Crypto Week 1

Dr Tay Kian Boon

NTU, SCSE

2024 Semester 1 Week 1

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Copyright Notice

WHAT IS CRYPTOGRAPHY?

- https://www.youtube.com/watch?v=Kf9KjCKmDcU

7 Pillars of Computer Security

Computer Security

There are 7 key concepts in the field of computer security:

1. Authentication
2. Authorization
3. Confidentiality
4. Data/message integrity
5. Accountability
6. Availability
7. Non-repudiation

Computer Security

There are 7 key concepts in the field of computer security:

1. Authentication (identity -solved by eg 2FA)
2. Authorization (permission -solved by Access control list)
3. Confidentiality (secrecy contents -solved by encryption)
4. Data/message integrity (unmodified-solved by MAC-msg auth code)
5. Accountability (who is responsible -solved by log trail)
6. Availability (access –solved by adding redundancy)
7. Non-repudiation (undeniability -solved by digital signatures)

Computer Security: CRYPTO

There are 7 key concepts in the field of computer security:

1. Authentication (identity -solved by eg 2FA)
2. Authorization (permission -solved by Access control list)
3. Confidentiality (secrecy contents -solved by encryption)
4. Data/message integrity (unmodified-solved by MAC-msg auth code)
5. Accountability (who is responsible -solved by log trail)
6. Availability (access –solved by adding redundancy)
7. Non-repudiation (undeniability -solved by digital signatures)

Overview of cryptography

- Introduction to Cryptography
  - Private Key
  - Public Key
- Hash Functions and Digital Signatures

What is Cryptography - Informal

- The Science and Math of scrambling data (using a specific method with a ‘key’) into meaningless gibberish to render data incomprehensible to eavesdropper.
- In use since ancient times (>2000 yrs)

Cryptography can be found EVERYWHERE!

- Secret Messages sent between government & her embassies
- E-commerce with Amazon etc
- Cellphone calls
- Microsoft office password protection
- ATM & Smart Cards
- Whatsapp messages sent over the air
- Surfing https websites
- Your digital signatures

Overview Week 1 Lectures

- Intro Crypto
- Caesar Cipher
- Simple Substitution
- Vignere Cipher
- One-Time Pad
- Randomness

At 5:55pm a girl dropped a note to a boy

- She asked him to find out the meaning of the note by 6pm & go to LT 14 to meet her if he manages to understand the message.
- So boy has 5 minutes to figure this note:

This is the Note –What does it mean?

L OLNH X

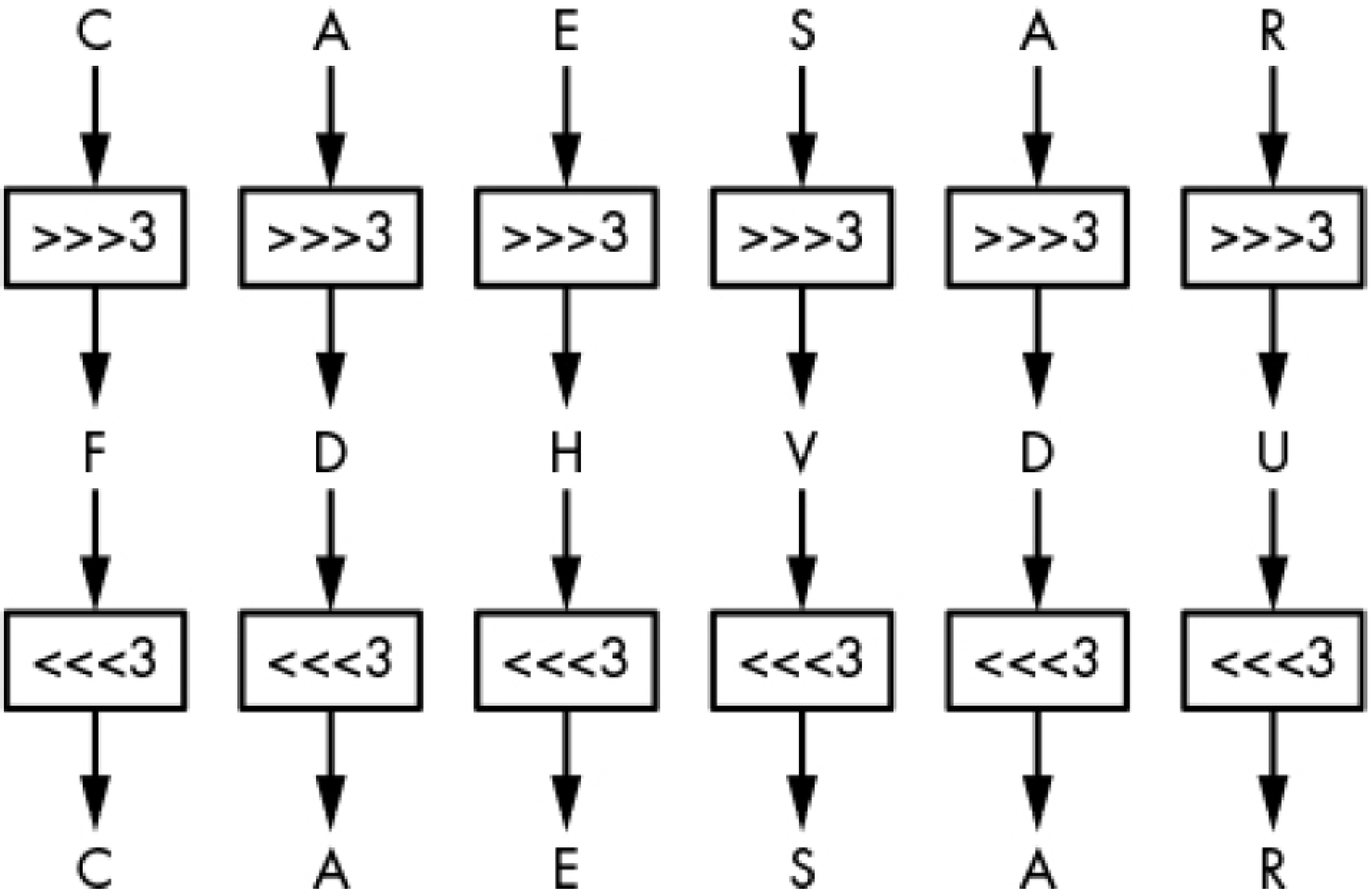


Figure 1-2: The Caesar cipher

Caesar Cipher

- Used 2000 years ago!
- Extremely easy to break, just shift 3 letters to left to break it.
- How to make cipher harder?
- Use all possible shifts, 25 of them (26 alphabets)!
- Easy for computers – instant break!
- It's the simplest mono-alphabetic encryption
    - Means each letter means uniquely some other letter (1-1 match)

Caesar Cipher – Main Weakness

- Only 25 possible keys-- trivial for computers to try all keys
- We Say Key space
- Too small!

How Big Should Key Space be ?

- Depends who you are guarding against!
- A 3 GHz Pc can roughly crack a key space of 2^34 in 1 day
- I am jumping the gun here, but I will state now:
- Present day Minimum acceptable security – 2^128!
- This is an extraordinary huge number

Ancient Famous Story

- A king wants to reward a wise man for his contribution to his kingdom
- King asked him what he wants as a reward
- Tricky question for the wise man
- Ask too much, his head might be gone
- Too little, he would have wasted an opportunity to get rich
- He told the king:

Ancient Famous Story

- Find a field and draw 8x8 squares (like a chessboard)
- Put 1 grain of rice in 1^{st} square, 2 grains in 2^{nd}, 4 grains in 3^{rd},…
- My reward: I want all the grains added up to the 64^{th} square
- King thought it's a reasonable request. He agreed in front of court.
- Ordered the servants to do what wise man had asked.
- After a while the servants reported to the king.
- WE don't have enough grain in the whole kingdom for this wise man!
- What happened?

Ancient Famous Story

- 1,2,2^2,2^3,….,2^63 – sum up total to get number of grains of rice
- Note by the time at the 20^{th} square, sum of rice grains hits almost 2^20 = roughly 1 million grains
- Put in a bag.
- Total number of bags of rice = (tutorial question)

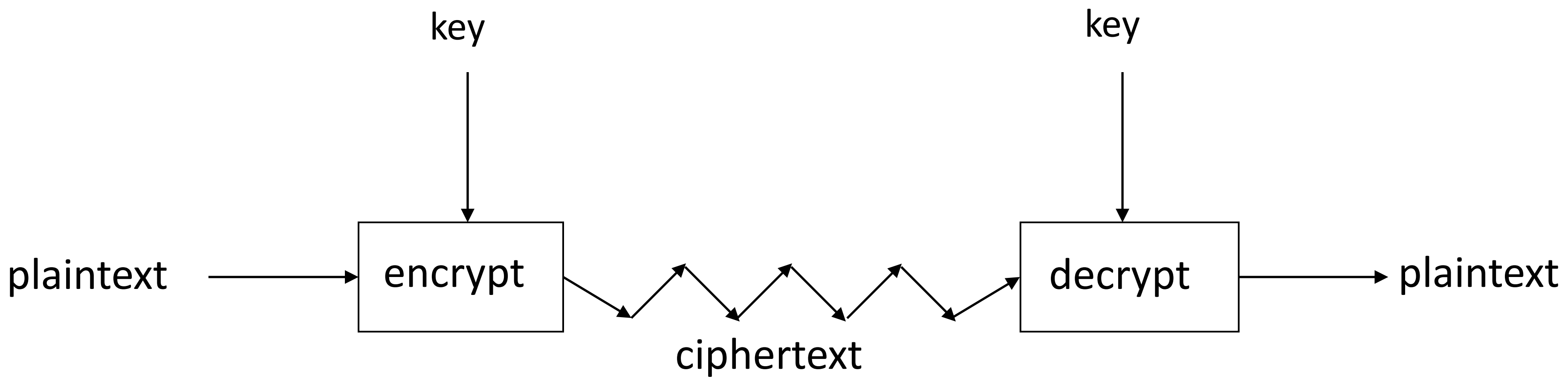
Crypto Terminology

- **Cryptology** — The art and science of making and breaking "secret codes"
- **Cryptography** — making "secret codes"
- **Cryptanalysis** — breaking "secret codes"
- **Crypto** — all of the above (and more)

How to Speak Crypto

- A *cipher* or *cryptosystem* is used to *encrypt* the *plaintext*
- The result of encryption is *ciphertext*
- We *decrypt* ciphertext to recover plaintext
- A *key* is used to configure a cryptosystem
- A *symmetric key* cryptosystem uses the same key to encrypt as to decrypt (eg Caesar Cipher)
- A *public key* cryptosystem uses a *public key* to encrypt and a *private key* to decrypt -2nd half course

Crypto as Black Box



A generic view of symmetric key crypto

Caesar Cipher-Simple Substitution

- Plaintext: fourscoreandsevenyearsago
- Key:

Plaintext	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

- Ciphertext:
  IRXUVFRUHDQGVHYHQBHDUVDJR
- Shift by 3 is "Caesar's cipher"

Caesar Cipher Decryption

- Suppose we know a Caesar's cipher is being used:

Plaintext	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

- Given ciphertext:
  VSRQJHEREVTXDUHSDQWV

- Plaintext: spongebobsquarepants

Easy to harden Caesar Cipher

- Instead of shifting all letters by 3 letters, we can scramble the 26 letters A-Z into other random permutations of A-Z.
- How many possible permutations of A-Z?
- 26! (26x25x24x….x3x2x1)
- Then encrypt accordingly based on your scrambled key.
- Decrypt using same scrambled key!

Simple Substitution: General Case

- In general, simple substitution key can be any **permutation** of letters
  - Not necessarily a shift of the alphabet
- For example

Plaintext	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext	J	I	C	A	X	S	E	Y	V	D	K	W	B	Q	T	Z	R	H	F	M	P	N	U	L	G	O

❑ Then $26! > 2^{88}$ possible keys (tutorial)-beyond BF

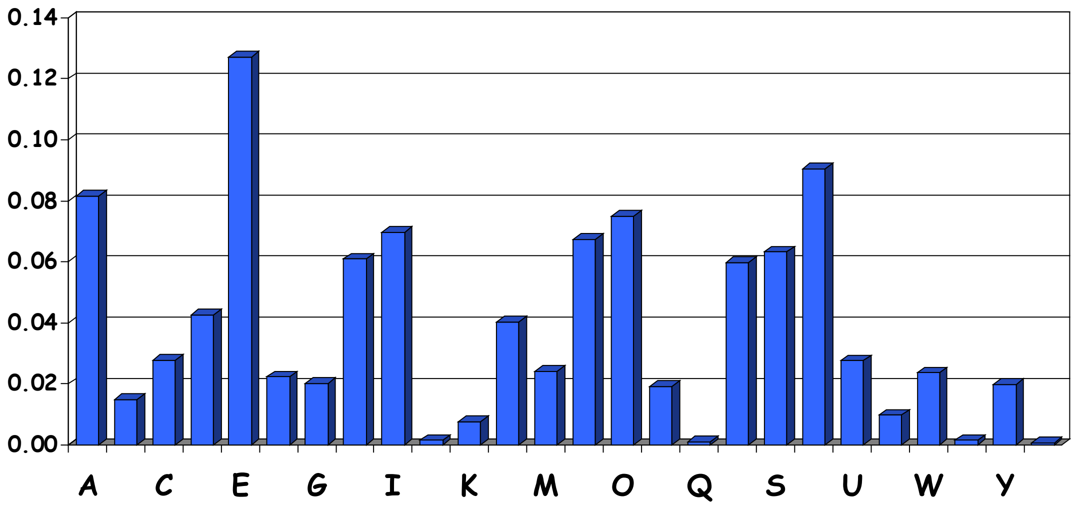
Cryptanalysis II: Be Clever

- We know that a simple substitution is used
- But not necessarily a shift by n
- Find the key given the ciphertext: (EXERCISE)

PBFPVYFBQXZTYFPBFEQJHDXXQVAPTPQJKTOYQWIPBVWLXTOXBTFXQWAXBVCXQW
AXFQJVWLEQNTOZQGGQLFXQWAKVWLXQWAEBIPBFXFQVXGTVJVWLBTPQWAEBFP
BFHCVLXBQUFEVWLXGDPEQVPQGVPPBFTIXPFHXZHVFAGFOTHFEFBQUFTDHZBQPO
THXTYFTODXQHFTDPTOGHFQPBQWAQJJTODXQHFOQPWTBDHHIXQVAPBFZQHCFW
PFHPBFIPBQWKFABVYYDZBOTHPBQPQJTQOTOGHFQAPBFEQJHDXXQVAVXEBQPEFZ
BVFOJIWFFACFCCFHQWAUVWFLQHGFXVAFXQHFUFHILTTAVWAFFAWTEVOITDHFHF
QAITIXPFHXAFQHEFZQWGFLVWPTOFFA

Cryptanalysis II

- Cannot try all 2^{88} simple substitution keys
- Can we be more clever?
- English letter frequency counts…



Letter Frequencies of English Letter

Letter	Frequency	Letter	Frequency
A	0.0817	N	0.0675
B	0.0150	O	0.0751
C	0.0278	P	0.0193
D	0.0425	Q	0.0010
E	0.1270	R	0.0599
F	0.0223	S	0.0633
G	0.0202	T	0.0906
H	0.0609	U	0.0276
I	0.0697	V	0.0098
J	0.0015	W	0.0236
K	0.0077	X	0.0015
L	0.0403	Y	0.0197
M	0.0241	Z	0.0007

Cryptanalysis II

- Ciphertext:

PBFPVYFBQXZTYFPBFEQJHDXXQVAPTPQJKTOYQWIPBVWLXTOXBTFXQWAXBVCXQWAXFQJV
WLEQNTOZQGGQLFXQWAKVWLXQWAEBIPBFXFQVXGTVJVWLBTPQWAEBFPBFHCVLXBQUFE
VWLXGDPEQVPQGVPPBFTIXPFHXZHVFAGFOTHFEFBQUFTDHZBQPOTHXTYFTODXQHFTDPTO
GHFQPBQWAQJJTODXQHFOQPWTBDHHIXQVAPBFZQHCFWPFHPBFIPBQWKFABVYYDZBOTHP
BQPQJTQOTOGHFQAPBFEQJHDXXQVAVXEBQPEFZBVFOJIWFFACFCCFHQWAUVWFLQHGFXVA
FXQHFUFHILTTAVWAFFAWTEVOITDHFHFQAITIXPFHXAFQHEFZQWGFLVWPTOFFA

- Analyze this message using statistics below

Ciphertext frequency counts-then compare this with std frequency table to map alphabets:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
21	26	6	10	12	51	10	25	10	9	3	10	0	1	15	28	42	0	0	27	4	24	22	28	6	8

Cryptanalysis: Terminology

- Cryptosystem is **secure** if best known attack is to try all keys (brute force -Exhaustive key search)
- Cryptosystem is termed '**broken**' if ***any*** shortcut attack is known without trying all keys)
- Although substitution > 26! Keys (beyond brute-force range), it succumbs to frequency analysis.
- Main weakness – letters used in English are very unevenly distributed! (eg compare Q with E)
- ***So Don't ever use Substitution ciphers!!!***

Lessons Learned So Far

- Key space must be large
- All letters must be equally likely to happen for ciphers to be maximally secure!
- Average freq for equal appearance is 1/26, which is roughly 4%
- “e” occurs 12% in English text, 3 times higher than average
- “z” appears 0.7%
- So e appears abt 18 times more frequent than z!

Vigenere Cipher

- Took 1500 years to see a meaningful improvement of the Caesar cipher - Vigenère cipher, created in the 16th century
- Used during the American Civil War by Confederate forces and during WWI by the Swiss Army, among others.

Vigenere Cipher

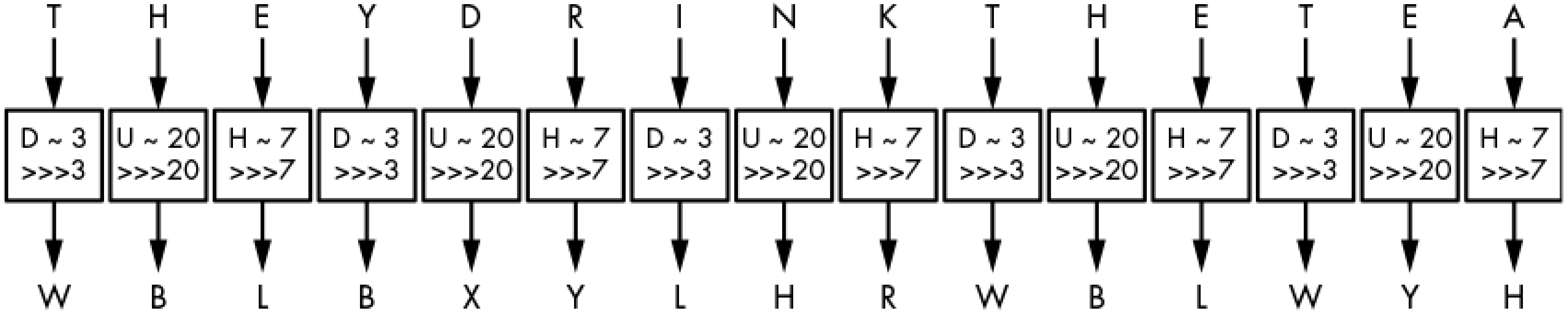
- Vigenère cipher is similar to the Caesar cipher, except that letters aren't shifted by three places but rather by values defined by a *key*, a collection of letters that represent numbers based on their position in the alphabet.
- For example, if the key is DUH, letters in the plaintext are shifted using the values 3, 20, 7
  - because *D* is 3 letters after *A*,
  - *U* is 20 letters after *A*, and
  - *H* is 7 letters after *A*.
- The 3, 20, 7 pattern repeats until you've encrypted the entire plaintext.

Vigenere Cipher

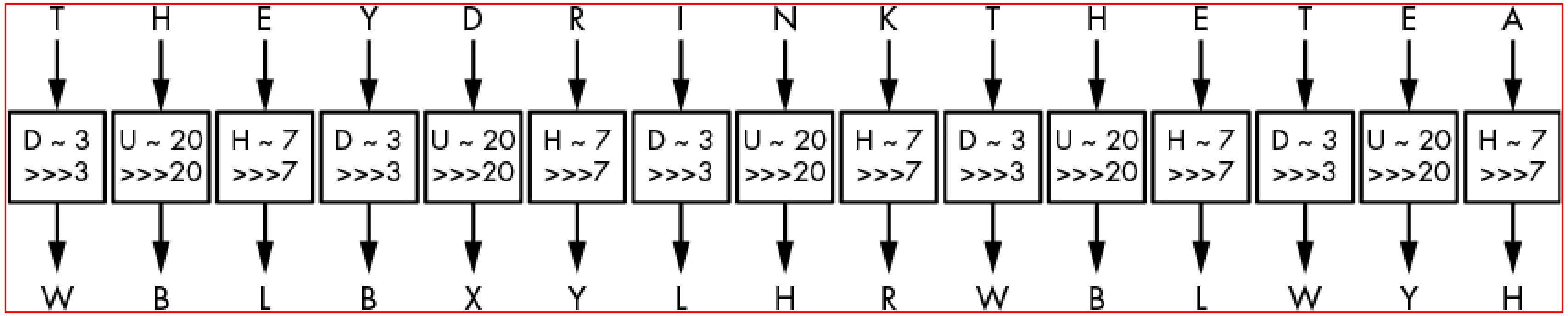
- For example, the word CRYPTO would encrypt to FLFSNV using DUH as the key:
  - *C* is shifted 3 positions to *F*,
  - *R* is shifted 20 positions to *L*, and so on.

- Figure 1-3 illustrates this principle when encrypting the sentence THEY DRINK THE TEA.

Vigenere Cipher



Vigenere Cipher : E maps to both L & Y!



Vigenere Cipher

- In this cipher, same letter can be mapped to different letters
  - Eg:3rd E & 2nd last E mapped to L & Y respectively!

- Also different letters can be mapped into a single letter
  - Eg: 2nd & 4th letter was B, mapped from H & Y
- We call this poly-alphabetic substitution
- Much more secure than simple mono-alphabetic sub!

Vigenere Cipher

- Vigenère cipher > Caesar cipher, yet it's still fairly easy to break.
- The first step to breaking it is to figure out the key's length.
- **THEYDRINKTHETEA** encrypts to
- **WBLBXYLHRWBLWYH** , with the key **DUH**.
- Notice in ciphertext WBLBXYLHRWBLWYH, the group of three letters WBL appears twice in the ciphertext at nine-letter intervals.

Vigenere Cipher

- Suggest same 3-letter word was encrypted using the same shift values, producing WBL each time.
- A cryptanalyst can then deduce that the key's length is either **9** or a value that divides nine (that is, **3**).
- Furthermore, they may guess that this repeated 3-letter word is THE and **therefore determine DUH** as a possible encryption key.

Vigenere Cipher -Status

- Long keywords implies stronger Vigenere (Tutorial)
- Short message implies stronger Vigenere (Tutorial)
- Not good enough for modern use!

Classical Ciphers

- The previous class of ciphers are all known as classical ciphers
- There are many more such ciphers, but all insecure due to computers
- Used before WW2, way before invention of computers

- Question:
- Any unbreakable ciphers, even with computers in attacker's hand?
- YES- ONE-TIME PAD!

Perfect Encryption: The One-Time Pad

- Essentially, a classical cipher can’t be secure unless it comes with a huge key, but encrypting with a huge key is impractical.
- However, the one-time pad is such a cipher, and it is the most secure cipher.
- In fact, it guarantees *perfect secrecy*:
  - even if an attacker has unlimited computing power, it’s impossible to learn anything about the plaintext except for its length.

Perfect Encryption: The One-Time Pad

- The one-time pad takes a plaintext, P, and a "random key", K, that's the same length as P and produces a ciphertext C, defined as
  $C = P \oplus K$,

where C, P, and K are bit strings of the same length and where $\oplus$ is the bitwise exclusive OR operation (XOR), defined as

$0 \oplus 0 = 0$,

$0 \oplus 1 = 1$,

$1 \oplus 0 = 1$,

$1 \oplus 1 = 0$.

Easy Encryption: The One-Time Pad

- Since $C = P \oplus K$, *we have*
- $C \oplus K = P \oplus K \oplus K$, *yielding*
- $C \oplus K = P \oplus \mathbf{0} = P$, *the plaintext!*
- *So for both encrypt & decrypt, just XOR!*
- *XOR is superfast – instantaneous encryption & decryption!*
- *So course should be over.*
- *Everyone just use one-time pad, end of story!*
- *Will answer this question shortly.*

Easy Encryption: The One-Time Pad -Letters

- Can be used for encrypting just letters too
- We need long random pads, as long as the message
- Suppose msg is YES & random pad generated is CAB
- For encryption
  - C – shift 3 to right
  - A – shift 1 to right
  - B- shift 2 to right
- **YES + CAB** (Y+C = B , becos Y shift 3 right –sequence Y,Z,A,B)
- **BFU (E shift 1 right, S shift 2 right)**

Easy Decryption: The One-Time Pad -Letters

- If user receive cipher BFU, and we know one time pad is CAB, we apply “inverse of add – that is minus
- So since encrypt means shift right, Decrypt means shift left
- For decryption
  - C – shift 3 to left
  - A – shift 1 to left
  - B- shift 2 to left
- **`BFU-CAB`** (`B-C=Y` , becos B shift 3 left –sequence B,A,Z,Y)
- **`YES (F shift 1 left, U shift 2 right)`**
- Useful to list ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMN…

OTP In Computation (letters)-encryption

- Earlier by shifting argument, we have Y + C = B
- If letters to add are large, not easy to see shifted letter by listing
- Use this trick: A=1, B=2,…,Y=25,Z=26
- Y + C (numerical) (mod 26) , α(mod26) means remainder of α when divided by 26.
- = 25 + 3 (mod 26)
- = 28 (mod 26)
- = 2
- = B , same answer as before

OTP In Computation (letters)-decryption

- Earlier by shifting argument, we have B - C = Y
- If letters to subtract are large, not easy to see shifted letter by listing
- Use this trick: A=1, B=2,..., Y=25, Z=26
- B - C (numerical) (mod 26)
- = 2 - 3 (mod 26)
- = -1 (mod 26)
- = -1 + 26 (mod 26) = 25
- = Y, same answer as before.

One-Time Pad Summary

- **Provably** secure
  - Ciphertext gives **no** useful info about plaintext
  - All plaintexts are ***equally likely, so for a 3 letter cipher, we will not be able to tell if plaintext is YES or NOT !***
- BUT, only when be used correctly
  - Pad must be random – (why? – Tutorial)
  - Pad used only once (why? -Tutorial)
  - Keep track of bits used (why? –Tutorial)
  - Pad is known only to sender and receiver
- Note: pad (key) is same size as message
- Got to generate a long random pad to your buddy so that you do not need to frequently meet up.

OPT instantaneous & unbreakable –End of Story?

- We have noted OTP operations are instantaneous using computers
- And its unbreakable.
- So should be end of story for crypto.
- Why not?

Challenges of Using OTP

- How to generate truly random LONG one-time pad (OTP)
- How to store OTP securely
- How to encrypt and decrypt securely
- Both parties have to keep in synchronization portions of pad that has already been used, so that both can keep on talking
- How to agree on new OTP if old OTP is used up or compromised

Informal Notions of Random used in crypto

- Suppose X_i is bit *i* of OTP.
- Random OTP (bits) informally means

1. $P(X_i = 0) = P(X_i = 1) = 0.5$, both equally likely
2. Successive bits are indep of each other i.e. $P(X_{i+1}/X_i) = P(X_{i+1})$

- One way out: Think of unbiased coin and repeatedly toss it & record heads or tails
- In practice: how to manufacture unbiased coin?
- Truly random source – from eg radioactive decay… slow to generate

Randomness

- Randomness is NEEDED everywhere in cryptography:
  - in the generation of secret keys,
  - in encryption schemes, and
  - even in the attacks on cryptosystems.

- Without randomness, cryptography would be impossible because all operations would become MORE predictable, and therefore insecure.

Randomness

- This section introduces you to the concept of randomness in the context of cryptography and its applications.
- We discuss pseudorandom number generators and how operating systems can produce reliable randomness, and we conclude with real examples showing how flawed randomness can impact security.

Randomness

- You’ve probably already heard the phrase “random bits,” but strictly speaking there is no such thing as a series of random bits.

- What is random is actually the algorithm or process that produces a series of random bits; therefore, when we say “random bits,” we actually mean randomly generated bits.

Randomness –how they look like

Is the 8-bit string 11010110 more random than 00000000?

Randomness

- To most people, the 8-bit string 11010110 is more random than 00000000
- **WRONG!**, Both have the same chance of being generated (namely, $(1/2)^8 = 1/256$)
- The value 11010110 looks more random than 00000000 because it has the signs typical of a randomly generated value. That is, 11010110 has no obvious pattern.
- But you are actually requiring the bits to follow that specific pattern 11010110 with same probability of occurrence.
- HHTHTHHT –imagine u r asked to toss exactly this sequence of toss!

Randomness

- When we see the string 11010110, our brain registers that it has about as many zeros (three) as it does ones (five), just like 55 other 8-bit strings (11111000, 11110100, 11110010, and so on), but only one 8-bit string has eight zeros.
- Because the pattern 3-zeros-and-5-ones is more likely to occur than the pattern 8-zeros, we identify 11010110 as random and 00000000 as non-random, and if a program produces the bits 11010110, you may think that it’s random, even if it’s not.
- Conversely, if a randomized program produces 00000000, you’ll probably doubt that it’s random.

Randomness

- This example illustrates two types of errors people often make when identifying randomness:
- **Mistaking non-randomness for randomness** Thinking that an object was randomly generated simply because it *looks* random.
- **Mistaking randomness for non-randomness** Thinking that patterns appearing by chance are there for a reason other than chance.
- The distinction between random-looking and actually random is crucial. Indeed, in crypto, non-randomness is often synonymous with insecurity.

Informal Notions of Random used in crypto

- Please note crypto notion of randomness needed is much more stringent than randomness needed in simulations (monte-carlo) used in video games and computing probabilities of complicated events
- It is also more stringent than random numbers generated from pseudo-random number generators
- This type of generation won't produce truly robust random numbers needed for crypto.
- Later in course I will suggest some famously good CSPRNG, crypto-secure pseudo random number generators