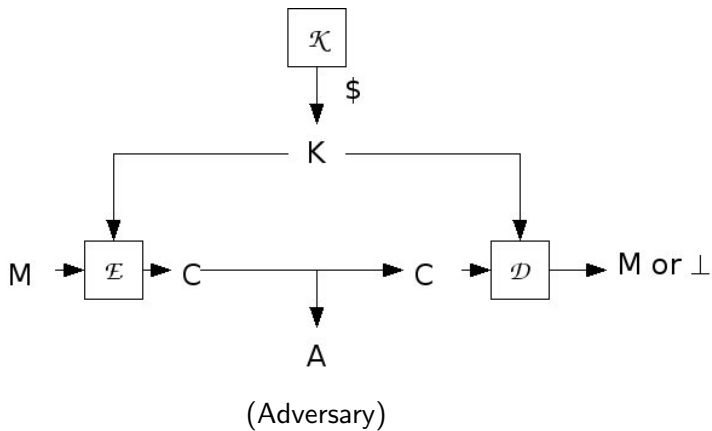


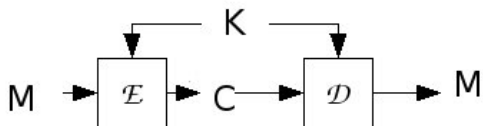
# CLASSICAL ENCRYPTION

# Syntax

A symmetric encryption scheme  $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$  consists of three algorithms:



# Correct decryption requirement



For all  $K, M$  we have

$$\mathcal{D}_K(\mathcal{E}_K(M)) = M$$

# Terminology recall

Alphabets:

- $\Sigma_1 = \{A, B, C, \dots, Z\}$
- $\Sigma_2 = \{A, B, C, \dots, Z\} \cup \{\sqcup, ., ?, \dots\}$
- $\Sigma_3 = \{0, 1\}$

Strings:

- Over  $\Sigma_1$ : HELLO, BZYK, ...
- Over  $\Sigma_2$ : HOW  $\sqcup$  ARE  $\sqcup$  YOU?
- Over  $\Sigma_3$ : 01101

Denote by  $\Sigma^*$  the set of all strings over alphabet  $\Sigma$ :

- $\{A, B, \dots, Z\}^*$
- $\{0, 1\}^*$

# Length and size

If  $s$  is a string then  $|s|$  is the number of symbols in it:

- $|\text{HELLO}| = 5$
- $|\text{HOW } \square \text{ ARE } \square \text{ YOU?}| =$

# Length and size

If  $s$  is a string then  $|s|$  is the number of symbols in it:

- $|\text{HELLO}| = 5$
- $|\text{HOW } \square \text{ ARE } \square \text{ YOU?}| = 12$

# Length and size

If  $s$  is a string then  $|s|$  is the number of symbols in it:

- $|\text{HELLO}| = 5$
- $|\text{HOW } \sqcup \text{ ARE } \sqcup \text{ YOU?}| = 12$
- $|01101| = 5$

We denote by  $s[i]$  the  $i$ -th symbol of string  $s$ :

- $s[3] = L$  if  $s = \text{HELLO}$
- $s[5] = A$  if  $s = \text{HOW } \sqcup \text{ ARE } \sqcup \text{ YOU?}$
- $s[2] = 1$  if  $s = 01101$

If  $S$  is a set then  $|S|$  is its size:

- $|\{A, B, \dots, Z\}| = 26$
- $|\{0, 1\}^8| =$

# Length and size

If  $s$  is a string then  $|s|$  is the number of symbols in it:

- $|\text{HELLO}| = 5$
- $|\text{HOW } \sqcup \text{ ARE } \sqcup \text{ YOU?}| = 12$
- $|\text{01101}| = 5$

We denote by  $s[i]$  the  $i$ -th symbol of string  $s$ :

- $s[3] = L$  if  $s = \text{HELLO}$
- $s[5] = A$  if  $s = \text{HOW } \sqcup \text{ ARE } \sqcup \text{ YOU?}$
- $s[2] = 1$  if  $s = \text{01101}$

If  $S$  is a set then  $|S|$  is its size:

- $|\{A, B, \dots, Z\}| = 26$
- $|\{0, 1\}^8| = 2^8$

Then notation  $\pi : D \rightarrow R$  means  $\pi$  is a map (function) with

- inputs drawn from the set  $D$  (the domain)
- outputs falling in the set  $R$  (the range)

**Example:** Define  $\pi : \{1, 4, 6\} \rightarrow \{0, 1\}$  by

$x$	1	4	6
$\pi(x)$	1	1	0

Functions can be specified as above or sometimes by code.

**Example:** The above can also be specified by

**Alg**  $\pi(x)$

Return  $x \bmod 3$

# Permutations

A map (function)  $\pi : S \rightarrow S$  is a permutation if it is one-to-one. Equivalently, it has an inverse map  $\pi^{-1} : S \rightarrow S$ .

**Example:**  $S = \{A, B, C\}$

A permutation and its inverse:

$x$	A	B	C
$\pi(x)$	C	A	B

$y$	A	B	C
$\pi^{-1}(y)$	B	C	A

Not a permutation:

$x$	A	B	C
$\pi(x)$	C	B	B

# Counting permutations

There are many different possible permutations  $\pi: S \rightarrow S$  on a given set  $S$ . How many?

**To be specific:** How many permutations  $\pi: S \rightarrow S$  are there on the set  $S = \{A, B, C\}$ ?

# Counting permutations

There are many different possible permutations  $\pi: S \rightarrow S$  on a given set  $S$ . How many?

**To be specific:** How many permutations  $\pi: S \rightarrow S$  are there on the set  $S = \{A, B, C\}$ ?

**Answer:**  $3! = 3 * 2 * 1 = 6$

$x$	$\pi(x)$	
A		← 3 choices: A, B, C
B		← 2 choices: not $\pi(A)$
C		← 1 choice: not $\pi(A), \pi(B)$

In general there are  $|S|!$  permutations  $\pi: S \rightarrow S$ .

We let  $\text{Perm}(S)$  denote the set of all these permutations.

# Substitution ciphers

- Alphabet  $\Sigma$
- Key is a permutation  $\pi : \Sigma \rightarrow \Sigma$  defining the encoding rule
- Plaintext  $M \in \Sigma^*$  is a string over  $\Sigma$
- Encryption of  $M = M[1] \cdots M[n]$  is

$$C = \pi(M[1]) \cdots \pi(M[n])$$

- Decryption of  $C = C[1] \cdots C[n]$  is

$$M = \pi^{-1}(C[1]) \cdots \pi^{-1}(C[n])$$

# Substitution ciphers

A substitution cipher over alphabet  $\Sigma$  is a symmetric encryption scheme  $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$  in which the key output by  $\mathcal{K}$  is a permutation  $\pi : \Sigma \rightarrow \Sigma$ , and

Algorithm  $\mathcal{E}_\pi(M)$

For  $i = 1, \dots, |M|$  do

$C[i] \leftarrow \pi(M[i])$

Return  $C$

Algorithm  $\mathcal{D}_\pi(C)$

For  $i = 1, \dots, |C|$  do

$M[i] \leftarrow \pi^{-1}(C[i])$

Return  $M$

# Setup for Examples

$$\Sigma = \{A, B, \dots, Z\} \cup \{\sqcup, ., ?, !, \dots\}$$

Plaintexts are members of  $\Sigma^*$ , which means any English text (sequence of sentences) is a plaintext.

For simplicity we only consider permutations that are punctuation respecting:

$$\pi(\sqcup) = \sqcup \quad , \quad \pi(.) = . \quad , \quad \pi(?) = ? \quad , \quad \dots$$

so punctuation is left unchanged by encryption.

# Example

$\sigma$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi(\sigma)$	B	U	P	W	I	Z	L	A	F	N	S	G	K
$\sigma$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi(\sigma)$	D	H	T	J	X	C	M	Y	O	V	E	Q	R

Then encryption of plaintext  $M = \text{HI THERE}$  is

$$C = \pi(\text{H})\pi(\text{I})\pi(\square)\pi(\text{T})\pi(\text{H})\pi(\text{E})\pi(\text{R})\pi(\text{E}) = \text{AF MAIXI}$$

$\tau$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$	H	A	S	N	X	I	L	O	E	Q	M	G	T
$\tau$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$	J	V	C	Y	Z	K	P	B	W	D	R	U	F

Decryption of ciphertext  $C = \text{AF MAIXI}$  is

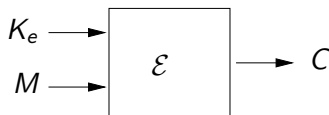
$$\pi^{-1}(\text{A})\pi^{-1}(\text{F})\pi^{-1}(\square)\pi^{-1}(\text{M})\pi^{-1}(\text{A})\pi^{-1}(\text{I})\pi^{-1}(\text{X})\pi^{-1}(\text{I}) = \text{HI THERE}$$

Basic adversary goal is plaintext recovery: given ciphertext  $C$  it aims to compute  $M = \mathcal{D}(\pi, C)$ .

This is easy if adversary knows  $\pi$  (hence  $\pi^{-1}$ ), but adversary is not given the key  $\pi$ .

However it does know what encryption scheme is used. (Meaning, in this case, a substitution cipher.)

# Kerchoff's principle



In some cases, designers hope to get security by keeping the description of the encryption procedure  $\mathcal{E}$  private.

But this prohibits standardization and usage. And it tends not to add to security since adversaries are remarkably good at reverse engineering a description of  $\mathcal{E}$  from its executable.

**Example:** RC4 and “alleged-RC4”

Good design (Kerchoff's principle):

- Adversary knows system
- The only thing it doesn't know is the key in use

# Cryptanalysis of a substitution cipher

Adversary has a ciphertext

```
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU  
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI  
OX PTI.
```

Exploit structure of English: In typical text

- E is the most common letter
- Next are T, A, O, I, N, S, H, R

A letter by itself (like T in ciphertext) can only be A or I.

Etc.

# Frequency counts

COXBX TBX CVK CDGXR DI T GTI'R A DHX VOXI OX ROKQA  
U IKC RNXPQA TCX: VOXI OX PTI'C THHKBU DC, TIU  
VOXI OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3												
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

# Frequency counts

COXB X TB X CVK CDGXR DI T GTI'R ADHX VOXI OX  
ROKQAU IKC RNX PQATCX: VOXI OX PTI'C THKBU DC,  
TIU VOXI OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3											
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

# Frequency counts

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU  
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI  
OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3	7	4	0	0	2	3	9	0	4	0	0

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	8	3	2	4	0	8	3	4	0	13	0	0

# Cryptanalysis

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

OX PTI:

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3	7	4	0	0	2	3	9	0	4	0	0

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	8	3	2	4	0	8	3	4	0	13	0	0

$\tau$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$													

$\tau$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$											E		

# Cryptanalysis

$E$   $E$   $E$   $E$   $E$   $E$   $E$   $E$   $E$   $E$   $E$   $E$   $E$   
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

$E$   $E$ :  $E$   $E$   $E$   $E$   $E$   $E$   $E$   $E$   
IKC RNXRPQATCX: VOXI OX PTI'C THHKBU DC', TIU VOXI

$E$   
OX PTI'.

$\tau$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$													

$\tau$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H									E		

OX in ciphertext  $\Rightarrow \pi^{-1}(0) \in \{B, H, M, W\}$

Guess  $\pi^{-1}(0) = H$  since O has pretty high frequency

HE E E E , E HE HE H  
 COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

E E: HE HE , HE  
 IKC RNX PQATCX: VOXI OX PTI'C THHKBU DC', TIU VOXI

HE  
 OX PTI.

$\tau$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$													

$\tau$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H									E		

HE E E E ' E HE HE H  
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU  
E E: HE HE ' HE  
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI  
HE .  
OX PTI.

\*HE\*E  
COXBX

Could be: THERE,THESE,WHERE,...

Guess  $\pi^{-1}(C) = T$  since there is no ? in ciphertext so WHERE is unlikely.

So  $\pi^{-1}(B) \in \{R,S\}$

# Cryptanalysis

THE E E T T E ' E HE HE H  
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

T E TE: HE HE 'T T, HE  
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE  
OX PTI.

$\tau$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$			T										

$\tau$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H									E		

# Cryptanalysis

THE E E T T E , E HE HE H  
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

T E TE: HE HE 'T T, HE  
IKC RNX PQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE  
OX PTI.

$\tau$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$			T										

$\tau$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H									E		

T is a single-letter word so  $\pi^{-1}(T) \in \{A, I\}$

We know  $\pi^{-1}(B) \in \{R, S\}$

So TBX could be: ARE, ASE, IRE, ISE

We guess ARE

# Cryptanalysis

THERE ARE T T E A A ' E HE HE H  
COVBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

T E ATE: HE HE A 'T A R T, A HE  
IKC RNX PQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE A .  
OX PTI.

$\tau$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$		R	T										
$\tau$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H					A				E		

# Cryptanalysis

THERE ARE T T E A A ' E HE HE H  
COVBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

T E ATE: HE HE A 'T A R T, A HE  
IKC RNX PQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE A .  
OX PTI.

$\tau$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$		R	T										
$\tau$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H					A				E		

\*T  
DC

D must be: A or I but T is A so D is I.

Etc....!

THERE ARE TWO TIMES IN A MAN'S LIFE WHEN HE SHOULD  
 COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

NOT SPECULATE: WHEN HE CAN'T AFFORD IT, AND WHEN  
 IKC RNX PQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE CAN.  
 OX PTI.

$\tau$	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$	L	R	T	I			M	F	N		O		
$\tau$	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$	P	H	C	U	S		A	D	W		E		

# Assessment of security of substitution ciphers

Defenders may argue

- Cryptanalysis requires long ciphertext
- Harder if  $\pi$  is not punctuation-respecting

In fact substitution ciphers or variations and enhancements have been almost universally used until relatively recently.

Yet they are fundamentally flawed.

# Voting

Vote Y or N for

- Proposition 107
- Bush or Gore

Voters  $V_1, V_2, V_3, V_4, V_5$  cast votes at polling station.

Example votes: YNYYN

Polling station

$$\xrightarrow{\pi(Y)\pi(N)\pi(Y)\pi(Y)\pi(N)}$$

Tally center

Is this secure?

Say  $\pi(Y) = A$  and  $\pi(N) = B$ . Adversary sees

$$\pi(Y)\pi(N)\pi(Y)\pi(Y)\pi(N) = ABAAB$$

Adversary can infer relations:  $V_1, V_3$  had same vote.

Adversary might be  $V_1$

- It knows its own vote is Y
- So given ciphertext ABAAB it infers that A represents Y
- But then B must represent N
- Adversary knows everyone's vote!

# The weakness

The weakness of a substitution cipher exploited above is simply that the same symbol is always encoded in the same way.

Attack does not require long plaintexts, and does not need  $\pi$  to be punctuation-respecting.

# What happened?

Critical security thinking yielded a scenario where substitution ciphers fail miserably:

- Few possible plaintext symbols (Y or N)
- Adversary is one of the users (voters)

# What did we learn?

- Security depends on usage
- Evaluating security requires being creative about coming up with usage scenarios that test the scheme

# Good cryptography

A good scheme is one that

- Is secure in ALL (reasonable) scenarios
- Is secure regardless of what type of data (e.g., Y,N strings) is being encrypted
- Even if adversary knows some decryptions, it shouldn't be able to produce others.

# One time pad

Key  $K \xleftarrow{\$} \{0, 1\}^m$  is a random  $m$ -bit string

Plaintext  $M \in \{0, 1\}^m$  is an  $m$ -bit string

Algorithm $\mathcal{E}_K(M)$	Algorithm $\mathcal{D}_K(C)$
$C \leftarrow K \oplus M$	$M \leftarrow K \oplus C$
Return $C$	Return $M$

Assume only a single message  $M$  is ever encrypted under one key.

Represent Y by 1 and N by 0

Voters  $V_1, \dots, V_m$  cast votes 1, 0, 1, 1, 0, ...

Let  $M = 10110\dots$

Encryption is  $C = K \oplus M$

Adversary has  $C$  but **NOT**  $K$

Adversary cannot tell whether two people have same vote.

Even if adversary is  $V_1$  and knows its own vote is 1, it cannot determine votes of other parties.

# A measure of security

Let  $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$  be a symmetric encryption scheme. For any message  $M$  and ciphertext  $C$  we are interested in

$$\Pr[\mathcal{E}_K(M) = C]$$

where the probability is over the random choice  $k \xleftarrow{\$} \mathcal{K}$  and over the coins tossed by  $\mathcal{E}$  if any.

# Example

		Messages:			
		00	01	10	11
Keys:	00	01	10	11	00
	01	01	11	10	00
	10	00	11	01	11
	11	11	10	01	11

The table entry in row  $K$  and column  $M$  is  $\mathcal{E}_K(M)$ .

- $\Pr[\mathcal{E}_K(00) = 01] =$

# Example

		Messages:			
		00	01	10	11
Keys:	00	01	10	11	00
	01	01	11	10	00
	10	00	11	01	11
	11	11	10	01	11

The table entry in row  $K$  and column  $M$  is  $\mathcal{E}_K(M)$ .

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{2}{4} = \frac{1}{2}$
- $\Pr[\mathcal{E}_K(01) = 01] =$

# Example

		Messages:			
		00	01	10	11
Keys:	00	01	10	11	00
	01	01	11	10	00
	10	00	11	01	11
	11	11	10	01	11

The table entry in row  $K$  and column  $M$  is  $\mathcal{E}_K(M)$ .

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{2}{4} = \frac{1}{2}$
- $\Pr[\mathcal{E}_K(01) = 01] = 0$
- $\Pr[\mathcal{E}_k(10) = 11] =$

# Example

		Messages:			
		00	01	10	11
Keys:	00	01	10	11	00
	01	01	11	10	00
	10	00	11	01	11
	11	11	10	01	11

The table entry in row  $K$  and column  $M$  is  $\mathcal{E}_K(M)$ .

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{2}{4} = \frac{1}{2}$
- $\Pr[\mathcal{E}_K(01) = 01] = 0$
- $\Pr[\mathcal{E}_k(10) = 11] = \frac{1}{4}$

# Perfect security

**Definition:** Let  $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$  be a symmetric encryption scheme. We say that  $\mathcal{SE}$  is **perfectly secure** if for any two messages  $M_1, M_2 \in \text{Plaintexts}$  and any  $C$

$$\Pr[\mathcal{E}_K(M_1) = C] = \Pr[\mathcal{E}_K(M_2) = C] .$$

In both cases, the probability is over the random choice  $K \xleftarrow{\$} \mathcal{K}$  and over the coins tossed by  $\mathcal{E}$  if any.

**Intuitively:** Given  $C$ , and even knowing the message is either  $M_1$  or  $M_2$  the adversary cannot determine which.

# Perfect security

Definition requires that

For all  $M_1, M_2, C$  we have

$$\Pr[\mathcal{E}_K(M_1) = C] = \Pr[\mathcal{E}_K(M_2) = C] .$$

If we want to show the definition is **not** met, we need to show that

There exists  $M_1, M_2, C$  such that

$$\Pr[\mathcal{E}_K(M_1) = C] \neq \Pr[\mathcal{E}_K(M_2) = C] .$$

# Example

		Messages:			
		00	01	10	11
Keys:	00	01	10	11	00
	01	01	11	10	00
	10	00	11	01	11
	11	11	10	01	11

The table entry in row  $K$  and column  $M$  is  $\mathcal{E}_K(M)$ .

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{2}{4} = \frac{1}{2}$
- $\Pr[\mathcal{E}_K(01) = 01] = 0$

Is this encryption scheme perfectly secure? **No**, because for  $M_1 = 00$ ,  $M_2 = 01$  and  $C = 01$  we have

$$\underbrace{\Pr[\mathcal{E}_K(M_1) = C]}_{1/2} \neq \underbrace{\Pr[\mathcal{E}_K(M_2) = C]}_0 .$$

# Perfect security of substitution ciphers

A substitution cipher is **NOT** perfectly secure.

Formally:

**Claim:** Let  $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$  be a substitution cipher over the alphabet  $\Sigma$  consisting of the 26 English letters. Assume that  $\mathcal{K}$  picks a random permutation over  $\Sigma$  as the key. That is, its code is

$$\pi \xleftarrow{\$} \text{Perm}(\Sigma) ; \text{ return } \pi .$$

Let Plaintexts be the set of all three letter English words. Then  $\mathcal{SE}$  is *not* perfectly secure.

# Proof of claim

To show: there exist  $M_1, M_2, C \in \Sigma^3$  such that

$$\Pr[\mathcal{E}_\pi(M_1) = C] \neq \Pr[\mathcal{E}_\pi(M_2) = C] .$$

We have replaced  $K$  with  $\pi$  because the key here is a permutation.

# Proof of claim

To show: there exist  $M_1, M_2, C \in \Sigma^3$  such that

$$\Pr[\mathcal{E}_\pi(M_1) = C] \neq \Pr[\mathcal{E}_\pi(M_2) = C] .$$

We have replaced  $K$  with  $\pi$  because the key here is a permutation.

Let

- $C = \text{XYY}$
- $M_1 = \text{FEE}$
- $M_2 = \text{FAR}$

# Proof of claim

To show: there exist  $M_1, M_2, C \in \Sigma^3$  such that

$$\Pr[\mathcal{E}_\pi(M_1) = C] \neq \Pr[\mathcal{E}_\pi(M_2) = C] .$$

We have replaced  $K$  with  $\pi$  because the key here is a permutation.

Let

- $C = \text{XYY}$
- $M_1 = \text{FEE}$
- $M_2 = \text{FAR}$

Then

$$\Pr[\mathcal{E}_\pi(M_2) = C] = \Pr[\pi(\text{F})\pi(\text{A})\pi(\text{R}) = \text{XYY}]$$

# Proof of claim

To show: there exist  $M_1, M_2, C \in \Sigma^3$  such that

$$\Pr[\mathcal{E}_\pi(M_1) = C] \neq \Pr[\mathcal{E}_\pi(M_2) = C] .$$

We have replaced  $K$  with  $\pi$  because the key here is a permutation.

Let

- $C = \text{XYY}$
- $M_1 = \text{FEE}$
- $M_2 = \text{FAR}$

Then

$$\begin{aligned}\Pr[\mathcal{E}_\pi(M_2) = C] &= \Pr[\pi(\text{F})\pi(\text{A})\pi(\text{R}) = \text{XYY}] \\ &= 0\end{aligned}$$

Because  $\pi(\text{A})$  cannot equal  $\pi(\text{R})$

$$\begin{aligned}\Pr[\mathcal{E}_\pi(M_1) = C] &= \Pr[\mathcal{E}_\pi(\text{FEE}) = \text{XYY}] \\ &= \frac{|\{ \pi \in \text{Perm}(\Sigma) : \mathcal{E}_\pi(\text{FEE}) = \text{XYY} \}|}{|\text{Perm}(\Sigma)|} \\ &= \frac{|\{ \pi \in \text{Perm}(\Sigma) : \pi(\text{F})\pi(\text{E})\pi(\text{E}) = \text{XYY} \}|}{|\text{Perm}(\Sigma)|}\end{aligned}$$

$$\begin{aligned}\Pr[\mathcal{E}_\pi(M_1) = C] &= \Pr[\mathcal{E}_\pi(\text{FEE}) = \text{XYY}] \\ &= \frac{|\{ \pi \in \text{Perm}(\Sigma) : \mathcal{E}_\pi(\text{FEE}) = \text{XYY} \}|}{|\text{Perm}(\Sigma)|} \\ &= \frac{|\{ \pi \in \text{Perm}(\Sigma) : \pi(\text{F})\pi(\text{E})\pi(\text{E}) = \text{XYY} \}|}{|\text{Perm}(\Sigma)|} \\ &= \frac{24!}{26!} \\ &= \frac{1}{650} .\end{aligned}$$

# Summary

**Definition:** Let  $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$  be a symmetric encryption scheme. We say that  $\mathcal{SE}$  is *perfectly secure* if for any two messages  $M_1, M_2 \in \text{Plaintexts}$  and any  $C$

$$\Pr[\mathcal{E}_K(M_1) = C] = \Pr[\mathcal{E}_K(M_2) = C] .$$

**Claim:** Let  $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$  be a substitution cipher over the alphabet  $\Sigma$  consisting of the 26 English letters. Assume that  $\mathcal{K}$  picks a random permutation over  $\Sigma$  as the key. Let Plaintexts be the set of all three letter English words. Then  $\mathcal{SE}$  is *not* perfectly secure.

We have proved the claim by presenting  $M_1, M_2, C$  such that

$$\Pr[\mathcal{E}_K(M_1) = C] \neq \Pr[\mathcal{E}_K(M_2) = C] .$$

# Intuition for OTP security

$$\mathcal{E}_K(M) = K \oplus M$$

Suppose adversary gets ciphertext  $C = 101$  and knows the plaintext  $M$  is either  $M_1 = 010$  or  $M_2 = 001$ . Can it tell which?

No, because  $C = K \oplus M$  so

- $M = 010$  iff  $K = 111$
- $M = 001$  iff  $K = 110$

but  $K$  is equally likely to be 111 or 110 and adversary does not know  $K$ .

# Perfect security of OTP

**Claim:** Let  $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$  be the OTP scheme with key-length  $m \geq 1$ . Then  $\mathcal{SE}$  is perfectly secure.

Want to show that for any  $M_1, M_2, C$

$$\Pr[\mathcal{E}_K(M_1) = C] = \Pr[\mathcal{E}_K(M_2) = C]$$

That is

$$\Pr[K \oplus M_1 = C] = \Pr[K \oplus M_2 = C]$$

when  $K \xleftarrow{\$} \{0, 1\}^m$ .

## Example: $m = 2$

Messages:

	00	01	10	11
00	00	01	10	11
01	01	00	11	10
10	10	11	00	01
11	11	10	01	00

Keys:

The entry in row  $K$ , column  $M$  of the table is  $\mathcal{E}_K(M) = K \oplus M$ .

- $\Pr[\mathcal{E}_K(00) = 01] =$

## Example: $m = 2$

Messages:

	00	01	10	11
00	00	01	10	11
01	01	00	11	10
10	10	11	00	01
11	11	10	01	00

Keys:

The entry in row  $K$ , column  $M$  of the table is  $\mathcal{E}_K(M) = K \oplus M$ .

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{1}{4}$
- $\Pr[\mathcal{E}_K(10) = 01] =$

## Example: $m = 2$

Messages:

	00	01	10	11
00	00	01	10	11
01	01	00	11	10
10	10	11	00	01
11	11	10	01	00

Keys:

The entry in row  $K$ , column  $M$  of the table is  $\mathcal{E}_K(M) = K \oplus M$ .

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{1}{4}$
- $\Pr[\mathcal{E}_K(10) = 01] = \frac{1}{4}$

$$\Pr[\mathcal{E}_K(M_1) = C] = \Pr[K \oplus M_1 = C]$$

$$\begin{aligned}\Pr[\mathcal{E}_K(M_1) = C] &= \Pr[K \oplus M_1 = C] \\ &= \frac{|\{K \in \{0,1\}^m : K \oplus M_1 = C\}|}{|\{0,1\}^m|}\end{aligned}$$

$$\begin{aligned}\Pr[\mathcal{E}_K(M_1) = C] &= \Pr[K \oplus M_1 = C] \\ &= \frac{|\{K \in \{0,1\}^m : K \oplus M_1 = C\}|}{|\{0,1\}^m|} \\ &= \frac{1}{2^m}.\end{aligned}$$

$$\begin{aligned}\Pr[\mathcal{E}_K(M_2) = C] &= \Pr[K \oplus M_2 = C] \\ &= \frac{|\{K \in \{0,1\}^m : K \oplus M_2 = C\}|}{|\{0,1\}^m|} \\ &= \frac{1}{2^m}.\end{aligned}$$

# Perfect security: Plusses and Minuses

+

Very good privacy

-

Key needs to be as  
long as message

# What next

We want schemes to securely encrypt

- arbitrary amounts of data
- with a single, short (e.g., 128 bit) key

This will be possible once we relax our goal from perfect to computational security.

Plan:

- Study the primitives we will use, namely block ciphers
- Understand and define computational security of block ciphers and encryption schemes
- Use (computationally secure) block ciphers to build (computationally secure) encryption schemes