Chapter 2

Private Key Cryptography and Computational Results

PRIVATE KEY CRYPTOGRAPHY

SHIFT CIPHERS

Computations will be done mod 26 because there are 26 letters in English. We will convert English to numerical equivalents, given by the following assignment:

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

In the shift cipher, Alice and Bob meet and pick a key number, say k. k is the private key, known only to them.

Alice can send a message to Bob by converting the letters to numbers, x. Compute $y \equiv x + k \bmod n$. Send y to Bob. Bob computes $x = y - k \bmod n$ to get the message. Note that Eve does not know k so can not readily do the final step

Example: Alice wants to send a message
Here is how it goes: $K=17$

A	T	T	A	C	K	A	T	D	A	W	N
0	19	19	0	2	10	0	19	3	0	22	13
17	10	10	17	19	1	17	10	20	11	13	4
R	K	K	R	T	B	R	K	U	R	N	E

1 → 2 Convert to numbers
2 → 3 Use $y = x + 17 \bmod 26$
3 → 4 Convert back to letters

The line 3 or 4 is the cipher text
The line 1 or 2 is the plain text
$K=17$ is the key

For Bob to recover the message, he reverses the steps 1 to 4. From 3 to 2 he uses $x = y - 17 \bmod 26$

How might Eve find the message
She knows the y's as they were sent. She can compute $x = y - k \bmod 26$ for all different K from 1 to 25
When she gets a message that make sense, she has broken the system.
So, we see, this method is not secure.
There are also tricks that she can do which speed up the process, frequency analysis. See text book.

Affine Ciphers

To make the system a little more secure, we use $y = ax + k$ where a is an integer from 1 to 25.

We will need $x = a^{-1}(x - k) \bmod 26$

Thus a must have an inverse mod 26. We will look at the process of finding inverses shortly but for now, a has an inverse mod 26 if and only if the largest positive integer that divides a and 26 is 1, ie greatest common divisor of a and 26 is 1, written $g.c.d(a, 26) = 1$

The numbers and their inverses mod 26 are: (1,1), (3,9), (5,21), (7,15) (11,19), (17,23), (25,25)

Ex Alice and Bob decide to use an affine cipher with $a = 3$, $k = 4$:

$$y = 3x + 4 \bmod 26$$

so $x = 9(y - 4) \bmod 26$

Again, use the message from the last example to get

A	T	T	A	C	K	A	T	D	A	W	N
0	19	19	0	2	10	0	19	3	0	22	13
4	9	9	4	10	8	4	9	13	4	18	17

The last line is the ciphertext and can be converted back to English if desired

To recover the message, Bob reverses the steps using $x = 9(y-4)$ mod 26 to go from the third line to the second.

THE HILL SYSTEM

Now a square matrix A is the key. It must have an inverse which happens when

$$\text{g.c.d}(\det A, 26) = 1$$

So $\det A$ is an odd number from 1 to 25, excluding 13. This statemen can be seen as follows: If $AB = I$ Then $\det A \det B = \det AB = \det I = 1$ so $\det B = (\det A)^{-1}$.

Conversely, we learn in linear algebra that $A \text{ adjoint } A = \det A \, I$

So $A \dfrac{\text{adjoint } A}{\det A} = I$ when $\det A$ has an inverse So $A^{-1} = \dfrac{1}{\det A} \text{ adjoint } A$.

Our examples will be 2 by 2 so finding inverses (finding adjoints) is easy

Example $A = \begin{pmatrix} 2 & 5 \\ 1 & 4 \end{pmatrix}$ $\det A = 3$

$\gcd(\det A, 26) = 1.$ $(\det A)^{-1} = 9$

$$A^{-1} = \begin{pmatrix} 4 & -5 \\ -1 & 2 \end{pmatrix} \frac{1}{\det A} = 9 \begin{pmatrix} 4 & -5 \\ -1 & 2 \end{pmatrix} =$$

$$9 \begin{pmatrix} 4 & 21 \\ 25 & 2 \end{pmatrix} \bmod 26 = \begin{pmatrix} 10 & 7 \\ 17 & 18 \end{pmatrix} \bmod 26$$

Example Alice and Bob pick key to be $A = \begin{pmatrix} 2 & 5 \\ 1 & 4 \end{pmatrix}$

Alice has message

M E E T A T S E V E N

12 4 4 19 0 19 18 4 21 4 13

She puts these in row vectors of length 2 (since A is 2 by 2)

$P_1 = (12, 4)$ $P_2 = (4, 19)$ $P_3 = (0, 19)$

$P_4 = (18, 4)$ $P_5 = (21, 4)$ $P_6 = (13, 25)$

Note the last 25 was picked at random so the final vector has length 2.

Alice computes

$C_1 = (12, 4) A = (12 \ 4) \begin{pmatrix} 2 & 5 \\ 1 & 4 \end{pmatrix} = (2 \ 25) \bmod 26$

and does similar steps for the other vectors. She gets

(2, 24), (1, 18), (19, 24), (14, 2), (20, 17), (25, 9)

ad sends this cipher Text to Bob
{ Call these C_1 C_2 C_3 C_4 C_5 C_6 }

Bob computes
$C_1 A^{-1}$, $C_2 A^{-1}$, ..., $C_6 A^{-1}$ all mod 26

ie

$$C_1 A^{-1} = (2, 24) \begin{pmatrix} 10 & 7 \\ 17 & 18 \end{pmatrix} \bmod 26 = (12, 4) \text{ etc}$$

Note that the first two E's become 24 and 1 when creating the ciphertext. Previously, they would go to the same number. The earlier ones are called substitution ciphers and there are techniques that Eve can use to break them. The Hill system is not a substitution cipher so is somewhat more secure.

PROBLEMS

I. Alice and Bob set up an affine cipher using

$$y = 9x + 2 \mod 20$$

Alice encripts a message and gets

U Y Z Y P C I M Z A I

as her ciphertext which is sent to Bob What is her message?

II Alice and Bob decide to use the Hill system with key

$$A = \begin{pmatrix} 3 & -1 \\ -1 & 2 \end{pmatrix}$$

Bob encripts a message and gets

14, 10, 24, 14, 15, 22, 1, 13, 9, 20, 13, 0
24, 14, 17, 25

what is his message?

Computational Tools

I We are interested in computing

$$a^m \bmod n$$

where a m and n are given positive integers. Always

$$m = x_0 2^0 + x_1 2^1 + x_2 2^2 + \cdots + x_t 2^t$$

where x_i is 0 or 1 and $x_t = 1$

Ex $21 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 0 \cdot 2^3 + 1 \cdot 2^4$

Then $a^m = a^{x_0 2^0 + \cdots + x_t 2^t} =$

$$a^{x_0 2^0} a^{x_1 2^1} \cdots a^{x_t 2^t}$$

We can compute the exponents efficiently as follows:

$$a^{2^j} = a^{2 \cdot 2^{j-1}} = (a^{2^{j-1}})^2$$

In short, we compute a list of a^{2^j} by squaring each element, one at a time

Ex $3^{218} \bmod 1000$

$$218 = 128 + 64 + 16 + 8 + 2$$
$$= 2^7 + 2^6 + 2^4 + 2^3 + 2$$

$$3^{218} = 3^{2^7} 3^{2^6} 3^{2^4} 3^{2^3} 3^2 \bmod 1000$$

The list is

$3^{2^1} = 3^2 = 9$

$3^{2^2} = 9^2 = 81$

$3^{2^3} = 81^2 = 561$

$3^{2^4} = 561^2 = 721 \mod 1000$

$3^{2^5} = 721^2 = 841 \mod 1000$

$3^{2^6} = 841^2 = 281 \mod 1000$

$3^{2^7} = 281^2 = 961 \mod 1000$

$3^{218} = 3^{2^7} \cdot 3^{2^6} \cdot 3^{2^4} \cdot 3^{2^3} \cdot 3^2 \mod 1000$

$= 961 \cdot 281 \cdot 721 \cdot 561 \cdot 9 =$

$489 \mod 1000$

To compute $a^m \mod n$ where t is the highest power of 2 that appears in the expansion of m, it takes $t \le \log_2 m$ steps to compute the table and at most t steps to multiply the answers in the table, so at most $t \log_2 m$ multiplications. This is a big savings over rasing a to powers m times.

II Euclidean Algorithm

To compute inverses mod n, the Euclidean algorithm is used. which goes as follows.

First we compute the g.c.d of a and b: Suppose $a > b$

$$a = b\,q_1 + r_1 \qquad 0 \le r_1 < b$$
$$b = r_1\,q_2 + r_2 \qquad 0 \le r_2 < r_1$$
$$r_1 = r_2\,q_3 + r_3 \qquad 0 \le r_3 < r_2$$
$$\vdots$$
$$r_{n-2} = r_{n-1}\,q_n + r_n \qquad 0 \le r_n < r_{n-1}$$
$$r_{n-1} = r_n\,q_{n+1}$$

Since $r_1, \ldots r_n$ are decreasing positive integers, this sequence terminates. r_n is the gcd

This can be seen by working with the equations

Ex $a = 81 \qquad b = 64$

$$81 = 64 \cdot 1 + 17$$
$$64 = 17 \cdot 3 + 13$$
$$17 = 13 \cdot 1 + 4 \rightarrow \gcd(81, 64) = 1$$
$$13 = 4 \cdot 3 + 1$$
$$4 = 1 \cdot 4$$

64 has an inverse mod 81 if and only if $\gcd(64,81)=1$ which it is. So we look for the inverse using the Euclidean algorithm Table

Row	R	Q	U	V
-1	a	$-$	1	0
0	b	$-$	0	1
1	r_1	q_1	u_1	v_1
2	r_2	q_2	u_2	v_2
$\vdots$				
n	r_n	q_n	u_n	v_n

The R and Q columns come from the list that finds the gcd. $u_{-1}=1$ $u_0=0$ $v_{-1}=0$ $v_0=1$ are initial conditions and are always the same. Note that

$$r_{i+1} = r_{i-1} - r_i q_{i+1}$$

Then $$u_{i+1} = u_{i-1} - u_i q_{i+1}$$

$$v_{i+1} = v_{i-1} - v_i q_{i+1}$$

is the way u_{i+1} and v_{i+1} are defined and are easy to remember since the follow r_{i+1}.

The reason for all this is, for each row

$$* \quad r_j = a u_j + b v_j$$

So the last row has

$r_n = a u_n + b v_n$. If $r_n = 1$, then

$1 = a u_n + b v_n$ and

$b v_n \equiv 1 \mod a$.

Hence v_n is $b^{-1} \mod a$

To show $*$ use induction

The first and second rows are seen by substitution. Assume up to $j-1$ that the result holds

Then $r_j = r_{j-2} - r_{j-1} q_j$

$= (a u_{j-2} + b v_{j-2}) - (a u_{j-1} + b v_{j-1}) q_j$

$= a(u_{j-2} - u_{j-1} q_j) + b(v_{j-2} - v_{j-1} q_j)$

$= a u_j + b v_j$.

Ex. We continue the example

$a = 81 \quad b = 64$

Row	R	Q	U	V
-1	81	-	1	0
0	64	-	0	1
1	17	1	1	-1
2	13	3	-3	4
3	4	1	4	-5
4	1	3	-15	19

$\rightarrow \quad 1 = 81(-15) + 64(19)$

$\rightarrow \quad 64 \cdot 19 \equiv 1 \bmod 81$

19 is the inverse of 64 mod 81

It is possible that the final v_n is negative, giving a negative inverse. Simply use the congruence relation (add enough multiples of a) to v_n to get a positive inverse

III A result of Fermat

Thm. If p is a prime and p does not divide e, then

$$e^{p-1} \equiv 1 \bmod p$$

Proof. Let $S = \{1, \ldots, p-1\}$

$eS = \{e, 2e, \ldots (p-1)e\} \bmod p$

If $ei \equiv ej \bmod p$ then p divides $e(i-j)$

$\rightarrow$ p divides $i-j$

$\rightarrow$ $i \equiv j \bmod p$

So $eS \bmod p$ just rearranges the elements in S. So the product of all the elements in S equals the product of all the elements in $eS \bmod p$

$\rightarrow$ $(p-1)! = e^{p-1}(p-1)! \bmod p$

$\rightarrow$ p divides $(e^{p-1}-1)(p-1)!$ ~~mod~~

$\rightarrow$ p divides $e^{p-1}-1$

$\rightarrow$ $e^{p-1} \equiv 1 \bmod p$

Thm. Suppose $\gcd(e, p-1) = 1$ Then there is a d such that $ed \equiv 1 \bmod (p-1)$

Then $a^{ed} \equiv a^{1+k(p-1)} = a \bmod p$

Proof When $\gcd(a, p) = 1$,

$$a^{ed} = a^{1+k(p-1)} = a\,(a^{p-1})^k = a \cdot 1 = a \bmod p$$

If $\gcd(a, p) \neq 1$, then $\gcd(a, p) = p$ and p divides a

So $a^{ed} \bmod p = 0 \bmod p = a \bmod p$

These facts allow the existence of a cipher, the exponential cipher, a predecessor of the RSA. It goes as follows. Alice and Bob pick a prime p and e with $\gcd(e, p) = 1$ They compute d such that $ed \equiv 1 \bmod p$ Bob wants to send a message a to Alice. Bob computes $a^e \bmod p = b$ and sends b to Alice who computes $b^d = a^{ed} \bmod p = a$ and gets the message. This is a private key system. For if Eve knows e and p, she then knows $p-1$ and can find d from

the Euclidean algorithm for $a = p-1$ and $b = e$.

Ex Bob ~~wants~~ and Alice pick $p = 181$, $e = 97$. They compute $d = 28$ from the Euclidean algorithm.
ie $ed \equiv 1 \bmod 180$

Bob is to send

B E T R A Y A L

to Alice. This converts to

1 4 19 17 0 24 0 11

Each of these is a in $a^{97} \bmod 181$. ie $19^{97} \bmod 181$ for T

Computing each gives

1 94 19 92 0 158 0 118

which is send to Alice who computes $b^d \bmod 181$ $(d = 28)$

This will give the original numbers which she converts to English. There is a lot of computation and a computer would be used. But it uses the repeated squaring algorithm discussed in the beginning of the computation section