Lecture #6: Cryptography

Reading: 4.5.1, 4.6.4, 4.8

Announcement: Handout

Cryptography: The art/science of keeping secrets

Key ingredients: encryption, decryption and a secret key.

Let's see how this works with some historical examples

Caesar Cipher: Modularly shift letters by k (secret key)

(k=3) e.g. H A I L C A E S A R (message)

K D L O F H D V D U (ciphertext)

What would Z become?

This is called encryption: message → ciphertext

How do we recover the message from ciphertext?

shift back

e.g. H W W X E U X W H

E T T U B R U T E

This is called decryption: ciphertext → message

But is this scheme secure against an eavesdropper?

No, could just try all 26 possible secret keys

Substitution Cipher: Choose a way to substitute letters, one-by-one

e.g. A → R
B → C
C → D
D → Q
⋮

hence BAD CAB becomes CRQ DRC

This is vulnerable to frequency analysis (e.g. E → ?)

Last example, from WWII

Enigma Machine: (show video)

Modern cryptography relies on modular arithmetic

messages ⟷ numbers

e.g.	T	A	N	K	
	20	01	14	11	= m

Our first attempt will use primes (e.g. n = 26170819)

Let k = 15 3 1 8 1, 9 2

then we compute c = rem(km, n)

$$15318192 * 20011411 \pmod{26170889}$$

$$= 16137645$$

Given $c = \text{rem}(km, n)$, how can we recover m?

From the pulverizer, we have:

$$sk + tn = 1$$

Here $s = 7993233$, $t = -4678565$

Thus we decrypt as $\text{rem}(sc, n)$:

$$7993233 * 16137645 \pmod{26170889}$$

$$= 20011411$$

(□)

Scheme #1: prime n, key k, message m

Enc: $c = \text{rem}(km, n)$ $\quad c \equiv km \pmod n$

Dec: $\text{rem}(sc, n) = \text{rem}(skm, n) = m$

$sc \equiv skm \equiv m \pmod n$

where $sk \equiv 1 \pmod n$

Let's prove something interesting about it

Lemma 1: Let n be prime, and $k \not\equiv 0 \pmod n$, then

$\text{rem}(k, n), \text{rem}(2k, n), \ldots \text{rem}((n-1)k, n)$

is a permutation of the sequence $1, 2, \ldots n-1$

Let's see on example first, $n=7$, $k=5$

$$\begin{array}{ccccc} \text{rem}(5,7) & , & \text{rem}(10,7), & \text{rem}(15,7) \\ \| & & \| & \| \\ 5 & & 3 & 1 \end{array}$$

$$\begin{array}{ccccc} \text{rem}(20,7) & , & \text{rem}(25,7) & , & \text{rem}(30,7) \\ \| & & \| & & \| \\ 6 & & 4 & & 2 \end{array}$$

All numbers 1–6 appear once and only once

Proof: The sequence contains $n-1$ numbers.

Moreover $ik \equiv jk \pmod n \leftrightarrow i \equiv j \pmod n$ (why?)

Also zero does not appear, hence all numbers in sequence are distinct ☒

So our scheme is a shuffling (of all numbers 1 – 26170818)

—

Now let's break it using a plain text attack

If an eavesdropper knows a message and its encoding:

$$m, \quad c = \text{rem}(km, n)$$

how can he recover k?

Since $\gcd(m,n)=1$, we can find integers s,t

$$sm+tn=1$$

then $sc \equiv skm \pmod n$

$$\equiv k \pmod n$$

—

Now we're finally ready to introduce RSA, recall

Thm [Euler]: If $n=pq$ for different primes p,q then for any k with $\gcd(k,n)=1$

$$k^{(p-1)(q-1)} \equiv 1 \pmod n$$

Last time we did an example, let's do one more to make sure you're comfortable

$n=2\cdot 5$, $k=3$, then

(*) → $3^{(2-1)(5-1)} = 3^4 = 81$, and $81 \equiv 1 \pmod{10}$

Scheme #2 [RSA] $n=pq$, for different primes p,q

Setup: Select e so that $\gcd(e,(p-1)(q-1))=1$
and let $de \equiv 1 \pmod{(p-1)(q-1)}$

Enc: $c = \text{rem}(m^e, n)$

Dec: $\text{rem}(c^d, n) (= m)$

Lemma 2: $\text{rem}(c^d, n) = m$

Proof: Using modular arithmetic, we have:

$$c \equiv m^e \pmod{n}, \text{ then}$$

$$c^d \equiv (m^e)^d \equiv m^{ed} \pmod{n}$$

Using Euler's theorem

$$m^{ed} \equiv m^{\text{rem}(ed, (p-1)(q-1))} \pmod{n}$$

Now using $ed \equiv 1 \pmod{(p-1)(q-1)}$ we have

$$\equiv m \pmod{n}$$

Since c^d and m differ by a multiple of n, and $0 \le m < n$ we have $m = \text{rem}(c^d, n)$ ☒

(*) Alternatively $k^5 \equiv k^{\text{rem}(5, (p-1)(q-1))} \pmod{n}$

e.g. $3^{31} \equiv 3^{27} \equiv 3^{23} \ldots \equiv 3^3 \equiv 7 \pmod{10}$

(11, 27)

In fact, this scheme is public key: Can tell everyone (e, n) which allows them to transmit to you, but only you know d and can decrypt

(□) Fact: $m \equiv \text{rem}(m,n) \pmod{n}$

Proof: By Division Algorithm

$$m = qn + r \qquad 0 \le r < n$$
$$r = \text{rem}(m,n)$$

then taking $(\text{mod } n)$ on both sides

$$m \equiv \cancel{qn} + r \pmod{n}$$ □

The security of RSA rests on:

Assumption: Given n (product of two primes), it is hard to factor it

If we knew (or could compute) $n = pq$, then we could use the Pulverizer to find

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

and decrypt secret messages

(Show Nash's declassified letter)

What else does the NSA know about?