Cryptography (14.1, 14.3)

Motivation: Want to send messages which only intended recipient can read, by "encrypting" message.

Example (Caesar cipher): Assign each Latin letter a number, $A=1$, $B=2$, $C=3$ etc, $Z=26$

Choose a $k \in \mathbb{Z}$, & replace each letter ~~key~~ no. i by letter no $i+k \pmod{26}$.

$k=2$

A	B	C	D	E	...	Y	Z
↓	↓	↓	↓	↓		↓	↓
C	D	E	F	G	...	A	B

DISCRETE $\xrightarrow{\text{encrypt}}$ FKUETGVG $\xrightarrow{\text{decrypt}}$ DISCRETE

Eavesdropper intercepts FKUETGVG & is stumped.

Remark: Caesar cipher & "substitution ciphers" vulnerable to frequency attacks, e.g. E is most common letter in English, can use this to crack.

Formal version: A cryptosystem is a triple $(\mathcal{T}, \mathcal{C}, \mathcal{K})$ & functions $e: \mathcal{T} \times \mathcal{K} \to \mathcal{C}$, $d: \mathcal{C} \times \mathcal{K} \to \mathcal{T}$,

where

- $\mathcal{T}$ is set of texts
- $\mathcal{C}$ is set of possible "cryptograms" (= encrypted texts)
- $\mathcal{K}$ is a set of keys

c is an encryption function
d is the decryption function.

Ex Caesar cipher, $A = \{A, _, Z\}$, $\mathcal{T} = A^*$, $\mathcal{C} = A^*$,

$K = \{0, 1, \ldots, 25\}$ $\quad c(\text{DISCRETE}, 3) = \text{FKUETGVG}$

$d(\text{FKUETGVG}, 3) = \text{DISCRETE}$

Want • c, d easy to compute (with key)

• $d(e, k)$ hard/impossible to compute without key k.

Ex A "perfect" cryptosystem, "one-time pad."

Let $A = \{0,1\}$, $\mathcal{T} = \mathcal{C} = K = A^*$. $(t_i) \in \mathcal{T}$, $(k_i) \in K$,

$$c((t_i),(k_i)) = (t_i + k_i \pmod 2)$$

$$d((c_i),(k_i)) = (c_i + k_i \pmod 2).$$

If key is "sufficiently random & not used more than once, this is unbreakable.

Problem: How to share keys?

Diffie-Hellman key exchange: Let p be a prime.

Fact: $\exists\, a \in \mathbb{Z}_p$ such that every $b \in \mathbb{Z}_p \setminus \{0\}$ is of form $b = a^k$ for some k. Call a a **primitive root mod p**.

Ex $p = 11$, $a = 2$ is a primitive root

k	0	1	2	3	4	5	6	7	8	9	10
$a^k \bmod 11$	1	2	4	8	5	10	9	7	3	6	1

Fact: Easy to compute $a^k \bmod p$

Given a, b, p, hard to find k

such that $a^k \equiv b \pmod p$.

$k = \log_a b$, the discrete logarithm.

Diffie-Hellman: A & B want to find a shared key.

(1) A & B agree on a prime p (large) & a primitive root m mod p.

(2) A & B choose secret integers a & b.

(3) A sends $c \equiv m^a \bmod p$, B sends $d \equiv m^b \bmod p$

(4) A computes $d^a \bmod p$ B computes $c^b \bmod p$

(5) Shared key is $k = d^a = (m^b)^a = m^{ab} = (m^a)^b = (c)^b \bmod p$.

Eavesdropper E has access to p, m, c & d. From this it is hard to compute k.

RSA protocol A wants to receive messages safely.

(1) A Chooses large primes p, q, & 2 numbers e, d such that $ed \equiv 1 \bmod (p-1)(q-1)$.

(2) Let $n = pq$, make n & e public.

(3) B encrypts message m to $c = m^e \pmod n$.

(4) A decrypts by $c^d \pmod n$.

Claim: $c^d \equiv m \pmod n$

$ed \equiv 1 \pmod{(p-1)(q-1)} \Rightarrow ed \equiv 1 \pmod{p-1}$

$\Rightarrow ed = k(p-1) + 1 \quad k \in \mathbb{Z}$

$\Rightarrow c^d \equiv m^{ed} = m^{k(p-1)+1} \equiv (m^{p-1})^k \cdot m \equiv 1^k \cdot m \equiv m \pmod p$.

Similarly $c^d \equiv m \pmod q$, so (Chinese remainder theorem)

$c^d \equiv m \pmod{n}$.

Eavesdropper sees n, c, e.

In principle can decrypt by

① finding p, q from $n = pq$

② solving equation $de \equiv 1 \pmod{(p-1)(q-1)}$.

Here ② is easy, but ① is hard.