MATH CIRCLE : CRYPTOLOGY

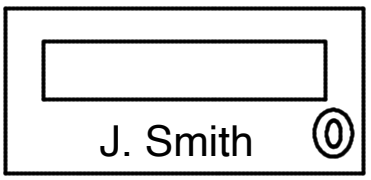
Uses:

- firewalls
- routers
- VPN devices
- printers
- VOIP phones
- ATM machines, electronic banking
- data encryption for transfer (email, file transfer, internet purchases, . . .)
- espionage.

Basic idea: disguise messages so that only intended recipient can understand them. Suppose I want to send you 3 numbers and we decide that I will multiply all the numbers I send by 2. I send 10, 58, 42. You know this message is really 5, 29, 21. This needed us to discuss with each other in advance what our encryption method was going to be, which is often not possible or practical.

Public-key Encryption

Analog:



lockable mailbox

1. Anyone can send mail to my mailbox
2. Sender cannot retrieve the mail once it is in the mailbox.
3. Noone else can ——"——

Public-key System:

1. Anyone can send me an encrypted message. knowing some public information about me (location of my mailbox)
2. Only I can decode the message (even the Sender cannot decode the message)
3. No need to exchange secret codes.

RSA Algorithm: (Rivest, Shamir, Adleman)

I have 2 keys:

- public key: $n=22$, $e=3$ (location of mailbox)
- private key: $d=7$ (known only to me)

You want to send me an integer m between 1 and 22, say $m=14$. Compute $m^e = 14^3$, ÷ by $n=22$, and find the remainder = the code, so

$$\frac{14^3}{22} = \frac{2744}{22} = 124 + \frac{16}{22} \Rightarrow \text{rem} = 16 \uparrow \text{code}$$

So code $c. = 14^3 \bmod 22 = 16.$

You only used 22 and 3 to encode the message.

Decode: (Bad) I could hunt for all integer solutions m to the equation

$$c = m^3 \bmod 22$$

i.e. $16 = m^3 \bmod 22$, or

(good) I could use my private key $d=7$

$$m = c^7 \bmod 22$$

= remainder after ÷ c^7 by 22

$$m = 16^7 \bmod 22 = 14.$$

Exercises:

1. I give you $n=22$, $e=3$ and $c=18$. Find m. (Ans: $18^7 \bmod 22 = 6.$)
2. In pairs, each picks a number $\in \{1,2,\ldots,22\}$, encodes it, and gives it to partner, who has to decode it.

The reason the RSA Algorithm works is due to number theory.

$n = pq, \quad p \neq q$ [$22 = 2 \cdot 11$]

public

private } prime numbers chosen by recipient of messages (type of lock on mailbox).

e = any integer such that $1 < e < (p-1)(q-1)$ and $\gcd(e, (p-1)(q-1)) = 1$.

Recipient finds $d > 0$ such that $ed = 1 \bmod (p-1)(q-1)$

[d is the mailbox key]

In our example, $p=2$, $q=11$, so $n = 2 \cdot 11 = 22$. $e=3$ and $\gcd(3, 10) = 1$, $3 \cdot 7 = 21 = 1 \bmod 10$.

Why does RSA Algorithm work? See exercises.

Exercises

3. Let p, q be prime numbers with $p \neq q$. Let $a, b \in \mathbb{Z}$. Show that $a = b \bmod pq$ iff $a = b \bmod p$ and $a = b \bmod q$.

Ans: $a = b \bmod pq \iff a - b = 0 \bmod pq \iff pq \mid a-b$

$\iff p \mid a-b$ and $q \mid a-b$ (since p, q are distinct primes) $\iff a - b = 0 \bmod p$ and $a - b = 0 \bmod q \iff a = b \bmod p$ and $a = b \bmod q$.

4. Let $r, n, d \in \mathbb{N}$. Show that
$(r \bmod n)^d = r^d \bmod n$.

5. Fermat's Little Theorem (FLT) states:
if p prime and $a \in \mathbb{Z}$ with $p \nmid a$, then
$a^{p-1} = 1 \bmod p$.
Use FLT and #3 and #4 to show that, in the RSA Algorithm,

$$c^d = m \bmod n.$$

Ans $c^d = (m^e \bmod n)^d$

$= (m^e)^d \bmod n$ by #4

$= m^{ed} \bmod n$

So we want $m^{ed} = m \bmod n$.

By #3, we must show that
$m^{ed} = m \bmod p$ and $\bmod q$.

$m^{ed} = m^{1+(p-1)(q-1)b}$ for some $b \in \mathbb{N}$

$= m(m^{p-1})^{(q-1)b} = m(m^{q-1})^{(p-1)b}$

$= \begin{cases} m(1)^{(q-1)b} \bmod p = m \bmod p \\ m(1)^{(p-1)b} \bmod q = m \bmod q \end{cases}$ by FLT

Remarks

1. If words or letters are in message, then they are converted to numbers.
2. The numbers used are usually converted to binary numbers.
3. The prime numbers p and q are taken to be very big, ≥ 200 digits each in base 10. As computers improve, p and q need to be bigger and bigger. Ideally, we want n to be so big, that no present-day computer can factor it in a decent amount of time.
4. m needs to be big enough that trying to find sols m to equation $c = m^e \bmod n$ takes too long to be helpful to a code breaker.
5. Once e is known, $\exists$ an algorithm to find d, using a computer.
6. There is a good deal of money spent on finding large prime numbers and on algorithms to generate them. The most efficient methods use both number theory and computers combined.
7. According to wikipedia, the largest prime number known (as of Oct 2013) is

$$2^{57,885,161} - 1,$$

a number with 17,425,170 digits.