Crypto: Confidentiality

Applied Information Security
Lecture 7

"When Julius Caesar sent messages to his generals, he didn't trust his messengers…

so he replaced every A in his messages with a D, every B with an E, and so on through the alphabet. Only someone who knew the "shift by 3" rule could decipher his messages.

And so we begin."

- Introduction to Cryptography

Today's Topics

cryptography!

● history (before XOR)		
● perfect secrecy	OTP	one-time pad
● key generation	RNG	random number generator
● encrypt/decrypt		
○ a block	AES	advanced encryption standard
○ a stream of blocks	CBC	cipher block chaining
○ a stream	III	salsa20
● key exchange	DH	Diffie-Hellman

“Securely”

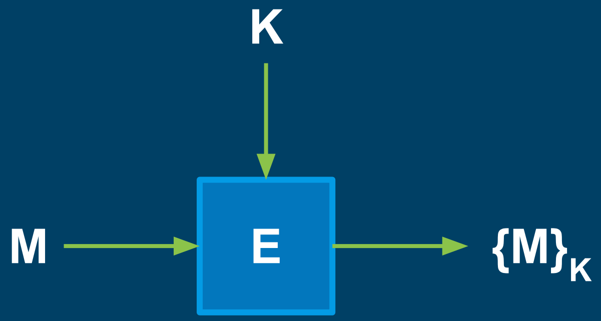
- Confidentiality:
  only the intended recipient of a message should be able to read it.

- Integrity:
  An adversary cannot (undetectedly) tamper with a message.

- Authenticity [new!]:
  An adversary cannot (undetectedly) forge a message from either party

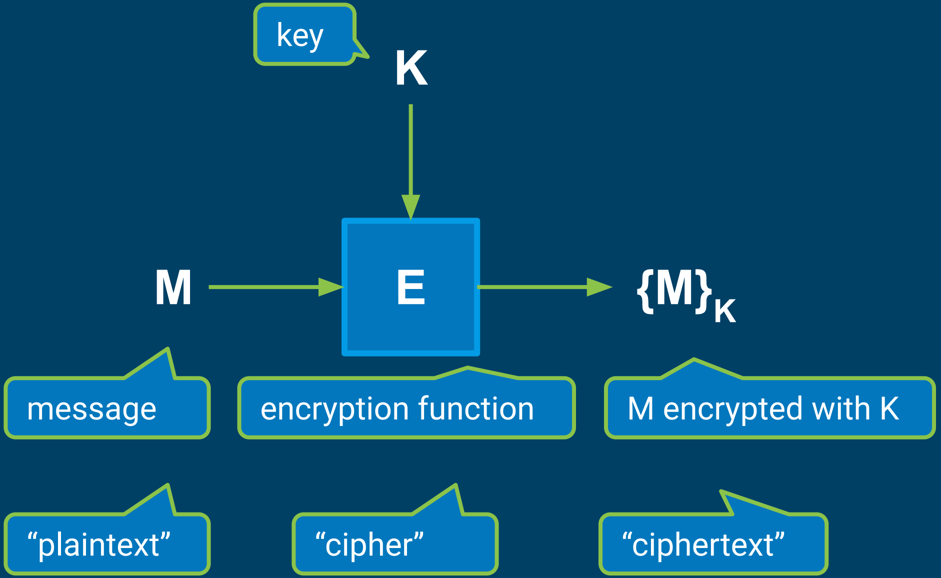
In Pictures: Symmetric-Key Cryptography

Encrypt



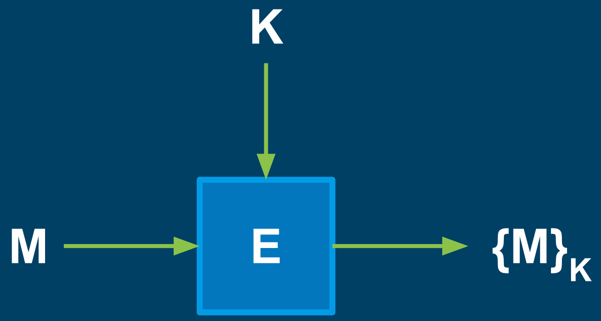
In Pictures: Symmetric-Key Cryptography

Encrypt

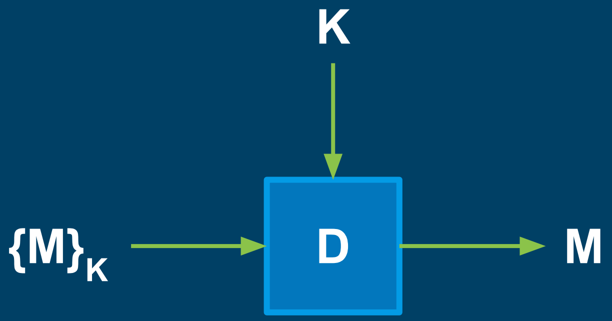


In Pictures: Symmetric-Key Cryptography

Encrypt



Decrypt



Encryption & decryption

- Encryption: function from *secret key* and *plaintext* to *ciphertext*
- Decryption: function from *secret key* and *ciphertext* to *plaintext*.
- Security depends on assumption that decryption is *infeasible* to compute when you don't know K.

Encryption/Decryption should be fast.

security / performance tradeoff

Kerckhoff's Principle: Security of encryption scheme depends only on K, not on E or D.
(why: compromised "dictionary" makes E lost forever)

Encryption

$E(K,M) = \{M\}_K$

Decryption

$D(K,\{M\}_K) = M$

Theorem

$D(K,E(K,M)) = M$

Assumption

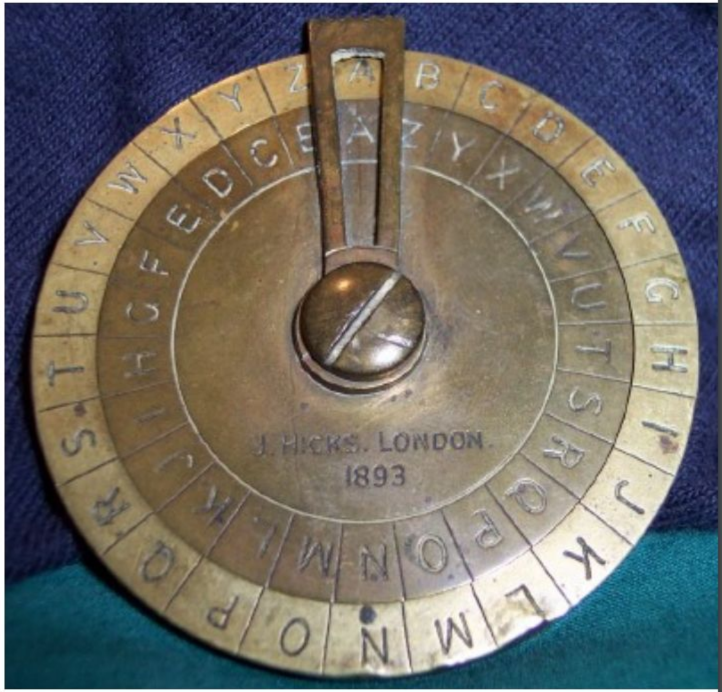
$D(-,\{M\}_K)$ is infeasible to compute when you don’t know K.

history

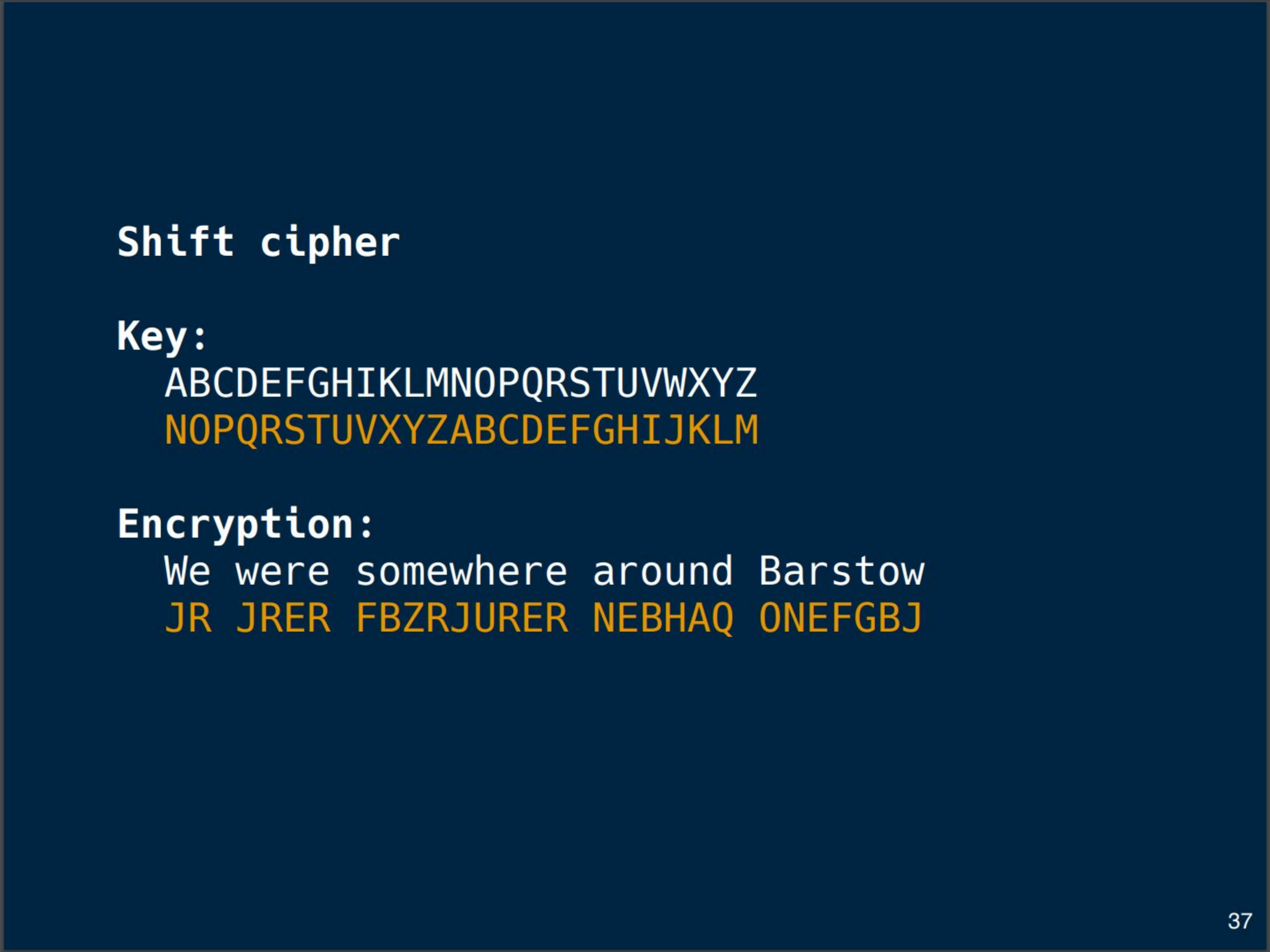
before XOR

Caesar-cipher

Aka “shift cipher”
Key is rotation of wheel.
Say, A becomes N.
Translate A -> N, B -> O, C -> P, …



history



Shift cipher

Key:

```
ABCDEFGHIKLMNOPQRSTUVWXYZ
NOPQRSTUVXYZABCDEFGHIJKLM
```

Encryption:

```
We were somewhere around Barstow
JR JRER FBZRJURER NEBHAQ ONEFGBJ
```

Shift cipher: Key-space is too small

```
iq iqdq eayqitqdq mdagzp nmdefai
hp hpcp dzxphspcp lczfyo mlcdezh
go gobo cywogrobo kbyexn lkbcdyg
fn fnan bxvnfqnan jaxdwm kjabcxf
em emzm awumepmzm izwcvl jizabwe
dl dlyl zvtldolyl hyvbuk ihyzavd
ck ckxk yuskcnkxk gxuatj hgxyzuc
bj bjwj xtrjbmjwj fwtzsi gfwxytb
ai aivi wsqialivi evsyrh fevwxsa
zh zhuh vrphzkhuh durxqg eduvwrz
yg ygtg uqogyjgtg ctqwpf dctuvqy
xf xfsf tpnfxifsf bspvoe cbstupx
we were somewhere around barstow
vd vdqd rnldvgdqd zqntmc azqrsnv
uc ucpc qmkcufcpc ypmslb zypqrmu
tb tbob pljbtebob xolrka yxopqlt
sa sana okiasdana wnkqjz xwnopks
rz rzmz njhzrczmz vmjpiy wvmnojr
qy qyly migyqbyly uliohx vulmniq
px pxkx lhfxpaxkx tkhngw utklmhp
ow owjw kgewozwjw sjgmfv tsjklgo
nv nviv jfdvnyviv rifleu srijkfn
mu muhu iecumxuhu qhekdt rqhijem
lt ltgt hdbtlwtgt pgdjcs qpghidl
```

how many keys are there?

key space size
= number of rotations
= size of latin alphabet
= 26

(2 are not depicted)

try decrypting with
each one!

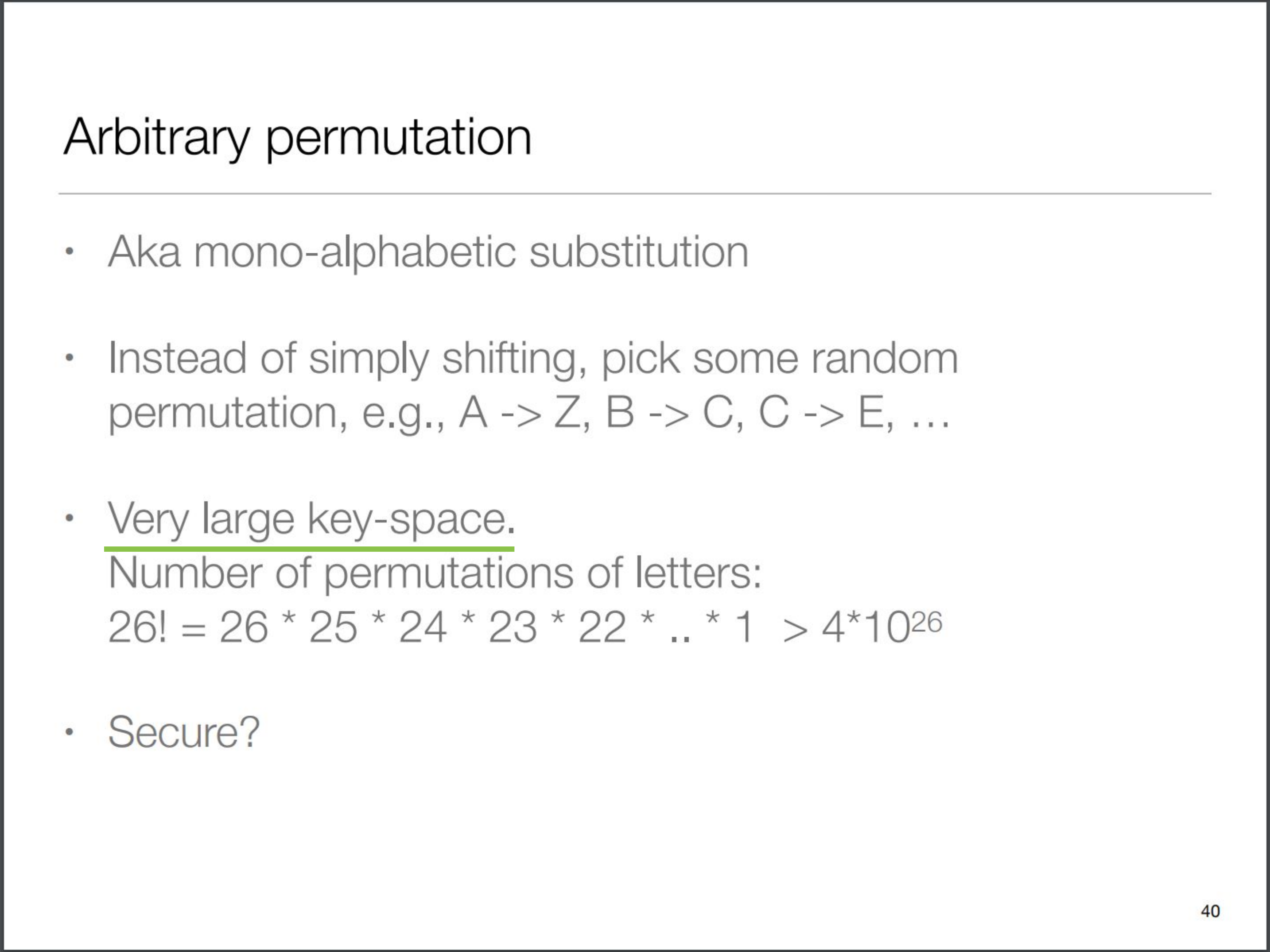
brute-force attack

history

Shift cipher: Key-space is too small

```
iq iqdq eayqitqdq mdagzp nmdefai
hp hpcp dzxphspcp lczfyo mlcdezh
go gobo cywogrobo kbyexn lkbcdyg
fn fnan bxvnfqnan jaxdwm kjabcxf
em emzm awumepmzm izwcvl jizabwe
dl dlyl zvtldolyl hyvbuk ihyzavd
ck ckxk yuskcnkxk gxuatj hgxyzuc
bj bjwj xtrjbmjwj fwtzsi gfwxytb
ai aivi wsqialivi evsyrh fevwxsa
zh zhuh vrphzkhuh durxqg eduvwrz
yg ygtg uqogyjgtg ctqwpf dctuvqy
xf xfsf tpnfxifsf bspvoe cbstupx
we were somewhere around barstow
vd vdqd rnldvgdqd zqntmc azqrsnv
uc ucpc qmkcufcpc ypmslb zypqrmu
tb tbob pljbtebob xolrka yxopqlt
sa sana okiasdana wnkqjz xwnopks
rz rzmz njhzrczmz vmjpiy wvmnojr
qy qyly migyqbyly uliohx vulmniq
px pxkx lhfxpaxkx tkhngw utklmhp
ow owjw kgewozwjw sjgmfv tsjklgo
nv nviv jfdvnyviv rifleu srijkfn
mu muhu iecumxuhu qhekdt rqhijem
lt ltgt hdbtlwtgt pgdjcs qpghidl
```

that looks readable.
the rest is not.



Arbitrary permutation

- Aka mono-alphabetic substitution
- Instead of simply shifting, pick some random permutation, e.g., A -> Z, B -> C, C -> E, …
- Very large key-space.
  Number of permutations of letters:
  26! = 26 * 25 * 24 * 23 * 22 * .. * 1 > 4*10^{26}
- Secure?

history

Mono-alphabetic substitution:

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Mono-alphabetic substitution: Vulnerable to statistical analysis

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
_ E T A O I N S R H D L U C M F Y W G P B V K X Q J Z   ENGLISH
```

Mono-alphabetic substitution: Vulnerable to statistical analysis

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
  U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
  _ E T A O I N S R H D L U C M F Y W G P B V K X Q J Z   ENGLISH
```

```
CE CEHE SOWECREHE AHOUND VAHSTOC ON TRE EDYE OB TRE DESEHT CREN TRE
DHUYS VEYAN TO TAJE ROLDF I HEWEWVEH SAGINY SOWETRINY LIJE XI BEEL A
VIT LIYRTREADED; WAGVE GOU SROULD DHIZEF F F FX AND SUDDENLG TREHE
CAS A TEHHIVLE HOAH ALL AHOUND US AND TRE SJG CAS BULL OB CRAT
LOOJED LIJE RUYE VATSQ ALL SCOOKINY AND SMHEEMRINY AND DIZINY AHOUND
TRE MAHQ CRIMR CAS YOINY AVOUT A RUNDHED WILES AN ROUH CITR TRE TOK
DOCN TO LAS ZEYASF
```

history

Mono-alphabetic substitution: Vulnerable to statistical analysis

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
  U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
  _ E T A O I N S R H D L U C M F Y W G P B V K X Q J Z   ENGLISH
```

```
CE CEHE SOWECREHE AHOUND VAHSTOC ON TRE EDYE OB TRE DESEHT CREN TRE
DHUYS VEYAN TO TAJE ROLDF I HEWEWVEH SAGINY SOWETRINY LIJE XI BEEL A
VIT LIYRTREADED; WAGVE GOU SROULD DHIZEF F F FX AND SUDDENLG TREHE
CAS A TEHHIVLE HOAH ALL AHOUND US AND TRE SJG CAS BULL OB CRAT
LOOJED LIJE RUYE VATSQ ALL SCOOKINY AND SMHEEMRINY AND DIZINY AHOUND
TRE MAHQ CRIMR CAS YOINY AVOUT A RUNDHED WILES AN ROUH CITR TRE TOK
DOCN TO LAS ZEYASF
```

Most common english trigram: THE

Mono-alphabetic substitution: Vulnerable to statistical analysis

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
  U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
  _ E T A O I N S H R D L U C M F Y W G P B V K X Q J Z   ENGLISH
```

```
CE CERE SOWECHERE AROUND VARSTOC ON THE EDYE OB THE DESERT CHEN THE
DRUYS VEYAN TO TAJE HOLDF I REWEWVER SAGINY SOWETHINY LIJE XI BEEL A
VIT LIYHTHEADED; WAGVE GOU SHOULD DRIZEF F F FX AND SUDDENLG THERE
CAS A TERRIVLE ROAR ALL AROUND US AND THE SJG CAS BULL OB CHAT
LOOJED LIJE HUYE VATSQ ALL SCOOKINY AND SMREEMHINY AND DIZINY AROUND
THE MARQ CHIMH CAS YOINY AVOUT A HUNDRED WILES AN HOUR CITH THE TOK
DOCN TO LAS ZEYASF
```

Mono-alphabetic substitution: Vulnerable to statistical analysis

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
  U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
  _ E T A O I N S H R D L U C M F Y W G P B V K X Q J Z   ENGLISH
```

```
CE CERE SOWECHERE AROUND VARSTOC ON THE EDYE OB THE DESERT CHEN THE
DRUYS VEYAN TO TAJE HOLDF I REWEWVER SAGINY SOWETHINY LIJE XI BEEL A
VIT LIYHTHEADED; WAGVE GOU SHOULD DRIZEF F F FX AND SUDDENLG THERE
CAS A TERRIVLE ROAR ALL AROUND US AND THE SJG CAS BULL OB CHAT
LOOJED LIJE HUYE VATSQ ALL SCOOKINY AND SMREEMHINY AND DIZINY AROUND
THE MARQ CHIMH CAS YOINY AVOUT A HUNDRED WILES AN HOUR CITH THE TOK
DOCN TO LAS ZEYASF
```

Long words/phrases with one error.

history

Mono-alphabetic substitution: Vulnerable to statistical analysis

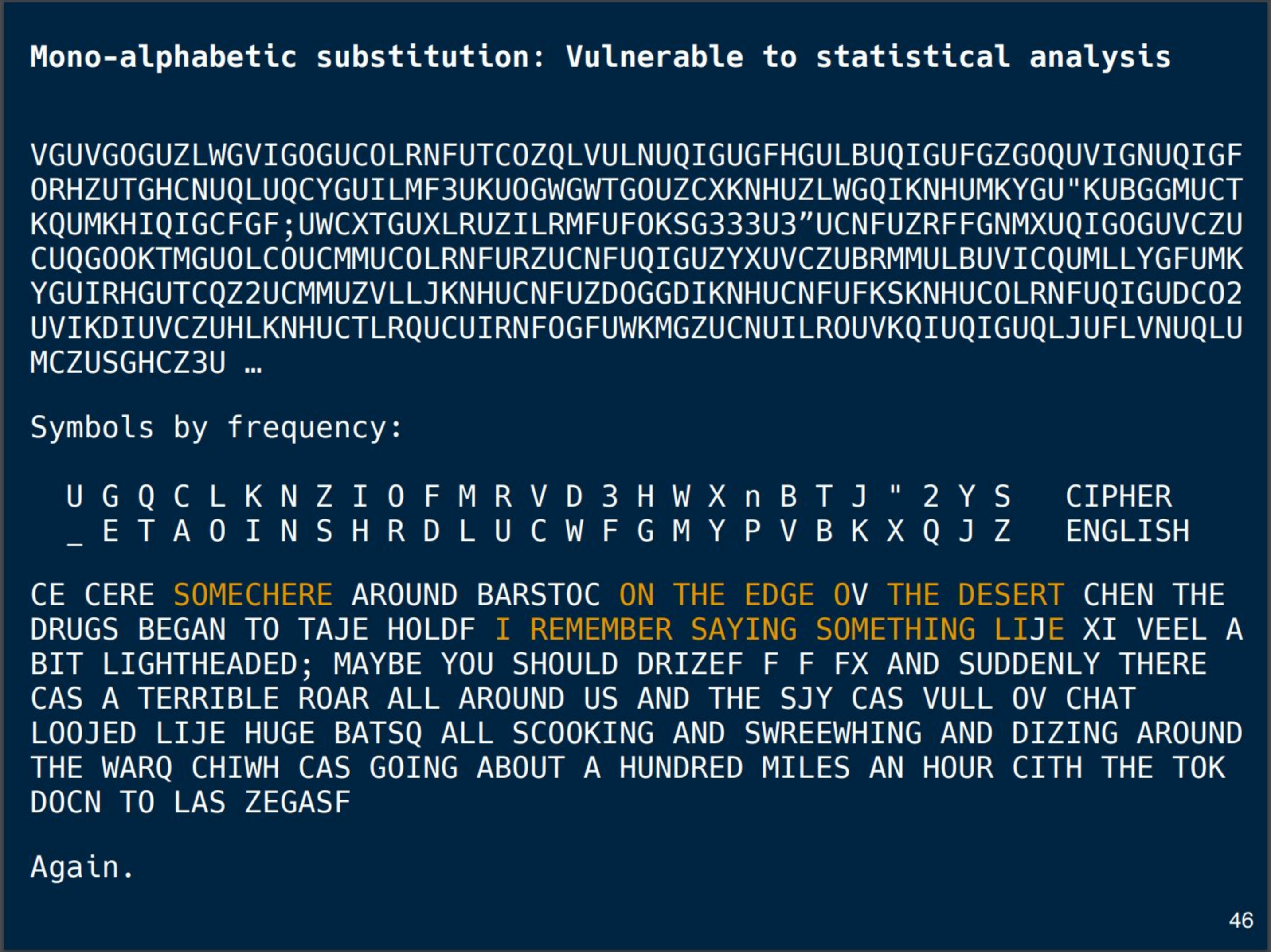
```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
  U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
  _ E T A O I N S H R D L U C W F G M Y P V B K X Q J Z   ENGLISH
```

```
CE CERE SOMECHERE AROUND BARSTOC ON THE EDGE OV THE DESERT CHEN THE
DRUGS BEGAN TO TAJE HOLDF I REMEMBER SAYING SOMETHING LIJE XI VEEL A
BIT LIGHTHEADED; MAYBE YOU SHOULD DRIZEF F F FX AND SUDDENLY THERE
CAS A TERRIBLE ROAR ALL AROUND US AND THE SJY CAS VULL OV CHAT
LOOJED LIJE HUGE BATSQ ALL SCOOKING AND SWREEWHING AND DIZING AROUND
THE WARQ CHIWH CAS GOING ABOUT A HUNDRED MILES AN HOUR CITH THE TOK
DOCN TO LAS ZEGASF
```

history



Mono-alphabetic substitution: Vulnerable to statistical analysis

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
  U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
  _ E T A O I N S H R D L U C W F G M Y P V B K X Q J Z   ENGLISH
```

CE CERE SOMECHERE AROUND BARSTOC ON THE EDGE OV THE DESERT CHEN THE DRUGS BEGAN TO TAJE HOLDF I REMEMBER SAYING SOMETHING LIJE XI VEEL A BIT LIGHTHEADED; MAYBE YOU SHOULD DRIZEF F F FX AND SUDDENLY THERE CAS A TERRIBLE ROAR ALL AROUND US AND THE SJY CAS VULL OV CHAT LOOJED LIJE HUGE BATSQ ALL SCOOKING AND SWREEWHING AND DIZING AROUND THE WARQ CHIWH CAS GOING ABOUT A HUNDRED MILES AN HOUR CITH THE TOK DOCN TO LAS ZEGASF

Again.

history

Mono-alphabetic substitution: Vulnerable to statistical analysis

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
  U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
  _ E T A O I N S H R D L U W C V G M Y P F B J X Q K Z   ENGLISH
```

```
WE WERE SOMEWHERE AROUND BARSTOW ON THE EDGE OF THE DESERT WHEN THE
DRUGS BEGAN TO TAKE HOLDV I REMEMBER SAYING SOMETHING LIKE XI FEEL A
BIT LIGHTHEADED; MAYBE YOU SHOULD DRIZEV V V VX AND SUDDENLY THERE
WAS A TERRIBLE ROAR ALL AROUND US AND THE SKY WAS FULL OF WHAT
LOOKED LIKE HUGE BATSQ ALL SWOOJING AND SCREECHING AND DIZING AROUND
THE CARQ WHICH WAS GOING ABOUT A HUNDRED MILES AN HOUR WITH THE TOJ
DOWN TO LAS ZEGASV
```

Mono-alphabetic substitution: Vulnerable to statistical analysis

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
  U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
  _ E T A O I N S H R D L U W C V G M Y P F B J X Q K Z   ENGLISH
```

```
WE WERE SOMEWHERE AROUND BARSTOW ON THE EDGE OF THE DESERT WHEN THE
DRUGS BEGAN TO TAKE HOLDV I REMEMBER SAYING SOMETHING LIKE XI FEEL A
BIT LIGHTHEADED; MAYBE YOU SHOULD DRIZEV V V VX AND SUDDENLY THERE
WAS A TERRIBLE ROAR ALL AROUND US AND THE SKY WAS FULL OF WHAT
LOOKED LIKE HUGE BATSQ ALL SWOOJING AND SCREECHING AND DIZING AROUND
THE CARQ WHICH WAS GOING ABOUT A HUNDRED MILES AN HOUR WITH THE TOJ
DOWN TO LAS ZEGASV
```

Final errors, punctuation.

Mono-alphabetic substitution: Vulnerable to statistical analysis

```
VGUVGOGUZLWGVIGOGUCOLRNFUTCOZQLVULNUQIGUGFHGULBUQIGUFGZGOQUVIGNUQIGF
ORHZUTGHCNUQLUQCYGUILMF3UKUOGWGWTGOUZCXKNHUZLWGQIKNHUMKYGU"KUBGGMUCT
KQUMKHIQIGCFGF;UWCXTGUXLRUZILRMFUFOKSG333U3”UCNFUZRFFGNMXUQIGOGUVCZU
CUQGOOKTMGUOLCOUCMMUCOLRNFURZUCNFUQIGUZYXUVCZUBRMMULBUVICQUMLLYGFUMK
YGUIRHGUTCQZ2UCMMUZVLLJKNHUCNFUZDOGGDIKNHUCNFUFKSKNHUCOLRNFUQIGUDCO2
UVIKDIUVCZUHLKNHUCTLRQUCUIRNFOGFUWKMGZUCNUILROUVKQIUQIGUQLJUFLVNUQLU
MCZUSGHCZ3U …
```

Symbols by frequency:

```
  U G Q C L K N Z I O F M R V D 3 H W X n B T J " 2 Y S   CIPHER
  _ E T A O I N S H R D L U W C . G M Y J F B P " , K V   ENGLISH
```

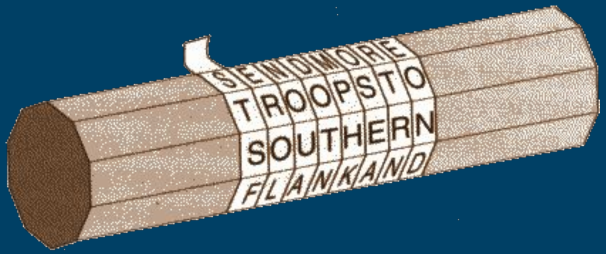
```
WE WERE SOMEWHERE AROUND BARSTOW ON THE EDGE OF THE DESERT WHEN THE
DRUGS BEGAN TO TAKE HOLD. I REMEMBER SAYING SOMETHING LIKE "I FEEL A
BIT LIGHTHEADED; MAYBE YOU SHOULD DRIVE. . . ." AND SUDDENLY THERE
WAS A TERRIBLE ROAR ALL AROUND US AND THE SKY WAS FULL OF WHAT
LOOKED LIKE HUGE BATS, ALL SWOOPING AND SCREECHING AND DIVING AROUND
THE CAR, WHICH WAS GOING ABOUT A HUNDRED MILES AN HOUR WITH THE TOP
DOWN TO LAS VEGAS.
```

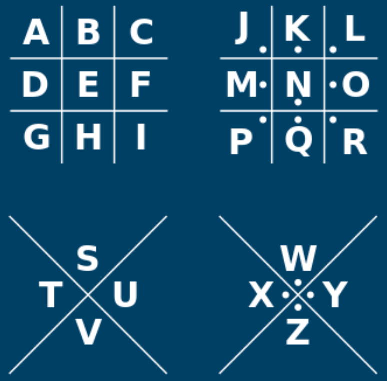
Broken.

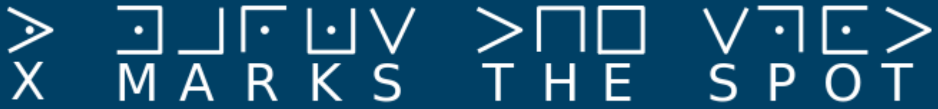
history

More Old Ciphers

700 BC	Scytale	Ancient Greece transposition cipher
600 BC	Atbash	Israel (Essenes, Jewish rebels) substitution cipher; maps each letter to its inverse.
1500s?	Pigpen	Knights Templars, Freemasons Substitution cipher, polyalphabetic







then came the World Wars...

history

Information Warfare

1917. UK declares war on Germany;
cuts undersea cables to/from Germany.

Germany instead uses international cables
& radio. encrypts.

German foreign secretary Zimmermann telegrams Mexico & Japan; asking them to pre-emptive strike USA.

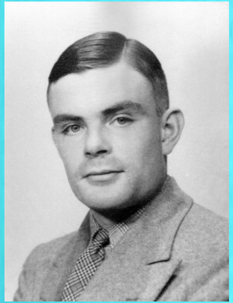
UK intercepts, breaks the cipher, informs USA.
USA enters WWI. Germany is defeated.

EASTERN TELEGRAPH Co's SYSTEM AND IT
ARCTIC SEA
PACIFIC OCEAN
NORTH AMERICA
SOUTH AMERICA
BAFFIN BAY
HUDSON BAY
Behring Sea
ANTARCTIC

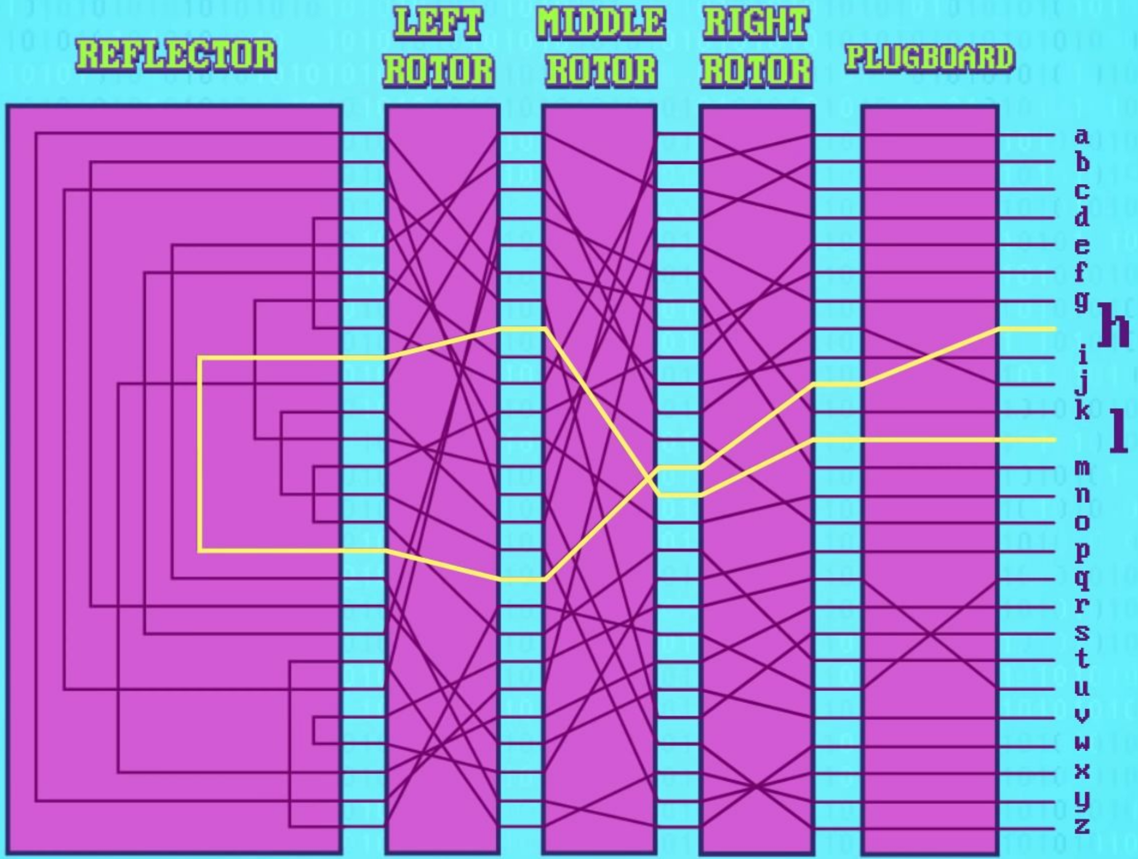
history

Enigma

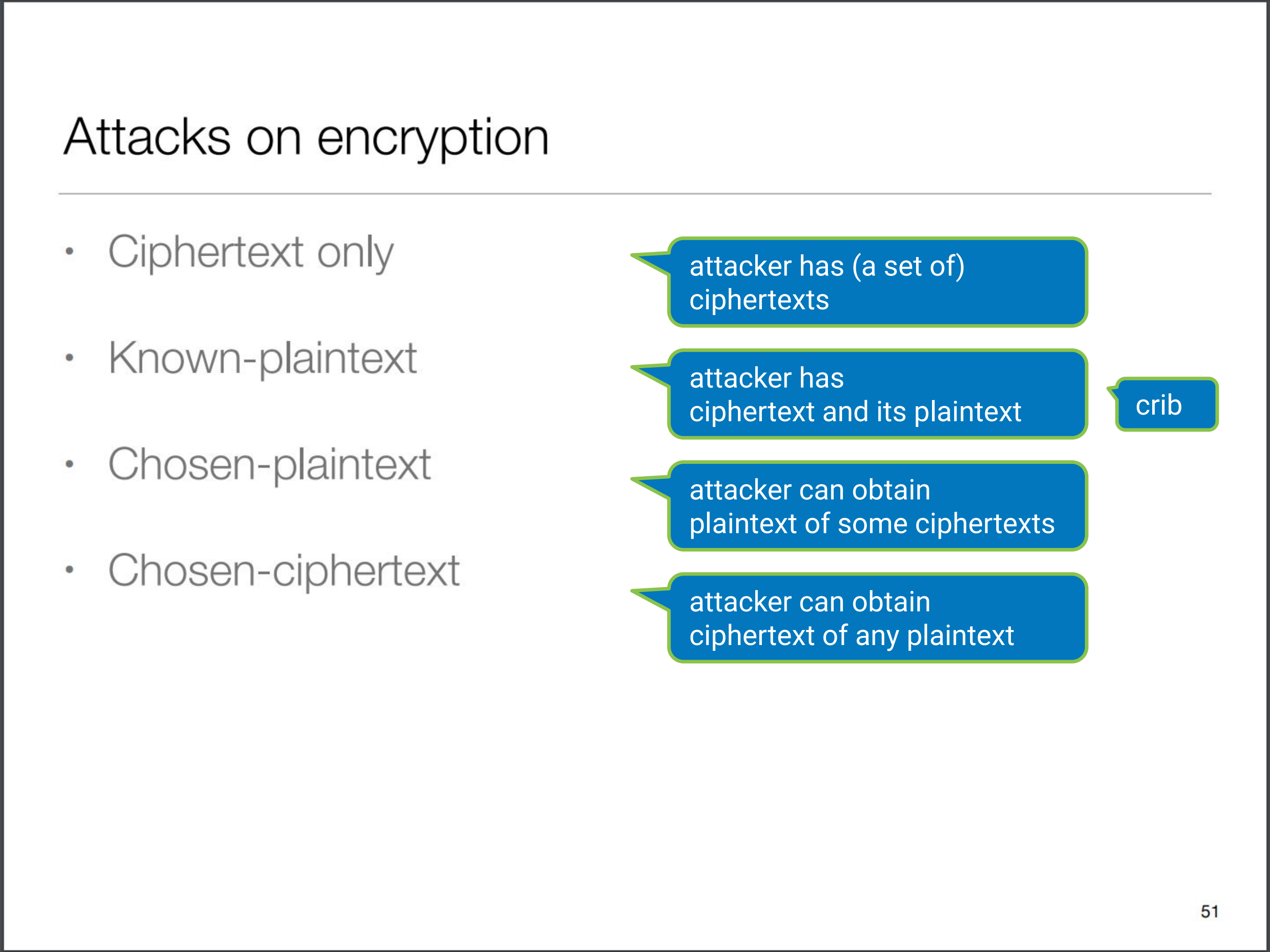
broken also; greatly shortened WWII



Alan Turing
("The Imitation Game")



Lost war due to broken cipher. Germany invests in stronger cipher machines.



Attacks on encryption

- Ciphertext only
- Known-plaintext
- Chosen-plaintext
- Chosen-ciphertext

attacker has (a set of) ciphertexts

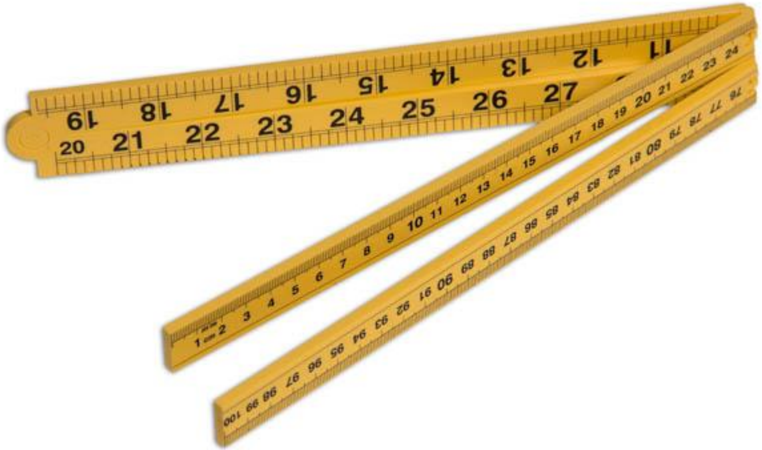
attacker has ciphertext and its plaintext

crib

attacker can obtain plaintext of some ciphertexts

attacker can obtain ciphertext of any plaintext

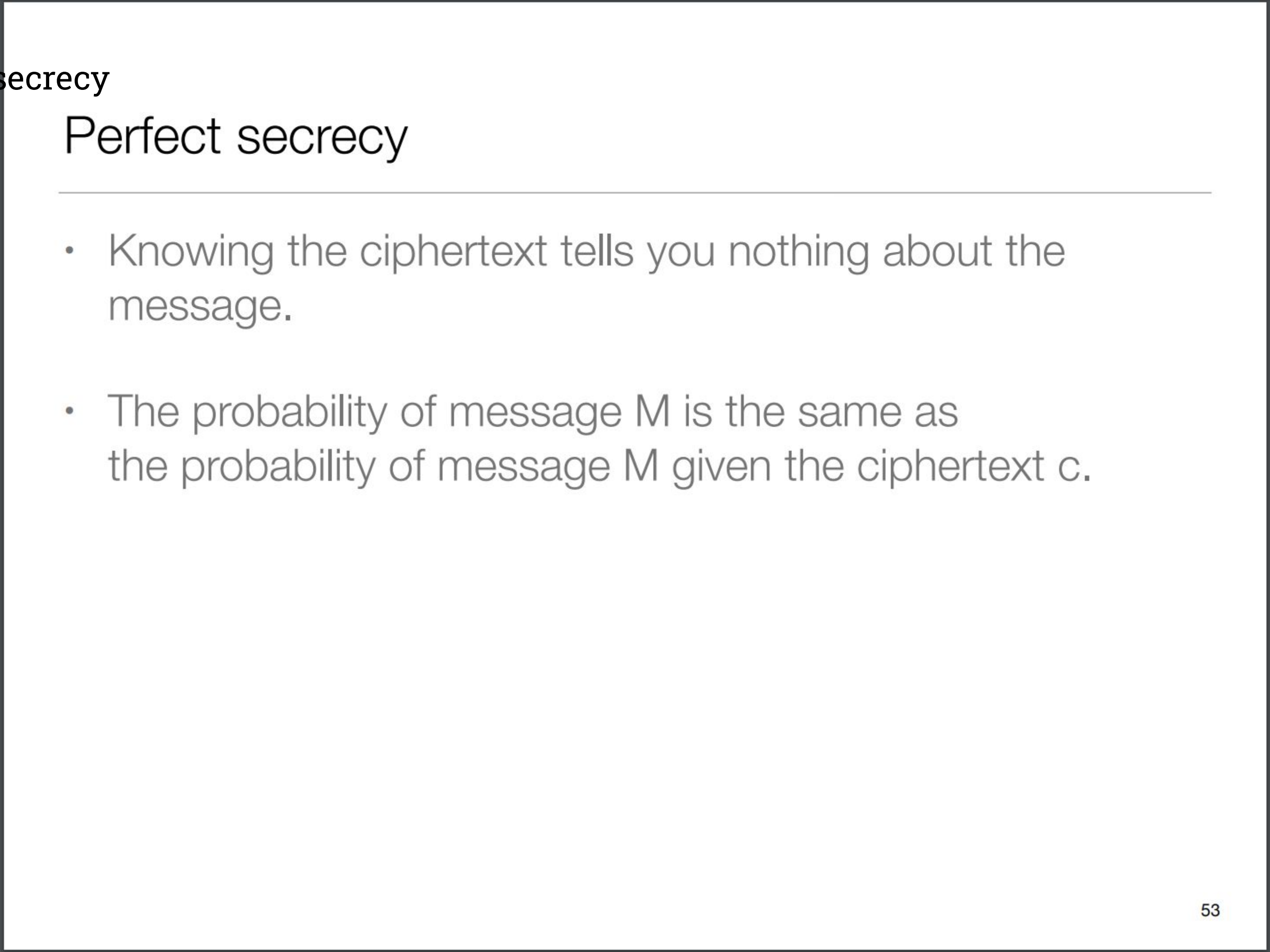
When is a cipher “secure”?



perfect secrecy

one-time pad
OTP

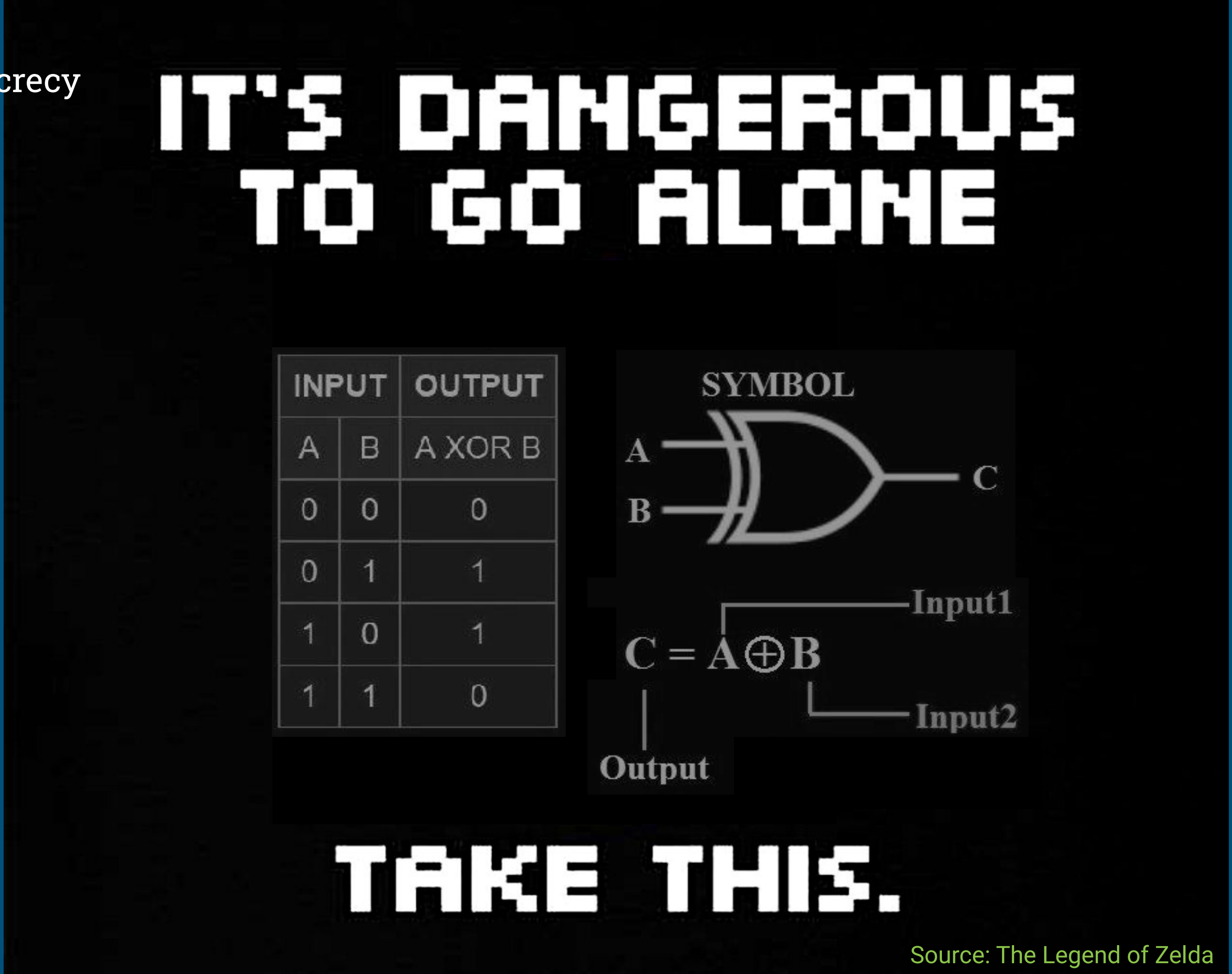
perfect secrecy



Perfect secrecy

- Knowing the ciphertext tells you nothing about the message.
- The probability of message M is the same as
the probability of message M given the ciphertext c.

perfect secrecy



INPUT		OUTPUT
A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

$C = A \oplus B$

Source: The Legend of Zelda

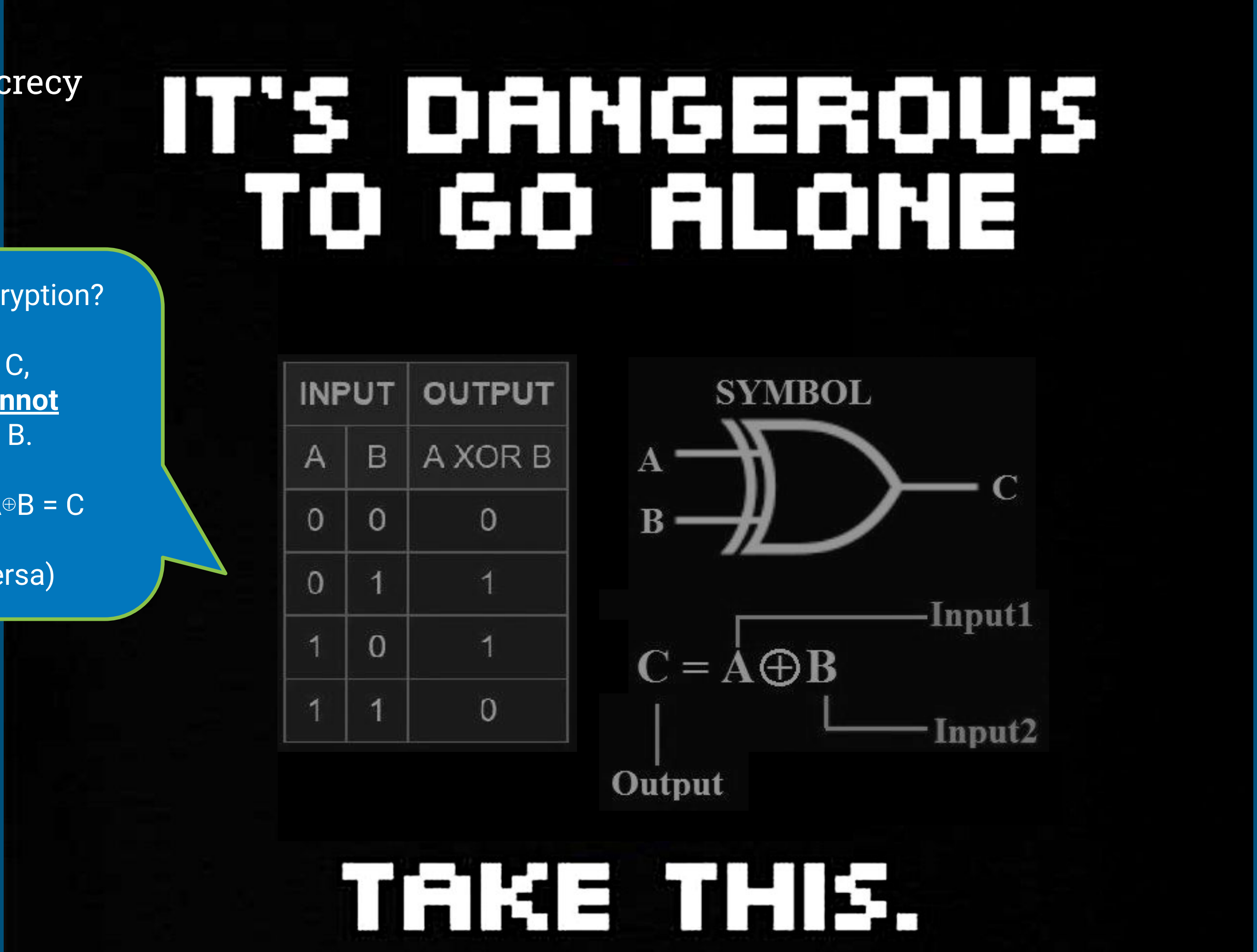
perfect secrecy

why is ⊕encryption?

if you know C,
then you **cannot**
predict A or B.

∀A . ∃B . A⊕B = C

(and vice versa)



INPUT		OUTPUT
A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

Source: The Legend of Zelda

perfect secrecy

Encrypt

Plaintext: **Hi!** 1001000 1101001 0100001

Key: **0l;** XOR 0110000 1101100 0111011

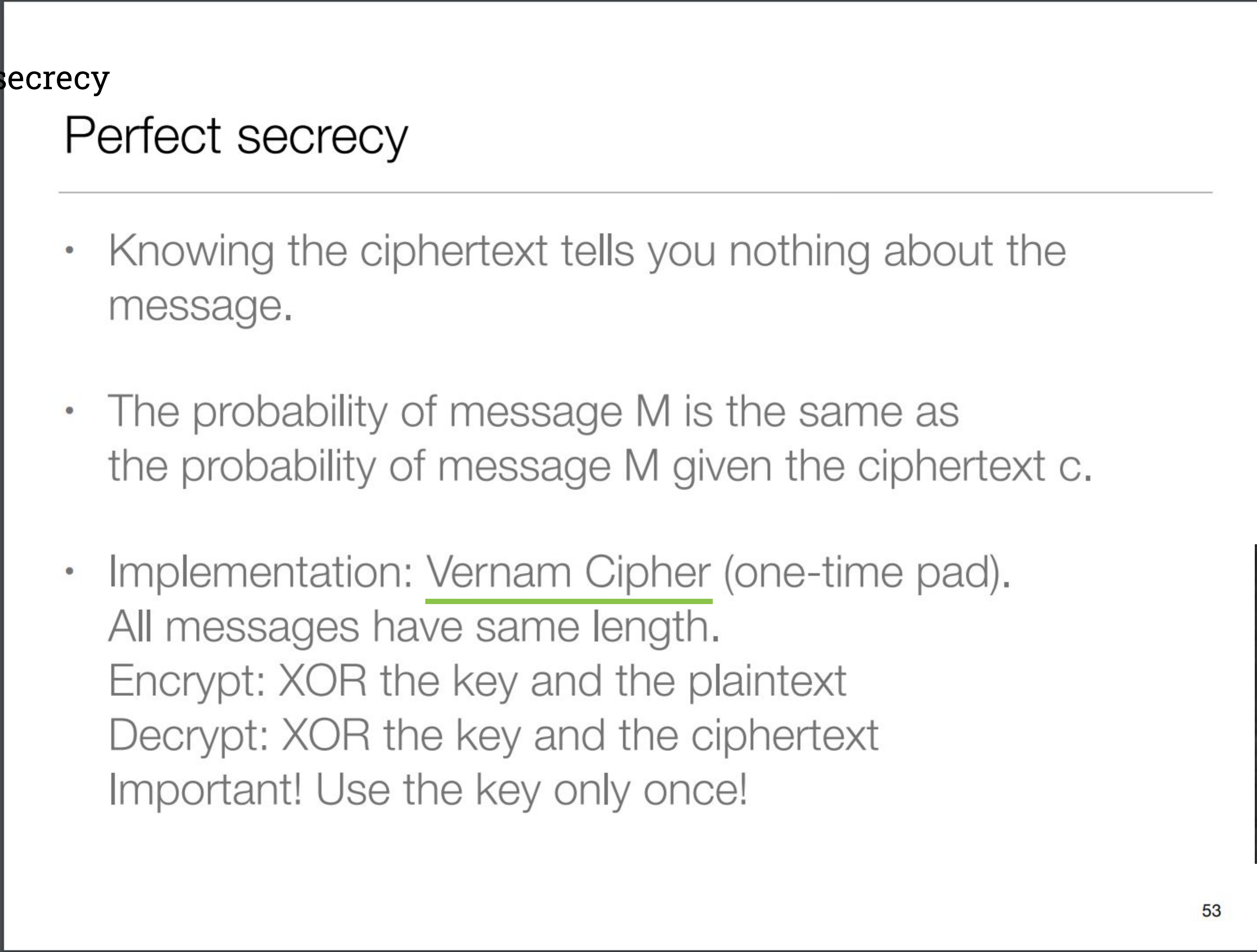
Ciphertext: **x□□** 1111000 0000101 0011010

ENQ SUB

- Ciphertext doesn't *need* to be converted to characters as they won't always make sense – it'll just be exchanged in binary

A	B	A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

perfect secrecy

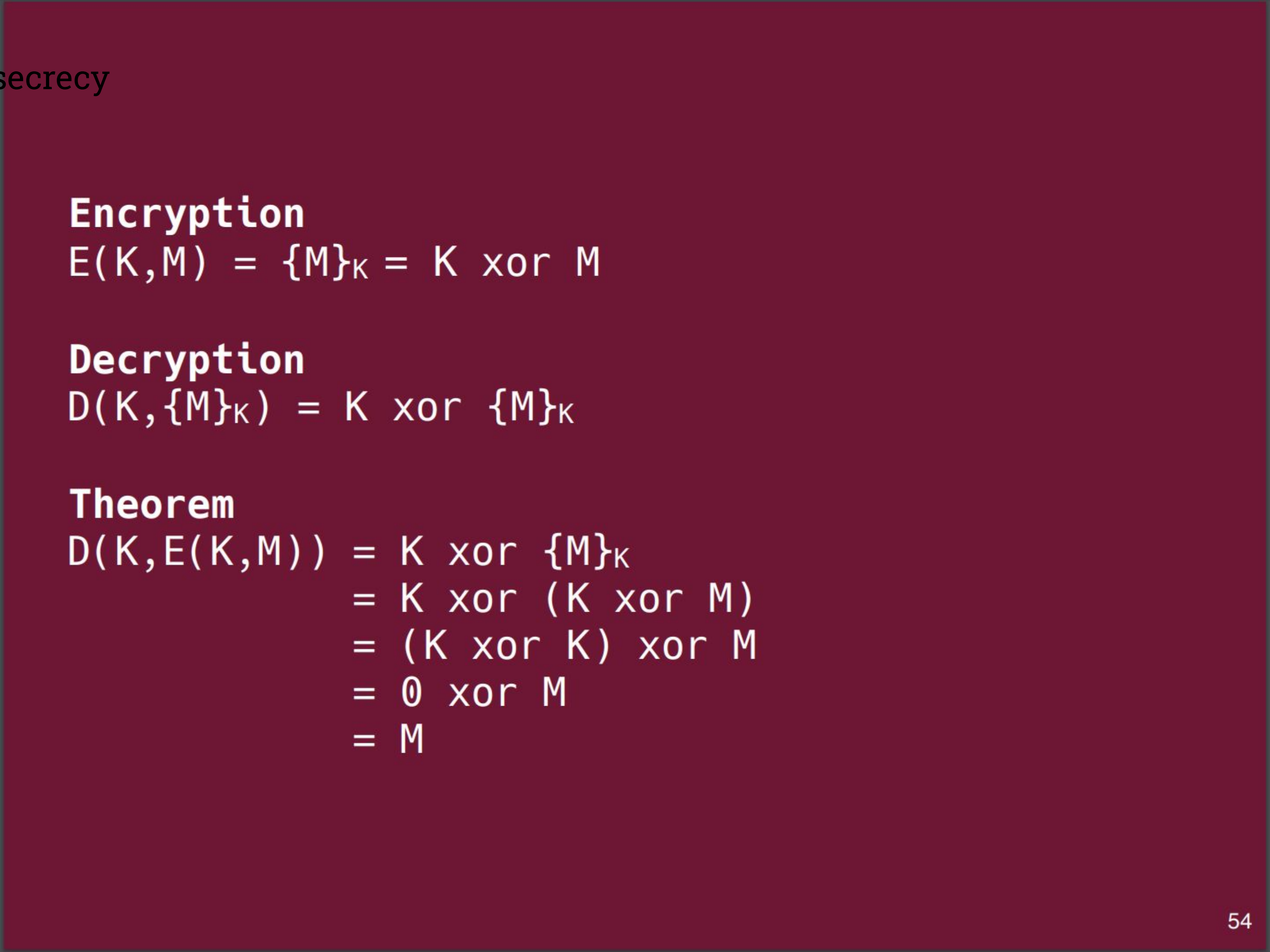


Perfect secrecy

- Knowing the ciphertext tells you nothing about the message.
- The probability of message M is the same as the probability of message M given the ciphertext c.
- Implementation: Vernam Cipher (one-time pad).
  All messages have same length.
  Encrypt: XOR the key and the plaintext
  Decrypt: XOR the key and the ciphertext
  Important! Use the key only once!

1917,
1919

perfect secrecy



Encryption

$E(K,M) = \{M\}_K = K \text{ xor } M$

Decryption

$D(K,\{M\}_K) = K \text{ xor } \{M\}_K$

Theorem

$$\begin{aligned} D(K,E(K,M)) &= K \text{ xor } \{M\}_K \\ &= K \text{ xor } (K \text{ xor } M) \\ &= (K \text{ xor } K) \text{ xor } M \\ &= 0 \text{ xor } M \\ &= M \end{aligned}$$

perfect secrecy

Perfect secrecy

- **Important! Use the key only once!**
- Vernon cipher not practical:
  Need as many bits of pre-agreed key as bits of plaintext.
- Think about how much mail you get.
- Need: fixed-size key for arbitrary amount of messages.
- Theorem (Shannon): Vernon cipher is optimal.
  Perfect secrecy requires as one bit key for each one bit of plaintext.

1949

Key Re-use ⇒ Crib-Dragging

$C_A = A \oplus K$
$C_B = B \oplus K$

$$\begin{aligned} C_A \oplus C_B &= (A \oplus K) \oplus (B \oplus K) \\ &= (A \oplus K) \oplus (K \oplus B) \\ &= A \oplus (K \oplus K) \oplus B \\ &= A \oplus 0 \oplus B \\ &= A \oplus B \end{aligned}$$

Is it bad to know $A \oplus B$, and not A, B?

perfect secrecy

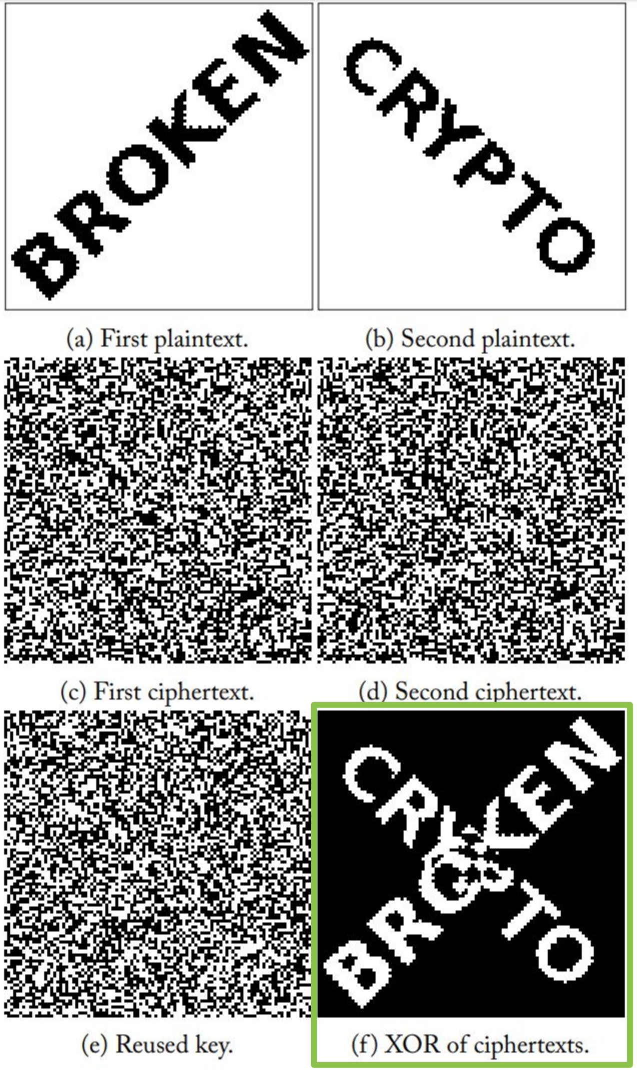
Key Re-use ⇒ Crib-Dragging

$C_A = A \oplus K$
$C_B = B \oplus K$

$$
\begin{aligned}
C_A \oplus C_B &= (A \oplus K) \oplus (B \oplus K) \\
&= (A \oplus K) \oplus (K \oplus B) \\
&= A \oplus (K \oplus K) \oplus B \\
&= A \oplus 0 \oplus B \\
&= A \oplus B
\end{aligned}
$$

Is it bad to know $A \oplus B$, and not A, B?

Can also be done on text.



(a) First plaintext.

(b) Second plaintext.

(c) First ciphertext.

(d) Second ciphertext.

(e) Reused key.

(f) XOR of ciphertexts.

victory!

use OTP for everything!

… but how do I share a key stream?
where do I get a key stream?

key generation

random number generator
RNG

key generation

Randomness on a Computer

recall OTP:

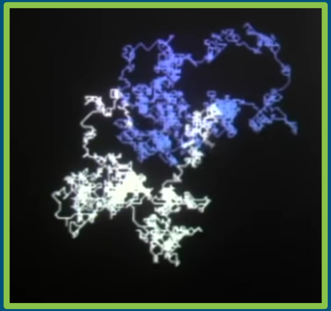
- key must be random,
- key must never be re-used.

how do we get infinite randomness (nondeterministic), on a finite machine (deterministic)?

- true RNG (HRNG)
- pseudo-RNG (PRNG)
- cryptographically-secure pseudo-RNG (CSPRNG)

True RNG

HRNG



sample an unpredictable physical process.

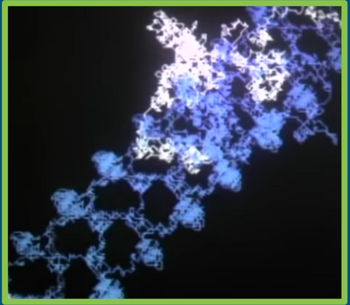
- quantum process
  radioactive decay, shot noise (e.g. photons)
- thermal process
  Nyquist (electrons through resistant medium)
- oscillator drift
  ring oscillator frequency drift
- timing events
  keyboard/network I/O

too slow (run out of entropy), too unreliable.

key generation

Pseudo-RNG

PRNG



take seed, use it to generate numbers.

John Von Neumann: $k_{n+1} = k_n^2$ w/ first and last digit removed.

ex: $k_n = 121$, $k_{n+1} = 1|464|1 = 464$

state

All eventually hit a period.

predictable.

"Any one who considers arithmetical methods of producing random digits is, of course, in a state of sin."

- John von Neumann

Cryptographically Secure PRNG

CSPRNG

unpredictable PRNG.
do not leak info on its state.

if you must pick yourself: **always** pick CSPRNG provided by your **OS**

- `/dev/urandom` (*NIX)
- `CryptGenRandom` (Windows)

big seed ⇒ big period.

PL interface to these. Python:
`os.urandom`, `random.SystemRandom`

Kerberos V4 used PRNG. broken.

victory!

CSPRNG to get our OTP key stream!

… but is that really secure?

key generation

A Practical Stream Cipher?

recall OTP:

- key must be random,
- key must never be re-used.

<u>idea:</u> Vernom stream cipher, w/ CSPRNG key stream.
finite HR ⇒ infinite PR. perfect secrecy?

key generation

A Practical Stream Cipher?

recall OTP:

- key must be random,
- key must never be re-used.

idea: Vernam stream cipher, w/ CSPRNG key stream.
finite HR ⇒ infinite PR. perfect secrecy?
no: $\mathbf{K} \le \mathbf{M}$ (because **K** is the *seed*)
security rests on unpredictability of the CSPRNG. **good/bad?**

instead: encryption & key-expansion together. (AES+CBC)
intuition: the more you encrypt w/ a **K**, the more breakable.

not enough randomness?
(used in **synchronous stream ciphers**)
(RC4 & Salsa20 are fancy versions)

capitalize on randomness
that may be present in the data

encrypt / decrypt

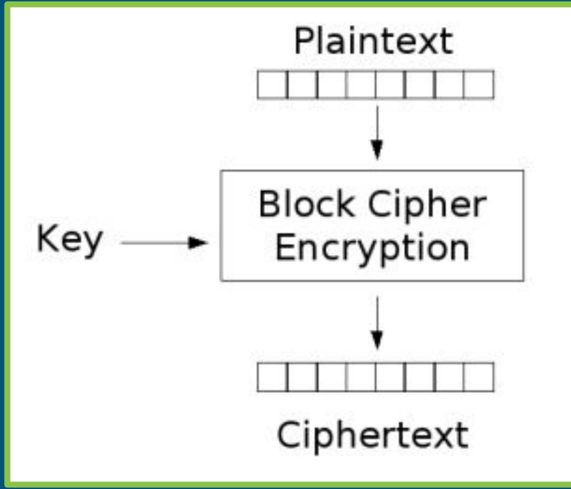
random number generator
RNG

encrypt / decrypt

a block

advanced encryption standard

AES



encrypt / decrypt - a block

What is a Block Cipher?

block = fixed-size sequence of bits

it’s just a giant lookup table. →

keyed permutation

- **D**(**K**, **E**(**K**, **M**)) = **M**
- given a **K**, **E** is a *permutation*.
- changing **K** should not make predictable which **E** emerges (random permutation).

example: Caesar not a block cipher.

Key	**Plaintext**	**Ciphertext**
K	M_1	C_1
K	M_2	C_2
K	M_3	C_3
K	M_4	C_4
…	…	…

M, **C** drawn from same set

encrypt / decrypt - a block

What is a Block Cipher?

block = fixed-size sequence of bits

it's just a giant lookup table.

- **D**(**K**, **E**(**K**, **M**)) = **M**
- given a **K**, **E** is a *permutation*.
- changing **K** should not make predictable which **E** emerges (random permutation).

keyed permutation

Key	**Plaintext**	**Ciphertext**
K	M_1	C_1
K	M_2	C_2
K	M_3	C_3
K	M_4	C_4
…	…	…

M, **C** drawn from same set

example: Caesar not a block cipher.

- changing **K**, you can predict which **E** emerges.

encrypt / decrypt - a block

AES

advanced encryption standard

3DES is still used, in ancient (financial) applications, w/ HW support. Slow, but secure?

NSA’s DES fails; NIST starts an open process for proposal AES.

by: Vincent Rijmen & Joan Daemen

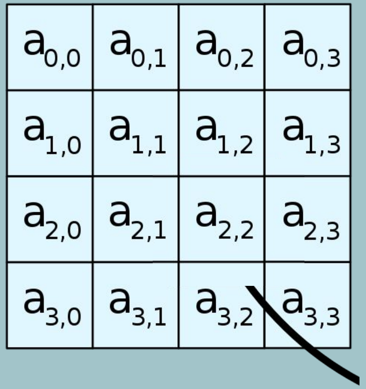
● confusion	substitute
● diffusion	permute
● key	the only secret

no known practical attacks.
parallelizable!

AES, in Pictures

prep: derive 10* separate 128^-bit keys from master key.

key expansion



$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$

load data into state matrix

*: or 12 rounds, or 14 rounds

^: or 192-bit, or 256-bit

encrypt / decrypt - a block - AES

AES, in Pictures

prep: derive 10* separate 128^-bit keys from master key.

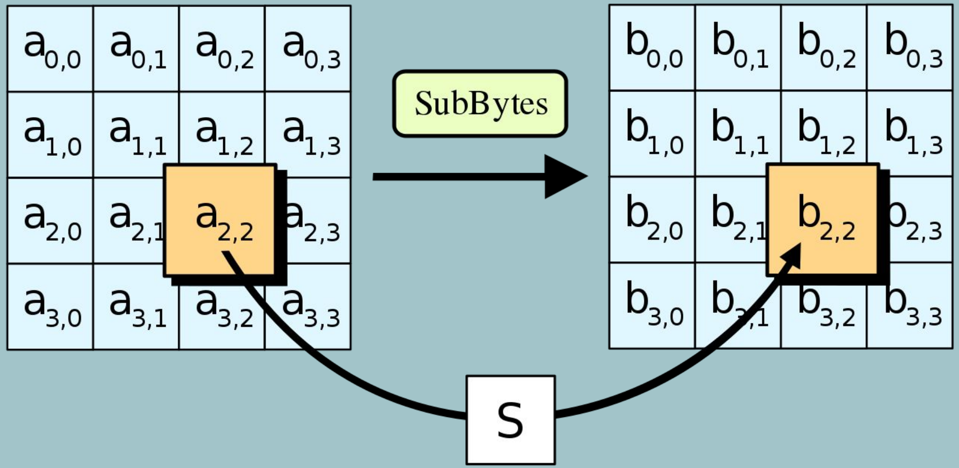
each round:

1. apply 8-bit S-box on each cell.

*: or 12 rounds, or 14 rounds

^: or 192-bit, or 256-bit

substitute



$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
SubBytes
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
S

encrypt / decrypt - a block - AES

AES, in Pictures

prep: derive 10* separate 128^-bit keys from master key.

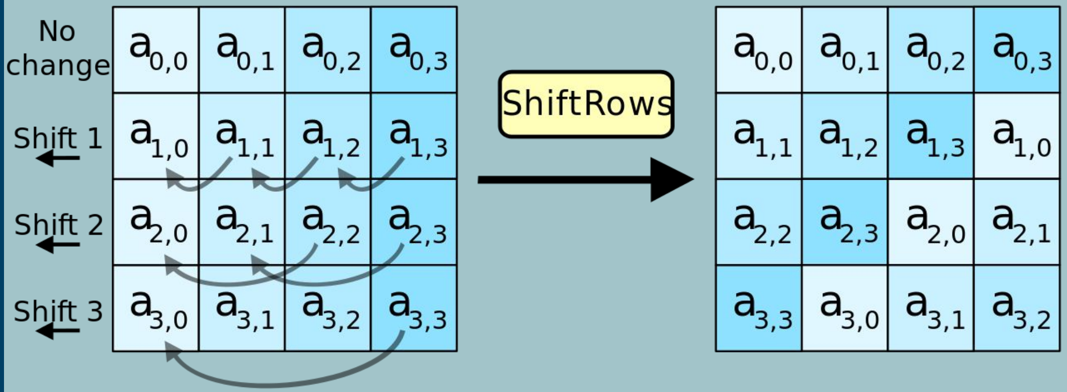
each round:

1. apply 8-bit S-box on each cell.
2. shift rows as depicted.

*: or 12 rounds, or 14 rounds

^: or 192-bit, or 256-bit

permute



No change	$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
Shift 1	$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
Shift 2	$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
Shift 3	$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,0}$
$a_{2,2}$	$a_{2,3}$	$a_{2,0}$	$a_{2,1}$
$a_{3,3}$	$a_{3,0}$	$a_{3,1}$	$a_{3,2}$

“rotate” row k steps

encrypt / decrypt - a block - AES

AES, in Pictures

prep: derive 10* separate 128^-bit keys from master key.

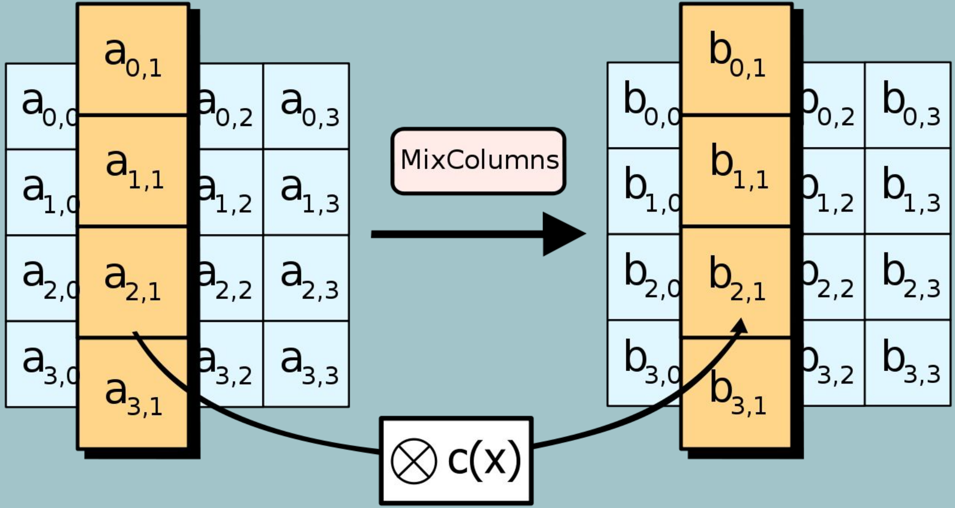
each round:

1. apply 8-bit S-box on each cell.
2. shift rows as depicted.
3. multiply each column w/ a constant (matrix)

*: or 12 rounds, or 14 rounds

^: or 192-bit, or 256-bit

substitute


$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
MixColumns
$\otimes c(x)$
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$

encrypt / decrypt - a block - AES

AES, in Pictures

prep: derive 10* separate 128^-bit keys from master key.

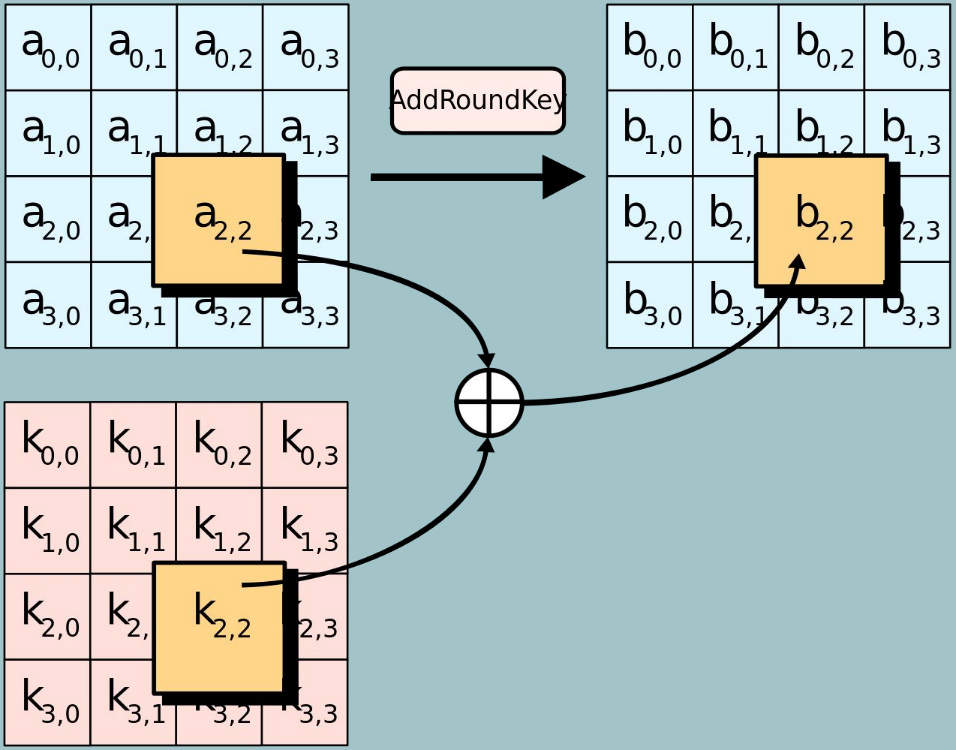
each round:

1. apply 8-bit S-box on each cell.
2. shift rows as depicted.
3. multiply each column w/ a constant (matrix)
4. XOR in the round-key.

*: or 12 rounds, or 14 rounds

^: or 192-bit, or 256-bit

key


$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
AddRoundKey
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
$k_{0,0}$ $k_{0,1}$ $k_{0,2}$ $k_{0,3}$
$k_{1,0}$ $k_{1,1}$ $k_{1,2}$ $k_{1,3}$
$k_{2,0}$ $k_{2,1}$ $k_{2,2}$ $k_{2,3}$
$k_{3,0}$ $k_{3,1}$ $k_{3,2}$ $k_{3,3}$

encrypt / decrypt - a block

victory!

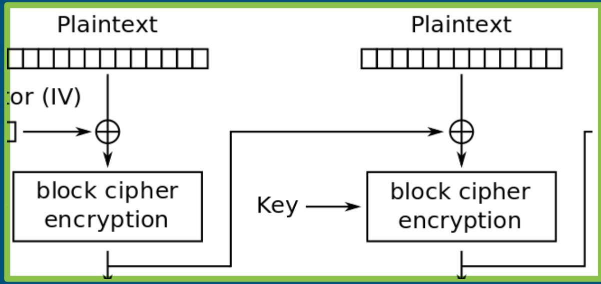
I can encrypt a block w/ a small key

… but my data is much larger than a block...

a stream of blocks

cipher block chaining mode

CBC



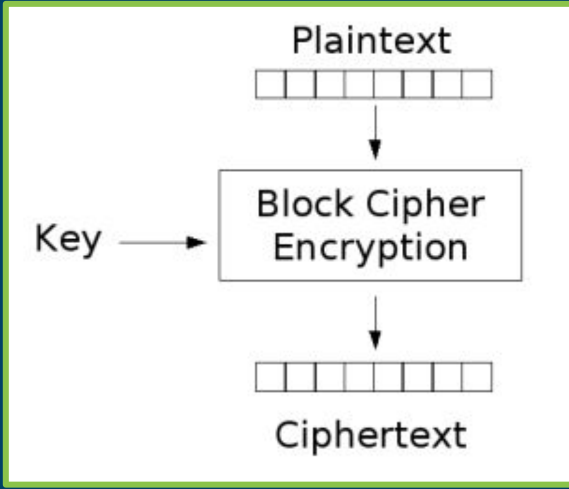
encrypt / decrypt - a stream of blocks

From Block Cipher to Stream Cipher

we have a block cipher.

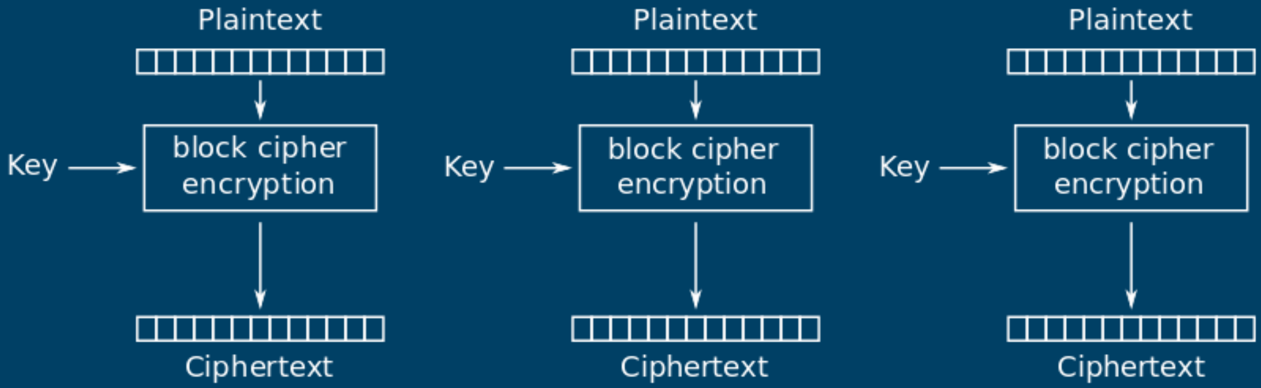
our data is larger than a block (pad to fit)

we can use our block cipher to encrypt our stream,
by cutting our stream into blocks,
and encrypting the blocks.
sounds easy...

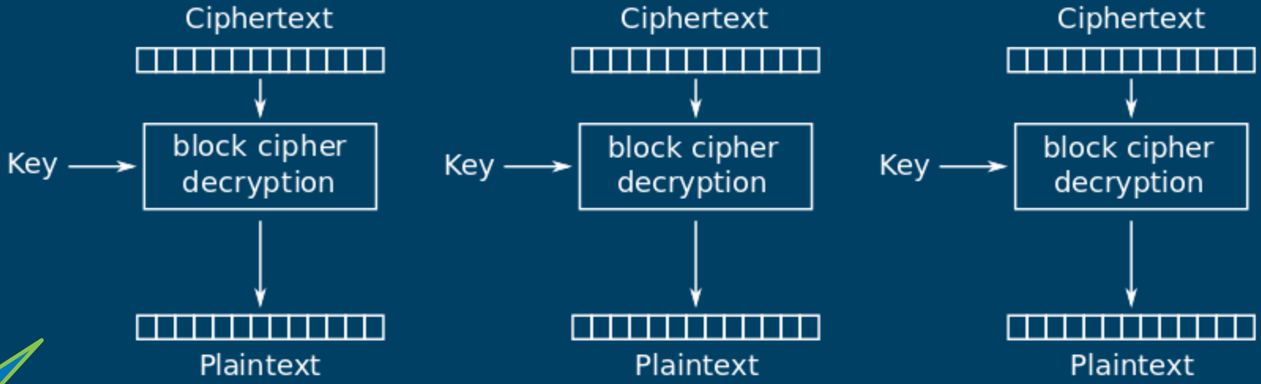


Electronic Codebook

encrypt / decrypt - a stream of blocks



Electronic Codebook (ECB) mode encryption

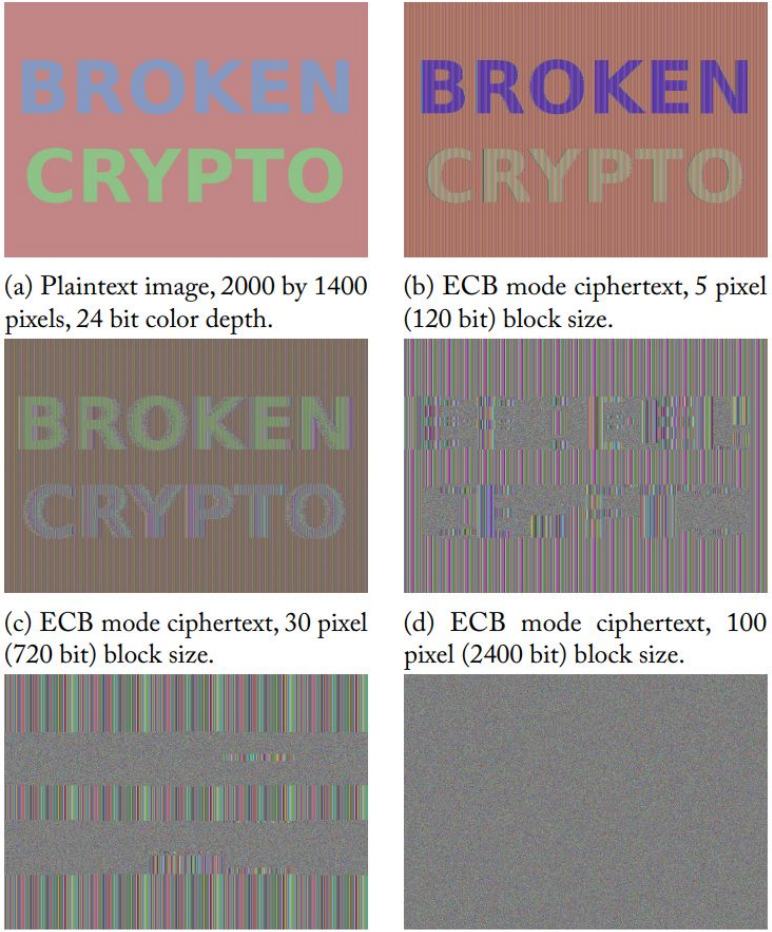


problem?

Electronic Codebook (ECB) mode decryption encrypt / decrypt - a stream of blocks

ECB Attack

Each block M in the stream always encrypts to the same ciphertext block C.



(a) Plaintext image, 2000 by 1400 pixels, 24 bit color depth.

(b) ECB mode ciphertext, 5 pixel (120 bit) block size.

(c) ECB mode ciphertext, 30 pixel (720 bit) block size.

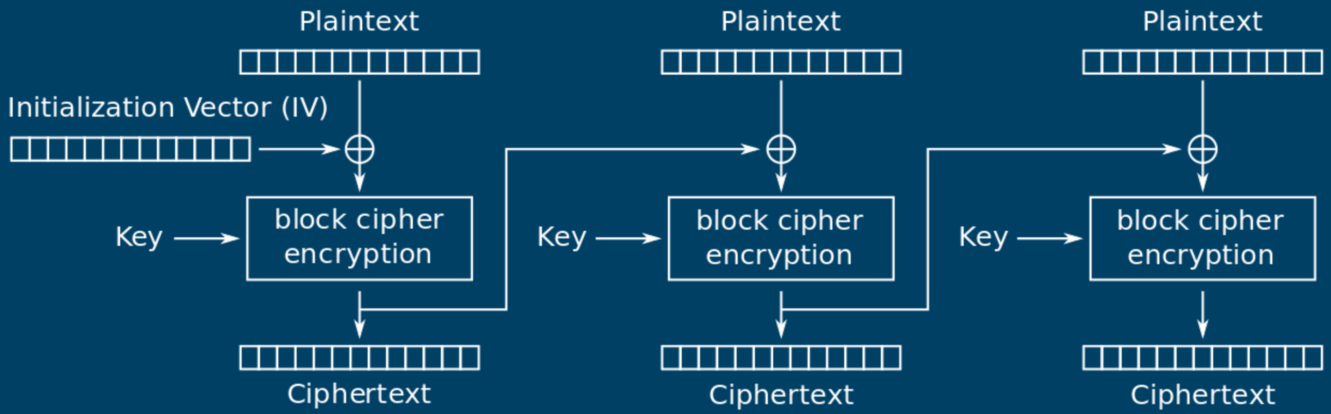
(d) ECB mode ciphertext, 100 pixel (2400 bit) block size.

(e) ECB mode ciphertext, 400 pixel (9600 bit) block size.

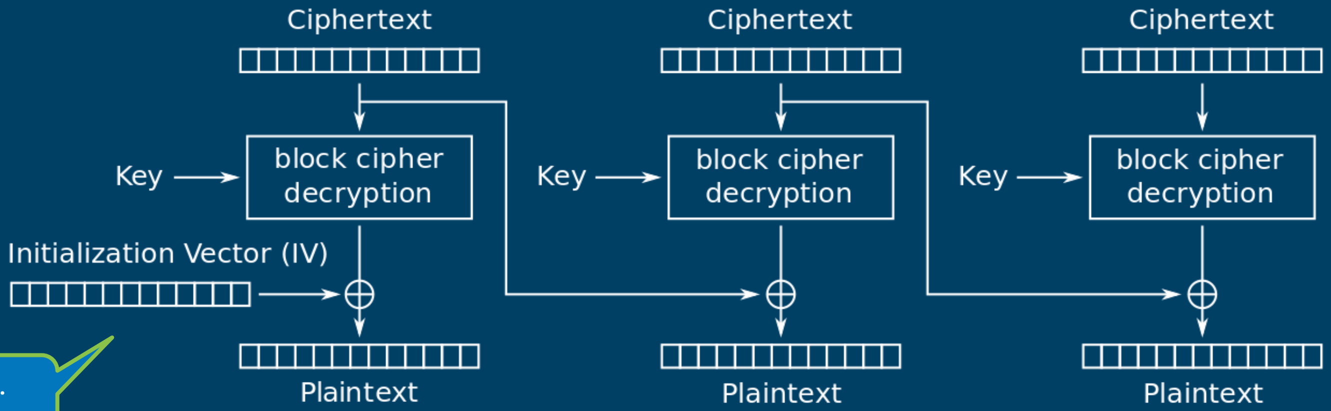
(f) Ciphertext under idealized encryption.

Cipher Block Chaining

encrypt / decrypt - a stream of blocks



Cipher Block Chaining (CBC) mode encryption

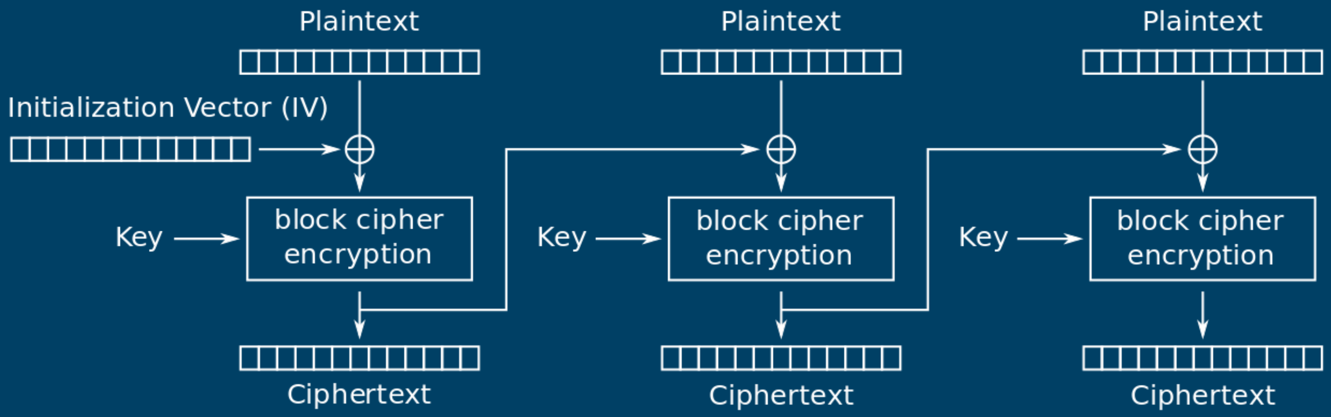


unpredictable.
problem?

Cipher Block Chaining (CBC) mode decryption

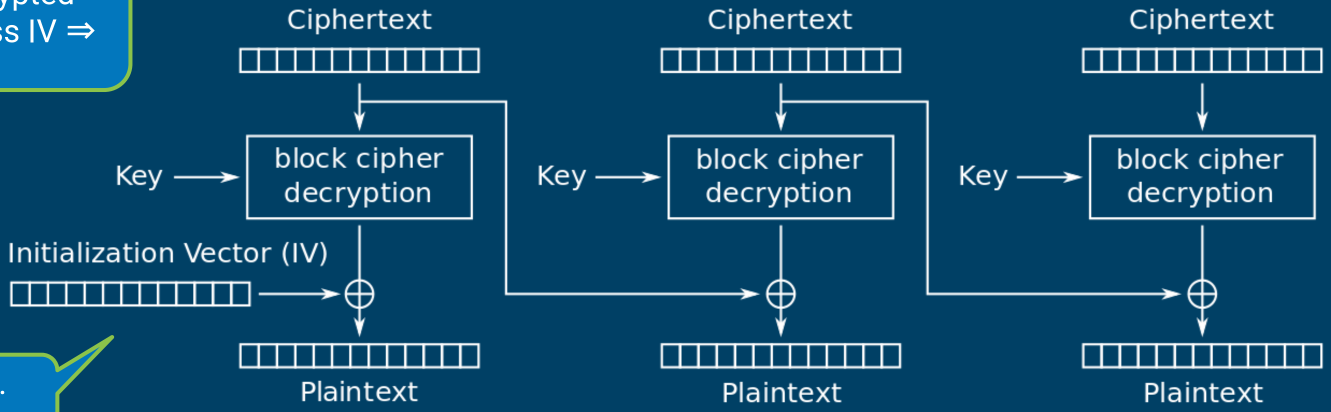
Cipher Block Chaining

encrypt / decrypt - a stream of blocks



Cipher Block Chaining (CBC) mode encryption

SQL injection to acquire encrypted blocks. Guess IV ⇒ decrypt.



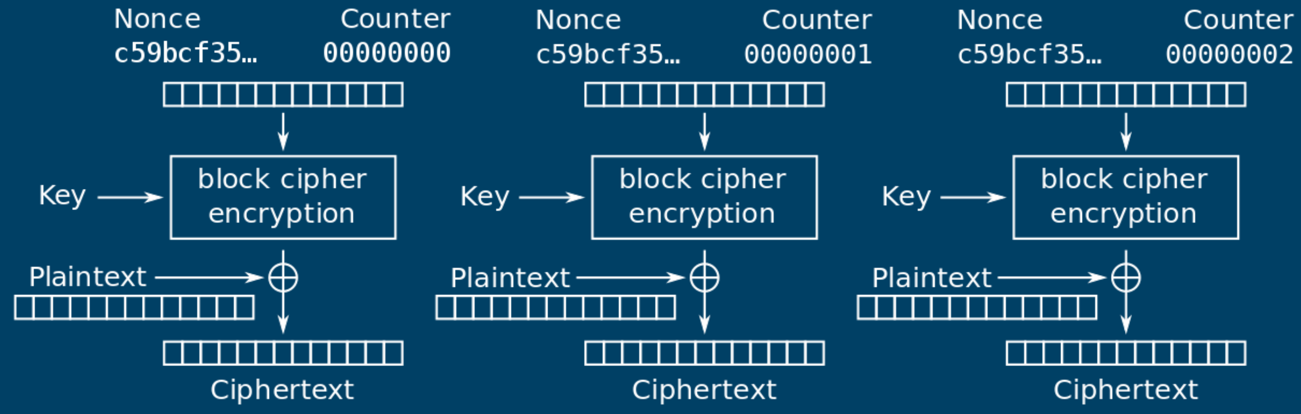
unpredictable. problem?

Cipher Block Chaining (CBC) mode decryption

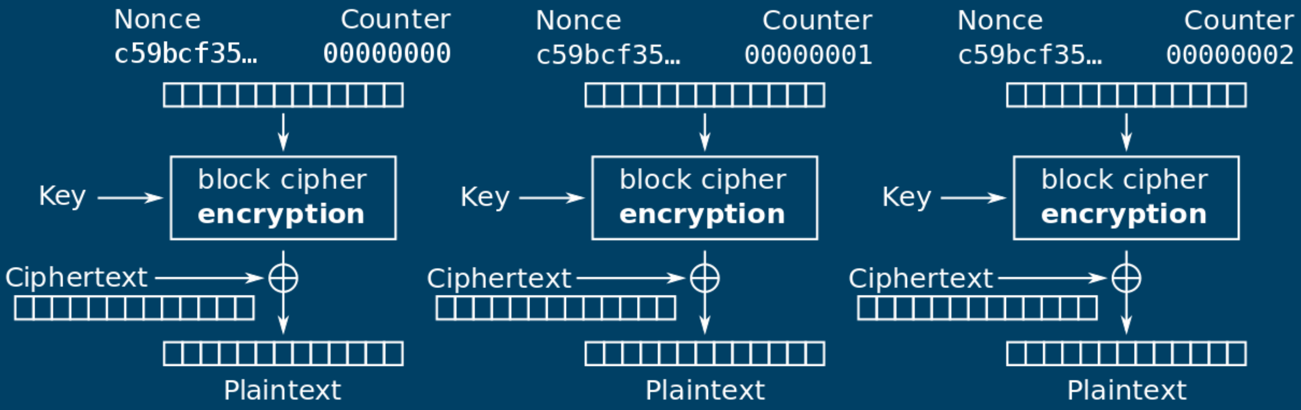
Counter

encrypt / decrypt - a stream of blocks

“jump”-able



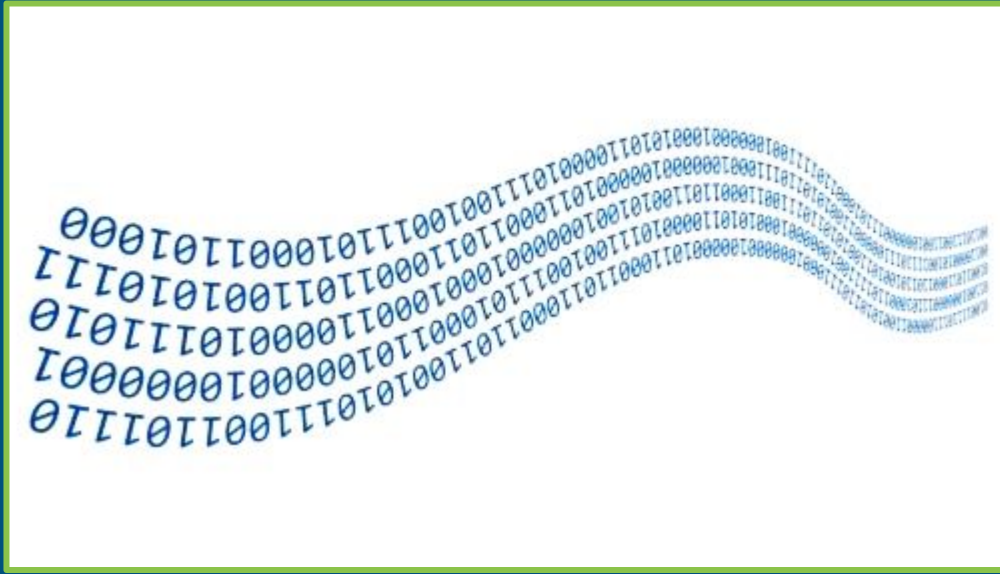
Counter (CTR) mode encryption



Counter (CTR) mode decryption

a stream

salsa20



encrypt / decrypt - a stream

RC4

Rivest cipher 4

By: Ron Rivest (RSA fame).

generates key stream.

used in WEP.

widely used on desktop and mobile!

fast!

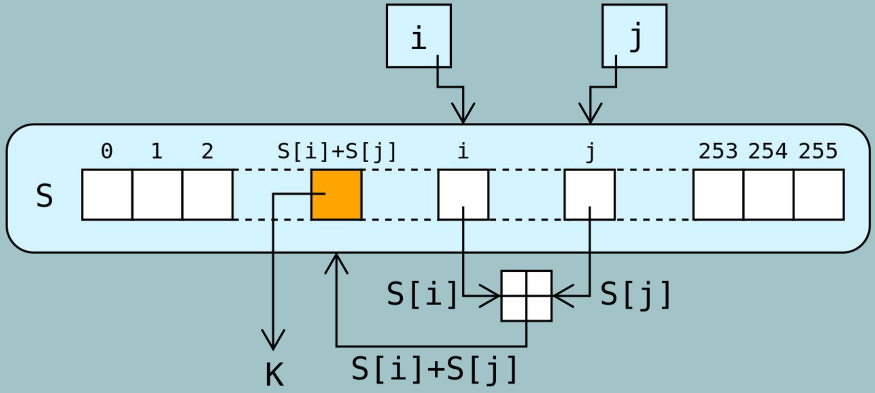
...broken :-/

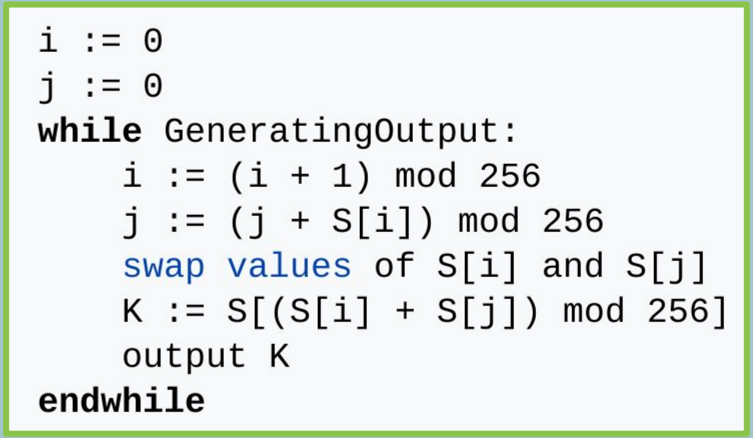
encrypt / decrypt - a stream

RC4

generates a keystream.

1. increments i
2. looks up the ith element of S, S[i], and adds that to j
3. exchanges the values of S[i] and S[j] then uses the sum S[i] + S[j] (modulo 256) as an index to fetch a third element of S (the keystream value K below)
4. then bitwise exclusive ORed (XORed) with the next byte of the message to produce the next byte of either ciphertext or plaintext.





```
i := 0
j := 0
while GeneratingOutput:
    i := (i + 1) mod 256
    j := (j + S[i]) mod 256
    swap values of S[i] and S[j]
    K := S[(S[i] + S[j]) mod 256]
    output K
endwhile
```

encrypt / decrypt - a stream - RC4

RC4 Attacks

bias in the output bytes.

- first three bytes of the key correlated with the first byte of the keystream.
- first few bytes of the state related to the key with a simple(linear) relation.

attacks only get better.

- second byte produced by cipher is twice as likely to be zero as it should be.

etc. etc. , eventually WEP broken!

encrypt / decrypt - a stream

Salsa20

By: Daniel J. Bernstein

generates key stream.

jumpable!

pretty fast

secure (so far); attacks break up to 8 out of 13 rounds

several rounds of **ARX**:
(A) modular addition +
(R) rotation with fixed rotation amounts <<<
(X) XOR ⊕

encrypt / decrypt

victory!

I can encrypt any amount of data!

… and send it over an untrusted medium?
how do we agree on a key?

key exchange

Diffie Hellman

key exchange

DH

Diffie-Hellman

by: Whitfield Diffie & Martin Hellman

allows two people to create a shared key, without ever having met.

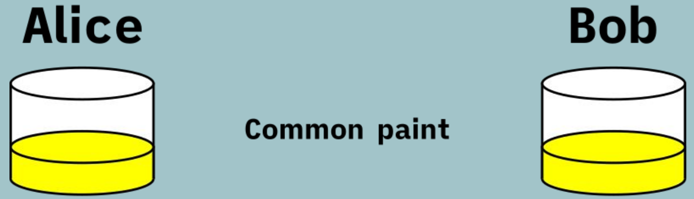
relies on the hardness of the discrete logarithm problem.

fundamental to security today.

key exchange - DH

Diffie Hellman w/ Colors

prep: common paint (public knowledge)

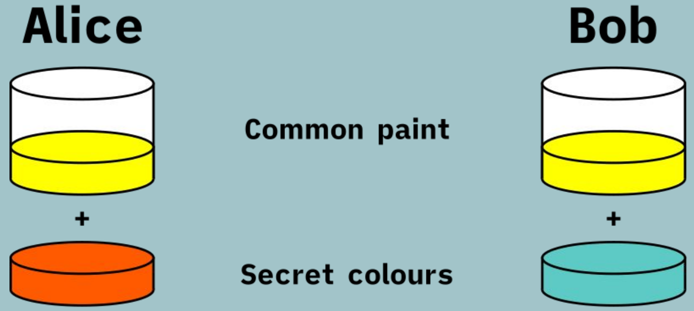


key exchange - DH

Diffie Hellman w/ Colors

<u>prep:</u> common paint (public knowledge)

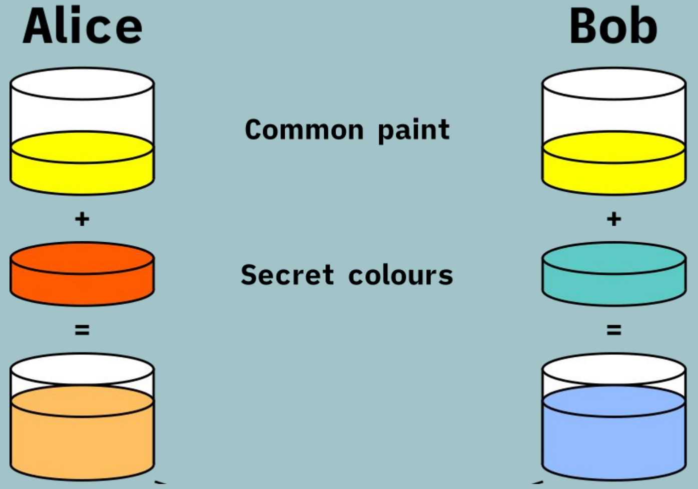
1. Alice & Bob each pick a secret.



Diffie Hellman w/ Colors

prep: common paint (public knowledge)

1. Alice & Bob each pick a secret.
2. each mixes secret w/ common,

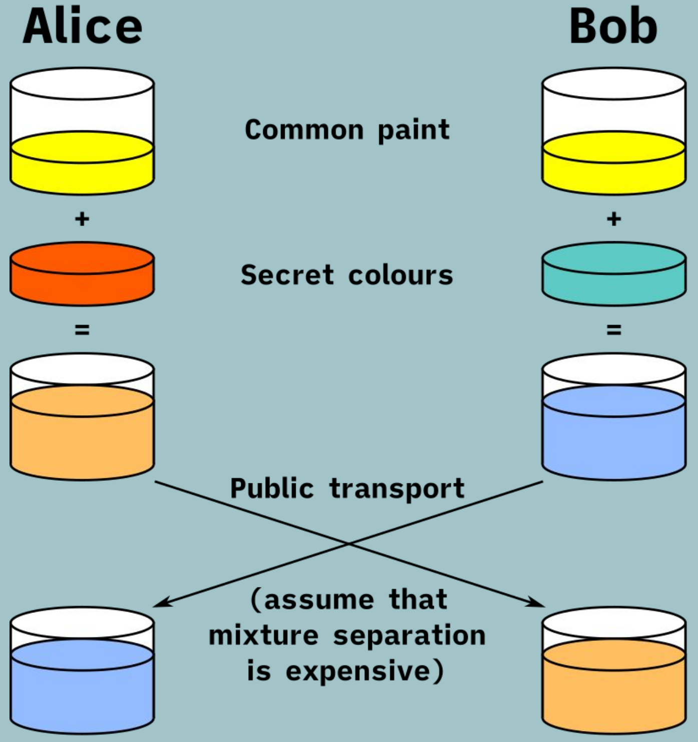


key exchange - DH

Diffie Hellman w/ Colors

prep: common paint (public knowledge)

1. Alice & Bob each pick a secret.
2. each mixes secret w/ common, and sends to the other.

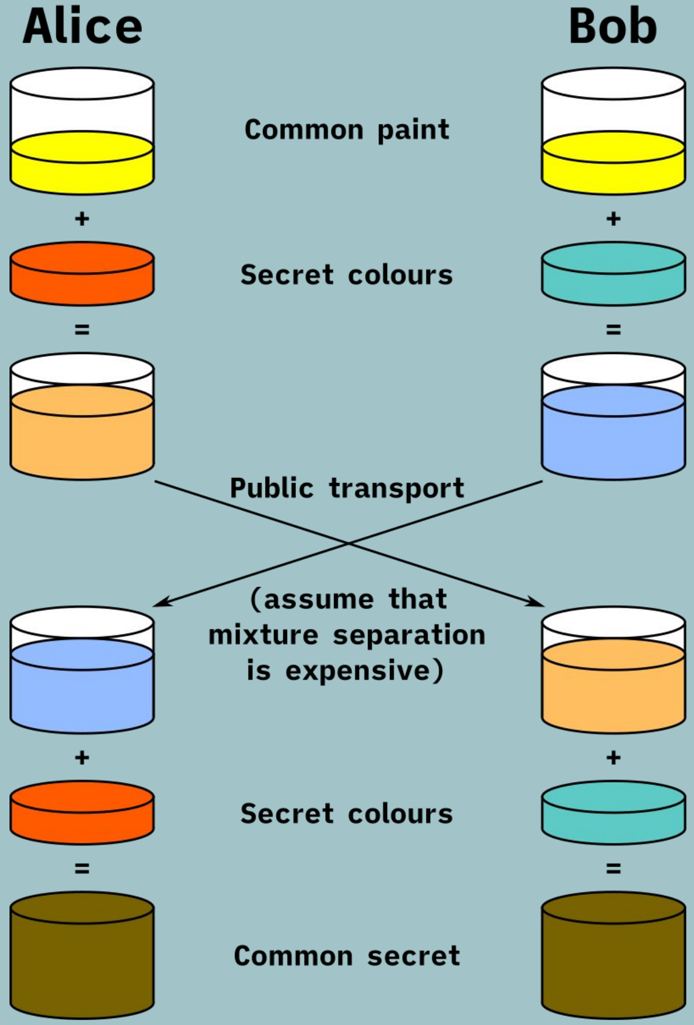


key exchange - DH

Diffie Hellman w/ Colors

prep: common paint (public knowledge)

1. Alice & Bob each pick a secret.
2. each mixes secret w/ common, and sends to the other.
3. each mixes secret w/ received.



Summary

summary

victory!

I can get a shared key with you!

… surely, we can talk securely now?

summary

we have only considered passive, eavesdropping adversaries…

summary

An attacker can do so much more.

summary

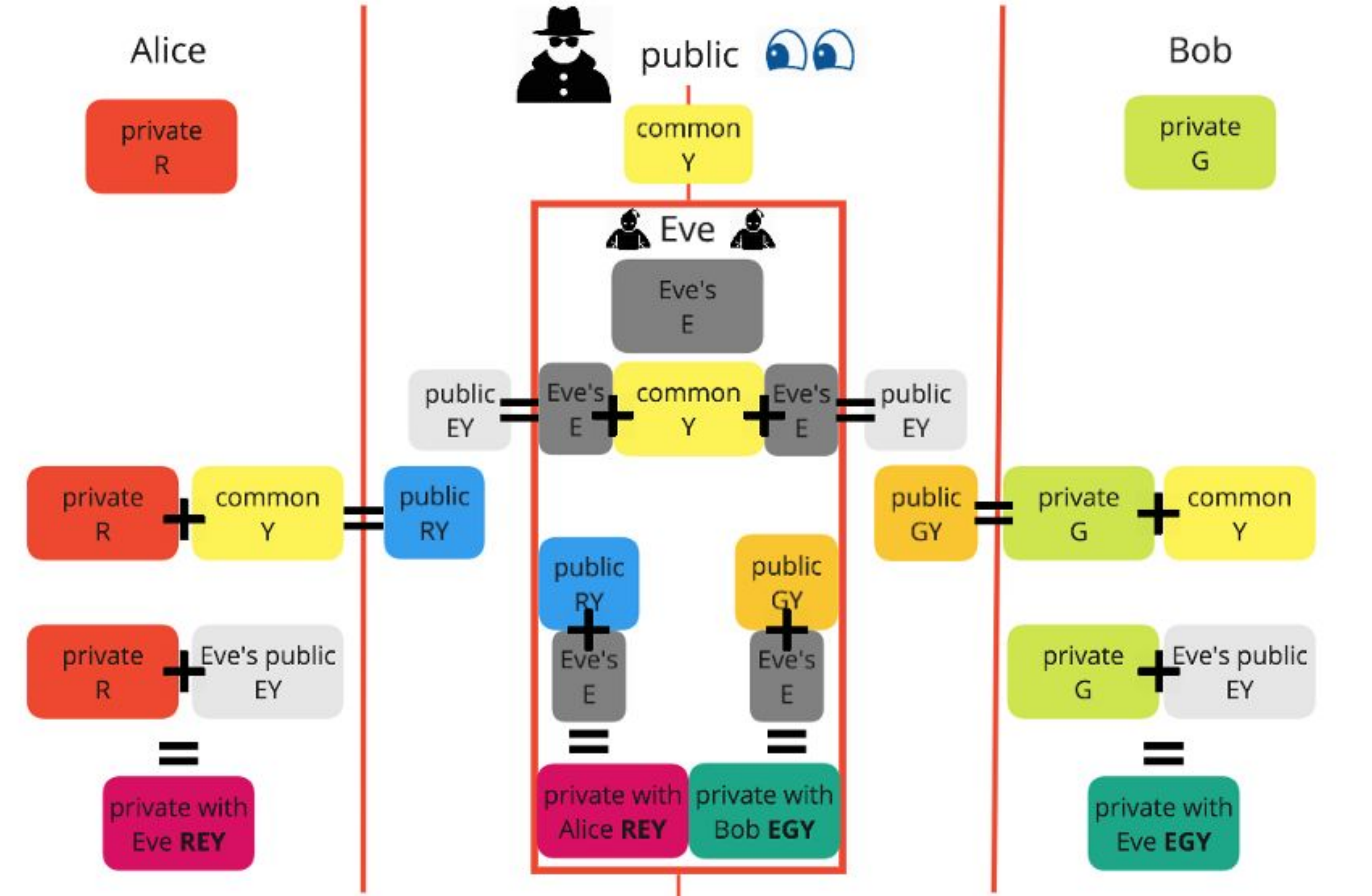
Replay Attack

just repeat encrypted traffic.

man in the middle

Diffie-Hellman MitM

summary



Alice

public

Bob

private R

common Y

private G

Eve

Eve's E

public EY = Eve's E + common Y + Eve's E = public EY

private R + common Y = public RY

public GY = private G + common Y

public RY + Eve's E = private with Alice **REY**

public GY + Eve's E = private with Bob **EGY**

private R + Eve's public EY = private with Eve **REY**

private G + Eve's public EY = private with Eve **EGY**

summary

IV Attack

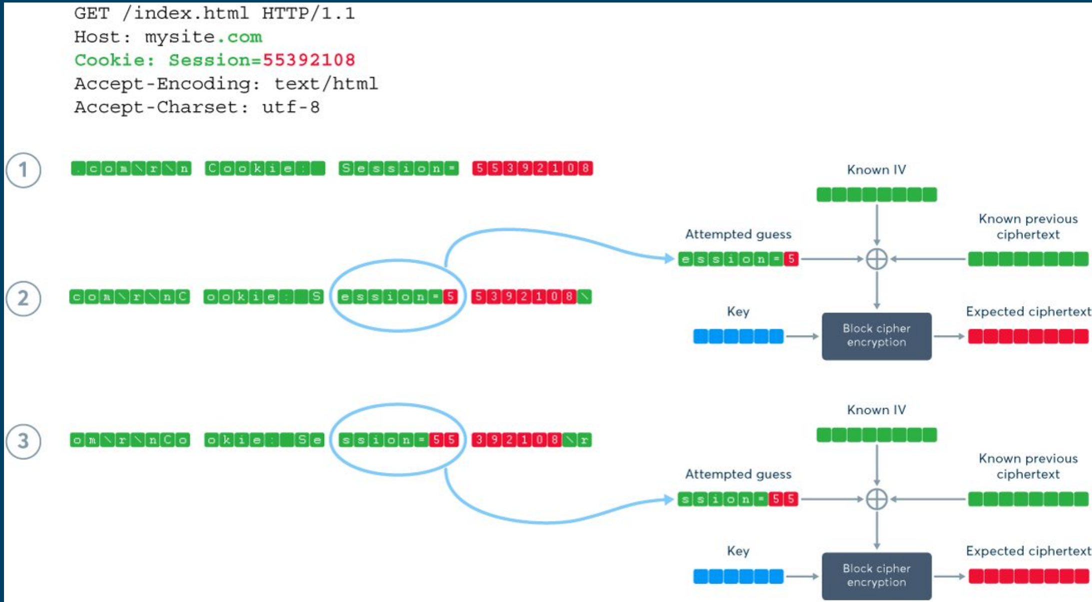
which initialization vector to pick?

- must be unpredictable.

BEAST attack on TLS1.0!

(MitM)

breaks encryption.



```
GET /index.html HTTP/1.1
Host: mysite.com
Cookie: Session=55392108
Accept-Encoding: text/html
Accept-Charset: utf-8
```

summary

IV Attack

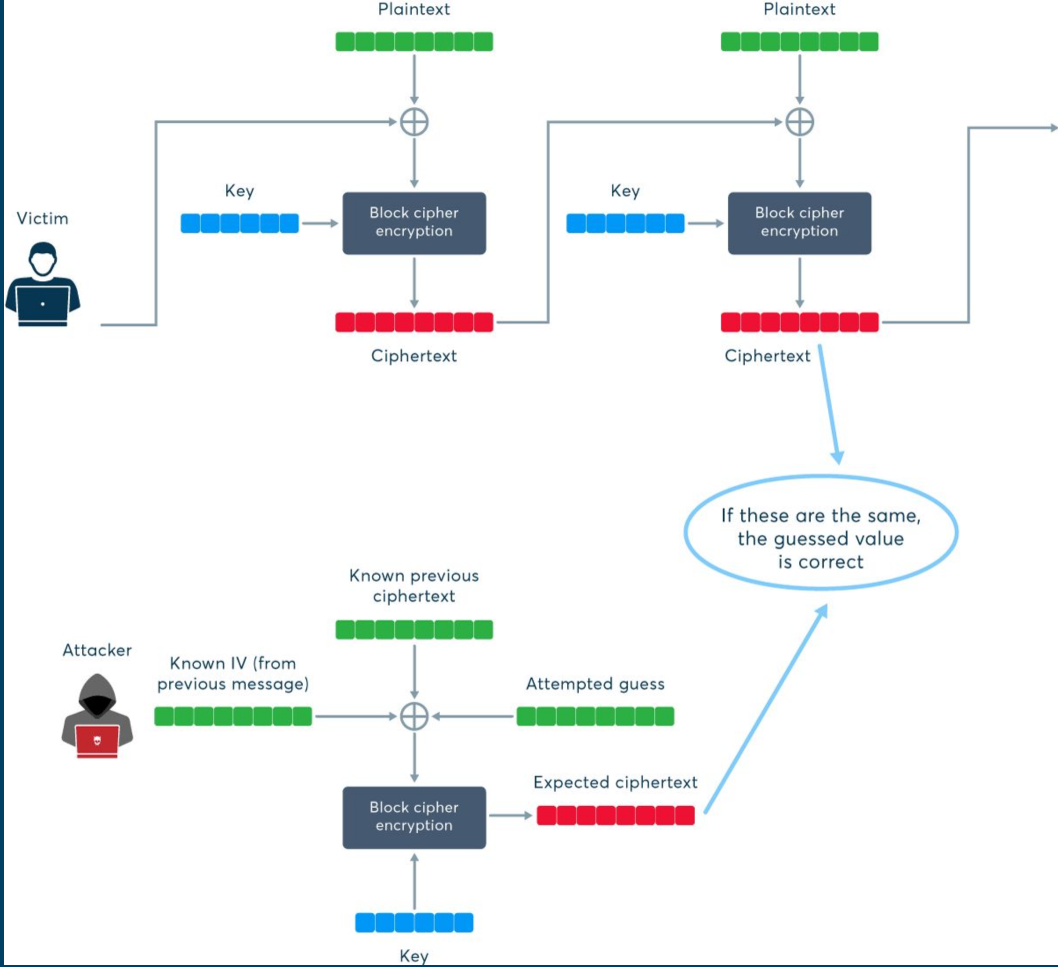
which initialization vector to pick?

- must be unpredictable.

BEAST attack on TLS1.0!

(MitM)

breaks encryption.



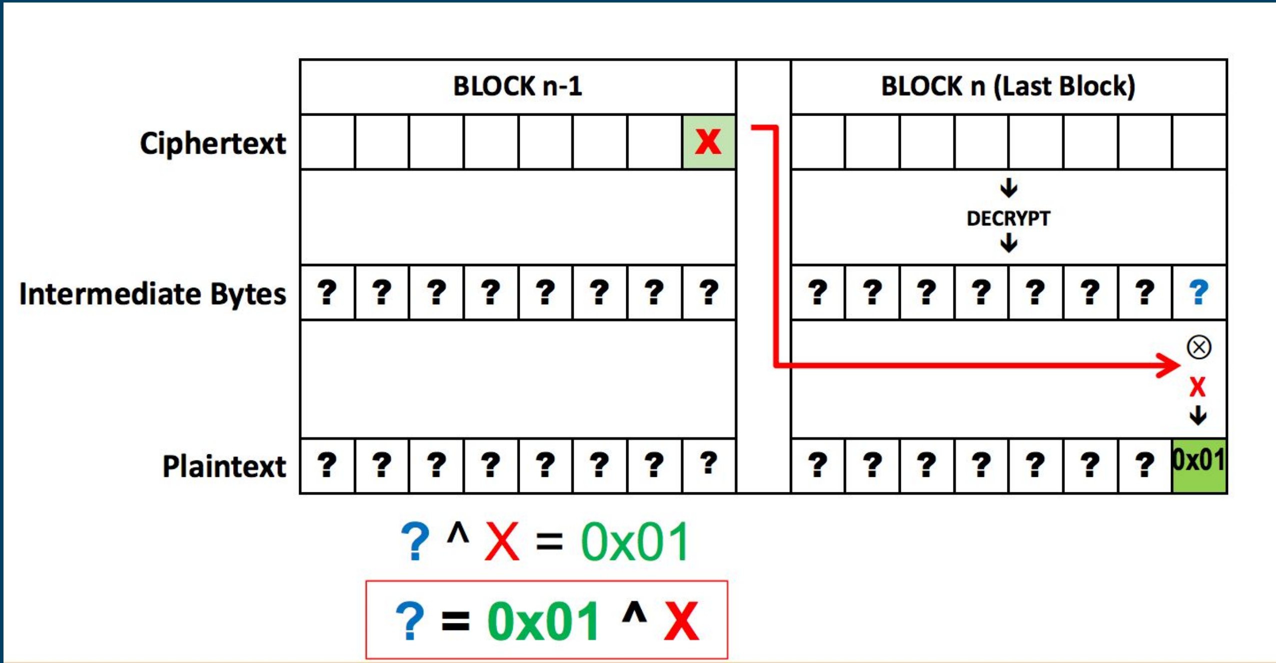
summary

CBC Padding Oracle Attack

trick server into decrypting any snooped block.

try flipping bits, ask server if padding is OK.

find valid padding ⇒ learn a byte.



worst-case 8*256 guesses.

summary

Need: Authenticated Encryption

you don’t control the wire. (Dolev-Yao adversary).
not enough to be able to exchange keys.

need to

- prevent tampering of messages,
- prevent spoofing.

we use hashing and signatures for that (next lecture!)