dots, p-1\}$.

Consider the list of numbers

$$a,\ 2a,\ 3a,\ \dots\ ,\ (p-1)a.$$

Claim: No two of these are equivalent modulo p.

To see why, suppose that $ka \equiv \ell a \pmod{p}$.

Multiplying by $a^{-1} \pmod{p}$ yields $k \equiv \ell \pmod{p}$.

Thus,

$$a \cdot 2a \cdot 3a \cdots (p-1)a \equiv 1 \cdot 2 \cdot 3 \cdots (p-1) \pmod{p}.$$

Rearranging terms, we get

$$a^{p-1}(p-1)! \equiv (p-1)! \pmod{p} \qquad \Longrightarrow \qquad a^{p-1} \equiv 1 \pmod{p}.$$

□

Fermat primality test

Fermat primality test (revisited)

Given a number $n \in \mathbb{N}$, compute $a^{n-1} \pmod{n}$ for *many* random values of $a < n$.

- If $a^{n-1} \not\equiv 1 \pmod{n}$, then n must be composite. We say a is a Fermat witness.
- If $a^{n-1} \equiv 1 \pmod{n}$, there are two cases:
  1. n is prime.
  2. n is composite; a is called a Fermat liar.

Lemma

If a composite number n has a Fermat witness, then at least half of all numbers $a \in \{1, 2, \ldots, n-1\}$ that are relatively prime to n are Fermat witnesses.

Proof (sketch)

Consider a Fermat witness a and Fermat liar b for n. Then

$$(ab)^{n-1} = \underbrace{a^{n-1}}_{\not\equiv 1} \cdot \underbrace{b^{n-1}}_{\equiv 1} \equiv a^{n-1} \not\equiv 1 \pmod{n}.$$

In other words, every Fermat liar b has a corresponding Fermat witness ab. □

Carmichael numbers

We just saw how if n has a Fermat witness, then it has many Fermat witnesses.

But... is it possible that n is composite, but has *no* Fermat witnesses?

Unfortunately, the answer is YES, but this is very rare.

Definition

A Carmichael number is a composite number n for which

$$a^{n-1} \equiv 1 \pmod{n}$$

holds for all $a = 1, \ldots, n-1$ relatively prime to n.

The first few Carmichael numbers are 561, 1105, 1729, 2465, 2821, 6601, 8911, ...

For 100-digit numbers, less than 1 in 10^{30} are Carmichael numbers. For 200-digit numbers, the chances are even less.

Take-away message

If we randomly choose a 200-digit number n, and test ≈ 100 different values of a without getting a Fermat witness, then we can be almost certain that n is prime.

Fermat primality test

Algorithm

Input: Integer $n > 0$.

is_composite = FALSE;

for $(i = 1, \ldots, 100)$ {

pick a random number a_i relatively prime to n;

if $(a_i^{n-1} \not\equiv 1 \pmod{n})$ `\\ a_i is a Fermat witness`

is_composite = TRUE;

end;

}

if (is_composite == FALSE)

print "*chances that n is composite is less than* 1 *in* $2^{100} \approx 10^{30}$";

else if (is_composite == TRUE)

print "*n is composite*";

Now that we know how go actually generate and compute with large primes, we can turn our attention to *why* the RSA encryption and decryption functions actually work.

Why RSA encryption and decryption work

Theorem

Let $n = pq$ and $ed \equiv 1 \pmod{(p-1)(q-1)}$.

Given a message $m < n$ with $\gcd(m, n) = 1$, set $c = m^e \pmod{n}$. Then $c^d \equiv m \pmod{n}$.

Proof

Lemma. $m^{(p-1)(q-1)} \equiv 1 \pmod{n}$.

Proof. Since $\gcd(m^{q-1}, p) = 1$, Fermat's little theorem says

$$\left(m^{q-1}\right)^{p-1} \equiv 1 \pmod{p}.$$

Similarly,

$$\left(m^{p-1}\right)^{q-1} \equiv 1 \pmod{q}.$$

Thus, for some $k, \ell \in \mathbb{Z}$,

$$m^{(p-1)(q-1)} = 1 + kp = 1 + \ell p.$$

This means that $m^{(p-1)(q-1)} - 1$ is a multiple of both p and q, and so

$$m^{(p-1)(q-1)} - 1 = bpq, \qquad \text{for some } b \in \mathbb{Z},$$

completing the proof of the Lemma. ✓

Why RSA encryption and decryption work

Theorem

Let $n = pq$ and $ed \equiv 1 \pmod{(p-1)(q-1)}$.

Given a message $m < n$ with $\gcd(m, n) = 1$, set $c = m^e \pmod{n}$. Then $c^d \equiv m \pmod{n}$.

Proof

Lemma (established). $m^{(p-1)(q-1)} \equiv 1 \pmod{n}$.

We know $c^d \equiv m^{ed} \pmod{n}$, and need to show $c^d \equiv m \pmod{n}$. Thus, it suffices to show

$$m^{ed} \equiv m \pmod{n}.$$

Note that $ed \equiv \pmod{(p-1)(q-1)} \Leftrightarrow \exists j \in \mathbb{Z}$ such that $ed = 1 + j(p-1)(q-1)$.

Now,

$$m^{ed} = m^{1+j(p-1)(q-1)} = m \cdot m^{j(p-1)(q-1)} = m \cdot \underbrace{\left(m^{(p-1)(q-1)}\right)^j}_{\equiv 1,\text{ by Lemma}} \equiv m \pmod{n}.$$

□