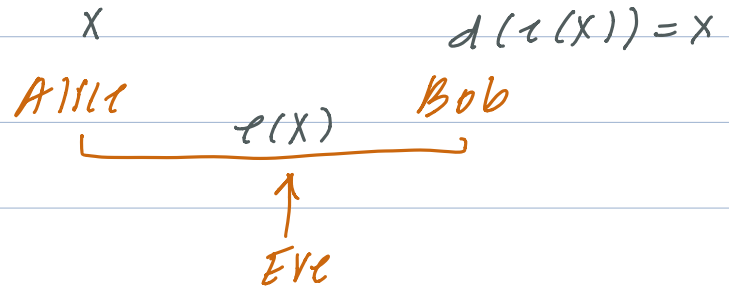


RSA and Crypto!

message x Encoding $e(x)$ Decoding $d(e(x))$ Information-Theoretic Approach - One Time Pad

$$\oplus := x \oplus k$$

$$\begin{array}{r} 110101 \\ \oplus 011001 \\ \hline 101100 \end{array}$$

message x

$$|r| = |x|$$

random string r

private

$$\text{encoding } e(x) = x \oplus r$$

$$\text{decoding } d(e(x)) = e(x) \oplus r$$

$$= (x \oplus r) \oplus r = x$$

Eve sees $e(x)$. Does she gain any information about x ?

$$\Pr(\text{message is } x \mid e(x))$$

$$= \Pr(\text{message is } x) \leftarrow \text{original guess}$$

* Using one-time pad multiple times gives info about r !

$$\begin{aligned} e(x) & \quad e(x) \oplus e(y) \\ e(y) & = x \oplus r \oplus y \oplus r \\ & = x \oplus y \end{aligned}$$

RSA - Public Key Cryptography

Based on computational hardness

→ If Eve could break RSA scheme, then she'd know how to solve a very hard class of problems

Needs:

- Generate big prime numbers (primality test)
- Fast Exponentiation (repeated squaring)
- Euclid's Algo + Extended Euclid's Algo

Euclid's Algo

Greatest Common Divisor

$\text{gcd}(a, b) = \text{largest int } d, d \text{ divides } a \text{ and } b$
 $= d \mid a, d \mid b$

$\text{gcd-Euclid}(a, b)$ poly logarithmic 360, 84
↳ time proportional to # of digits 84, 24
if $b == 0$ return a 24, 12
return $\text{gcd-Euclid}(b, a \bmod b)$ 12, 0

$\text{gcd}(b, a \bmod b) = \text{gcd}(a, b)$ $O(\log a)$ rounds
↓
must at least half every round

$b \leq a/2$: then in one round cut by $1/2$

$b > a/2$: $a \bmod b = a - b < a/2$
then in two rounds decreases by $1/2$ } at most $2 \times \log_2 a$ rounds

most number of rounds when a is a fib #

Extended Euclid's

$\text{ec}(a, b)$

returns $d = \text{gcd}(a, b)$

and integers x, y $ax + by = d$

method to find multiplicative inverses

$$\text{gcd}(a, b) = 1$$

$$\text{gcd}(a, p) = 1$$

when $p \nmid a$

$$ax = 1 \pmod{p}$$

$$ax + py = 1$$

$$ax = 1 \pmod{p}$$

RSA Protocol

Bob - public key
[private info]

Bob chooses p, q primes
512 bit
^
(of roughly equal length)

Bob computes $n = p \times q$

and finds random int e s.t.

$$\text{gcd}((p-1), (q-1), e) = 1$$

[$e=3$]

(n, e) is Bob's

public key

Bob's private info is

$$d = e^{-1} \pmod{(p-1)(q-1)}$$

by extended Euclid's Algo

Alice takes

message is a number mod n

$$c(x) = x^e \pmod{n} \leftarrow \text{by fast exponentiation}$$

To decode, Bob takes

$$d(c(x)) = (c(x))^d \pmod{n}$$

(claim. $d(c(x)) = x \pmod{n}$)

Pf. $d(c(x)) = x^{e \cdot d} \pmod{n}$

e and d are multiplicative inverses mod $(p-1)(q-1)$
 $d(e(x)) = x^{1+k(p-1)(q-1)} \bmod n$

Show: $x^{1+k(p-1)(q-1)} = x \bmod p = x \bmod q$

$x^{p-1} = 1 \bmod p$ by Fermat's Little Thm

↑ if $x \neq 0 \bmod p$

$$x^{k(p-1)(q-1)} = 1 \bmod p$$

$$x^{1+k(p-1)(q-1)} = x \bmod p$$

} same arg for q

How would Eve decode?

Factor n into $p \times q$ and compute d