

Cryptography

CMSC/MATH 456 : Spring 2020

ABOUT THE COURSE

Instructor: Gorjan Alagic (galagic@umd.edu); ATL 3102, office hours: by appointment

Textbook: *Introduction to Modern Cryptography*, Katz and Lindell;

Webpage: alagic.org/cmsc-456-cryptography-spring-2020/ (check for updates);

Piazza, Gradescope, ELMS: coming soon.

TAs:

- Elijah Grubb (egrubb@cs.umd.edu) 11am-12pm TuTh (Iribe);
- Justin Hontz (jhontz@terpmail.umd.edu) 1pm-2pm MW (Iribe);

Our designated spot: shared open area across from IRB 5234

Additional help:

- Chen Bai (cbai1@terpmail.umd.edu) 3:30-5:30pm Tu (2115 ATL, starting Feb 4)
- Bibhusa Rawal (bibhusa@terpmail.umd.edu) 3:30-5:30pm Th (2115 ATL, starting Feb 6)

ABOUT THE COURSE

The fun stuff (see syllabus for details.)

Grading: 40% homework, 30% midterm exam, 30% final exam

Homework (~ 10 sets):

- collaboration allowed, but must write up your own;
- no late homework whatsoever (but lowest grade will be dropped);
- first set distributed 2nd week.

Exams:

- closed book/device, one two-sided page of notes;
- midterm March 31st;
- final May 18th.

ABOUT THE COURSE

Approximate course plan:

Topic	Dates
Intro and symmetric-key cryptography (8 lectures)	January 28 – February 20
RSA and Diffie-Hellman (4 lectures) <i>Carl Miller</i>	February 25 – March 5
Secret sharing (2 lectures) <i>Bill Gasarch</i>	March 10 - 12
Midterm review and midterm; 2 fun guest lectures	March 24 – Apr 2
Public-key cryptography II, advanced topics (10 lectures)	Apr 7 – May 12

WHAT IS THIS COURSE ABOUT?

What it IS about:

- theoretical cryptography;
- “replacing trust with mathematics” (- Boaz Barak);
- exploring limits of what is possible *in principle*;
- fundamental tasks: encryption, authentication

What we WILL do:

- define concepts rigorously, prove theorems
- analyze cryptosystems and attacks in terms of “possible in principle” vs “impossible, even in principle”

What it is NOT about:

- practical IT security;
- hacking, spoofing, fishing, DOS attacks, etc.;
- real-world implementations;
- bleeding edge theory: obfuscation, quantum FHE

What we will NOT do:

- implement real cryptosystems or attacks
- analyze cryptosystems and attacks in terms of concrete costs (e.g., 20 minutes vs 2 hours on a four-core Xeon with 32GB RAM...?)

WHAT IS THIS COURSE ABOUT?

What background should you refresh?

- Discrete probability: random variables and events, conditional probability, expectation, etc.;
- Theory of computation: basic algorithms and programming concepts, asymptotic analysis (O -notation), etc;
- Mathematical rigor: formal definitions, notation, theorems, proofs;

Basically, the stuff you (hopefully) did in discrete math.

I. THE (SKETCHY) HISTORY OF CRYPTO

Reading: xv – p.24.

WHAT IS CRYPTOGRAPHY?

What is cryptography?

How will we study it in this course?

Why will we do it that way?

To answer all this: need to first look at how crypto has been done for most of history.

This is not a “boring history lesson” you can ignore!

- people were very clever before computers too!
- develop intuition about what “good crypto” and “bad crypto” look like;
- learn basic techniques for breaking cryptosystems;
- understand *why* we now do crypto the way we do it;
- some historical schemes still crop up in modern crypto!

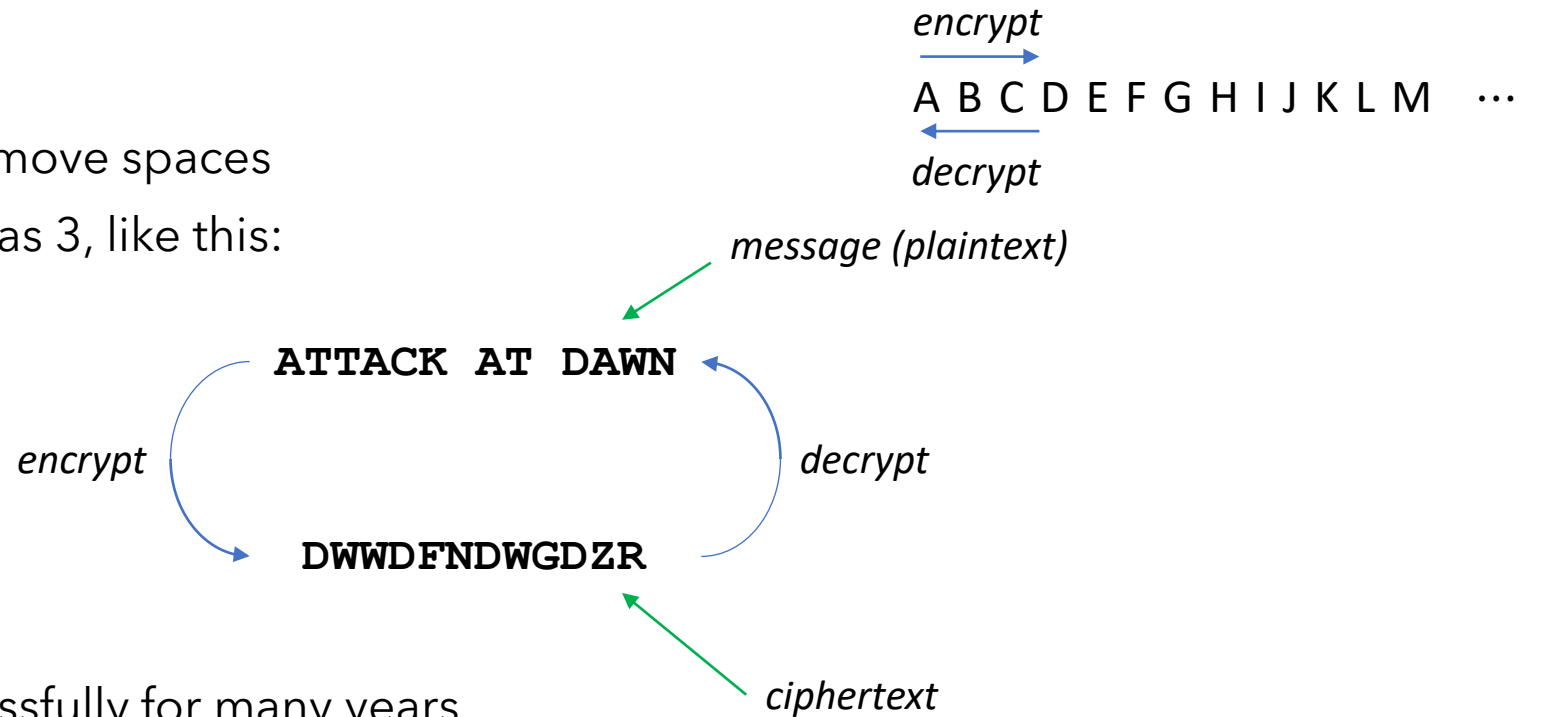
...and besides, history is awesome!

HISTORICAL CIPHERS: CAESAR CIPHER

Caesar cipher

Goal: "send secret messages"

- shift each letter in the message, remove spaces
- Caesar himself used this; his key was 3, like this:



- apparently, Caesar used this successfully for many years
- in 2011, used in a plot to attack airliners (no, really.)

Is it secure?

HISTORICAL CIPHERS: CAESAR CIPHER

No! Brute force keysearch:

Suppose you see the message "dwwdfndwrqfh" (but you don't know Caesar's key.)

Try all possible decryption keys:

0	dwwdfndwrqfh
-1	cvvcemcvqpeg
-2	buubdlbupodf
-3	attackatonce
-4	zsszbjzsnmbd
-5	yrryaiyrmlac
⋮	⋮

Only 26 possibilities, so easy! (The 2011 plot failed and the plotters were caught.)

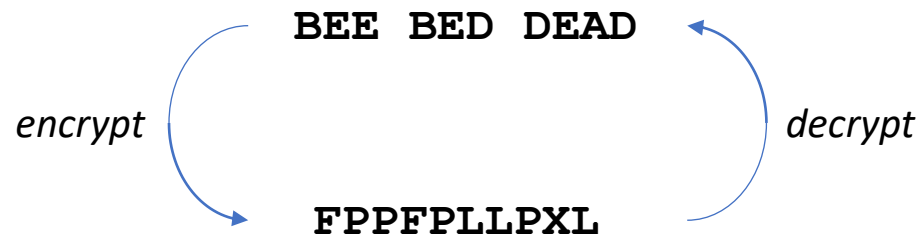
Must have: big keyspace.

HISTORICAL CIPHERS: SUBSTITUTION CIPHER

Substitution cipher

- each letter of the alphabet is mapped to another, randomly selected letter
- for example:

key
 $A \mapsto X$
 $B \mapsto F$
 $C \mapsto D$
 $D \mapsto L$
 $E \mapsto P$
...
...



- used in 1586 plot by Mary, Queen of Scots to assassinate Queen Elizabeth and install Mary as queen;
- Mary used the cipher to instruct her collaborators to kill the queen!

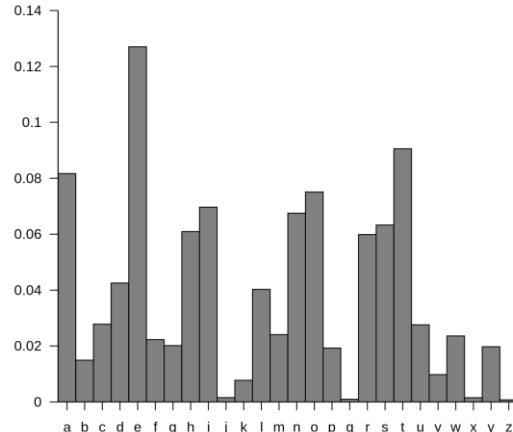
Key space: $26! \approx 10^{26}$

Is it secure?

HISTORICAL CIPHERS: SUBSTITUTION CIPHER

Unfortunately for Mary, an Arab philosopher named Al-Kindi broke this cipher over 700 years prior.

Frequency analysis

- plot average frequency of letters in spoken English;
 - do the same for the encrypted message;
 - permute the letters to make the plots match up;
 - the resulting permutation is (probably close to) the key!
- 
- | Letter | Frequency |
|--------|-----------|
| a | 0.082 |
| b | 0.015 |
| c | 0.028 |
| d | 0.042 |
| e | 0.128 |
| f | 0.022 |
| g | 0.020 |
| h | 0.060 |
| i | 0.070 |
| j | 0.005 |
| k | 0.010 |
| l | 0.040 |
| m | 0.025 |
| n | 0.068 |
| o | 0.075 |
| p | 0.020 |
| q | 0.005 |
| r | 0.060 |
| s | 0.065 |
| t | 0.090 |
| u | 0.028 |
| v | 0.010 |
| w | 0.025 |
| x | 0.005 |
| y | 0.020 |
| z | 0.005 |
- Mary's messages were intercepted and broken with frequency analysis;
 - using the key, the messages were even changed to get her to reveal her conspirators (*authentication?*);
 - based on this, Mary was found guilty and beheaded.

Crypto mattered a lot even in 1586!

HISTORICAL CIPHERS: VIGENÈRE CIPHER

If Mary had a better cryptographer, she would have used Vigenère cipher (discovered a few years prior.)

	YOU CAN EXPECT NO HELP FROM THIS SIDE OF THE RIVER
+	VICTOR VICTOR VICTOR VICTOR VICTOR VICTOR VICT
<hr/>	
=	UXXWPFAGSYRLJXKYAHBARGIZEBVCSWKOWBTJEEHL

+ means add letters
as numbers (mod 28)

Used by the Confederacy in the U.S. Civil War.

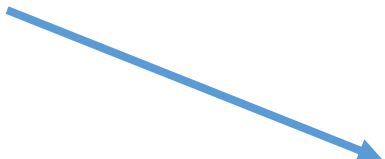


Is it secure?

HISTORICAL CIPHERS: VIGENÈRE CIPHER

YOU CAN EXPECT NO HELP FROM THIS SIDE OF THE RIVER
+ VICTOR VICTOR VICTOR VICTOR VICTOR VICTOR VICT
= UXXWPFAGSYRLJXKYAHBARGIZEBVCSWKOWBTJEEHL

Guess the length of the passphrase. Then split up ciphertext:



UXXWPF
AGSYRL
JXKYAH
BARGIZ
EBVCSW
KOWBTJ
EEHL

- each column is a Caesar cipher; 26 choices there, but $26^6 \approx 309$ million total! No good...
- instead, frequency analysis with a twist: plot of first column = English alphabet translated by v !

It took over 300 years for someone to figure this out and break Vigenère. (So Mary might have gotten away with it!)

WHAT WENT WRONG?

Lessons learned

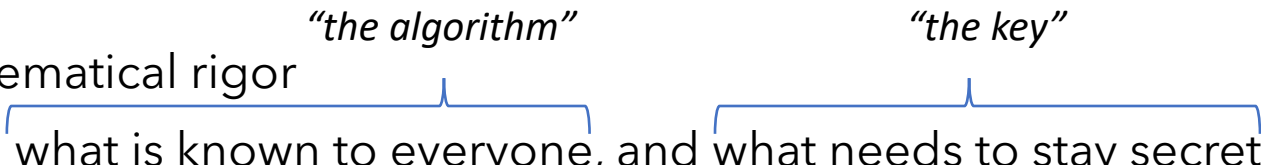
- key space needs to be large (prevent brute force key search);
- scheme needs to resist frequency analysis, sometimes in non-obvious ways;
- what else? Is that enough?
- ... as it turns out, it's not; throughout history, each attempt to "patch" was eventually circumvented.
- (fun read: Enigma in WW2.)

The first "unbreakable" cipher was not discovered until 1882!

- *why did it take so long?*
- people have been clever for a long time; that didn't start in 1882;
- modern crypto *seems to be* a lot more "stable" than the stuff we discussed above
- what changed?
- (also: if there's an unbreakable cipher, what is left to do? As we will see, a lot!)

WHAT DO WE DO DIFFERENTLY NOW?

The modern (theoretical) approach (~1970s on)

- emphasis on mathematical rigor
- formal definitions :  *“the algorithm”* and *“the key”* : what is known to everyone, and what needs to stay secret?
- formal definitions : what exactly is the cryptosystem trying to achieve?
- formal definitions : when is a cryptosystem considered “secure”?
- security proofs: mathematical theorems establishing security (with important caveats!)

Kerckhoffs's principle
A cryptosystem should be secure even if everything about the system, except the key, is public knowledge.

... and lots and lots of clever cryptographic (design) work and cryptanalytic (attack) work!

These will be the ideas that we will explore in this course.

II. (SIMPLE) ENCRYPTION

Reading: Ch.2 (p.25-40)

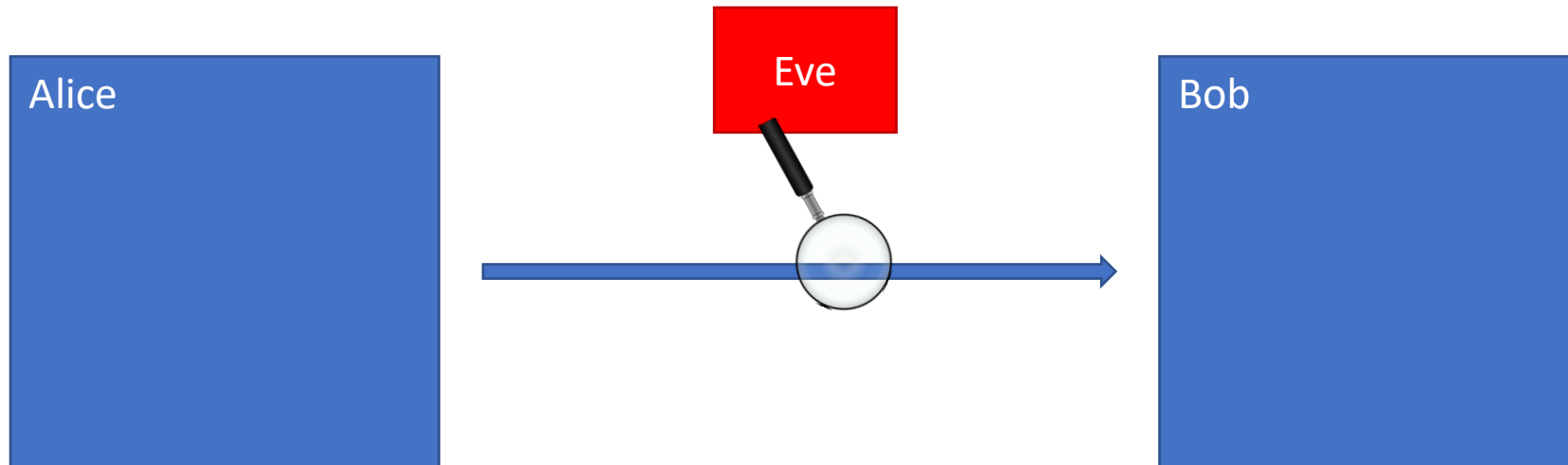
ENCRYPTION: THE SETTING

Task: Alice wants to send a single message to Bob, *but Eve is watching the channel.*

Assumptions:

- Alice and Bob can share a secret in advance;
- they have their own private spaces;
- Alice can send only one transmission, on a single channel;
- Eve (eavesdropper) can observe *everything* that is transmitted on that channel.
- *Eve cannot do anything else.*

Wait, why not just use this "assumption" to send the message?



ENCRYPTION SCHEMES

Generic approach to this task:

- generate key via some algorithm:
- encrypt via some algorithm:
- decrypt via some algorithm:

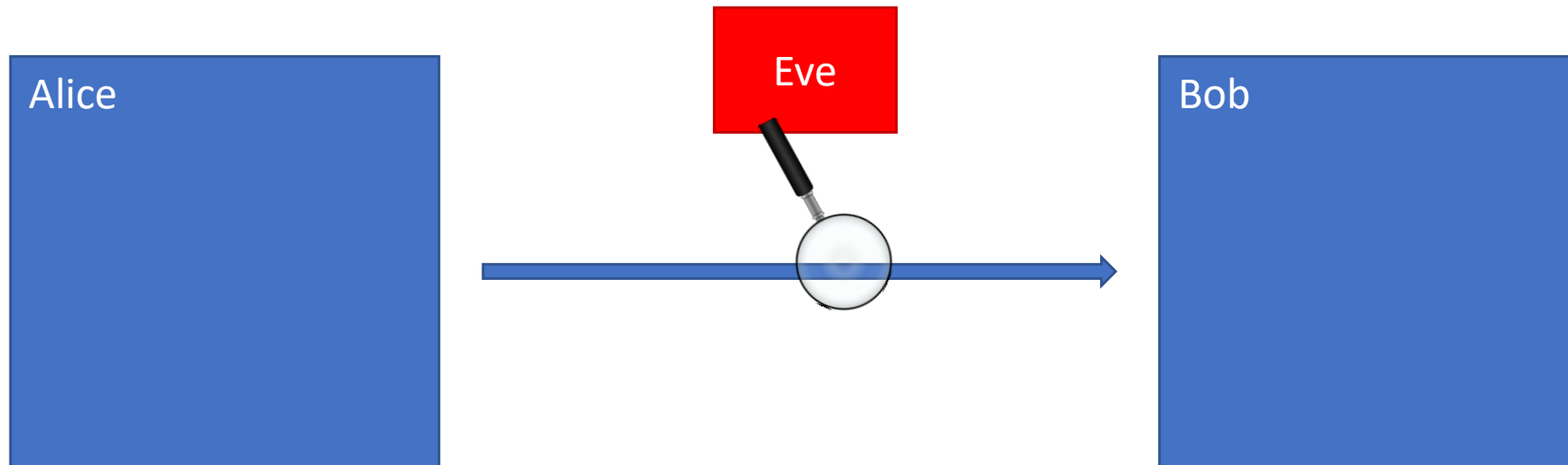
The triple (**KeyGen**, **Enc**, **Dec**) is called an *encryption scheme*.

Message-independent distribution.

$k \leftarrow \mathbf{KeyGen}$

$c \leftarrow \mathbf{Enc}_k(m)$

$m \leftarrow \mathbf{Dec}_k(c)$

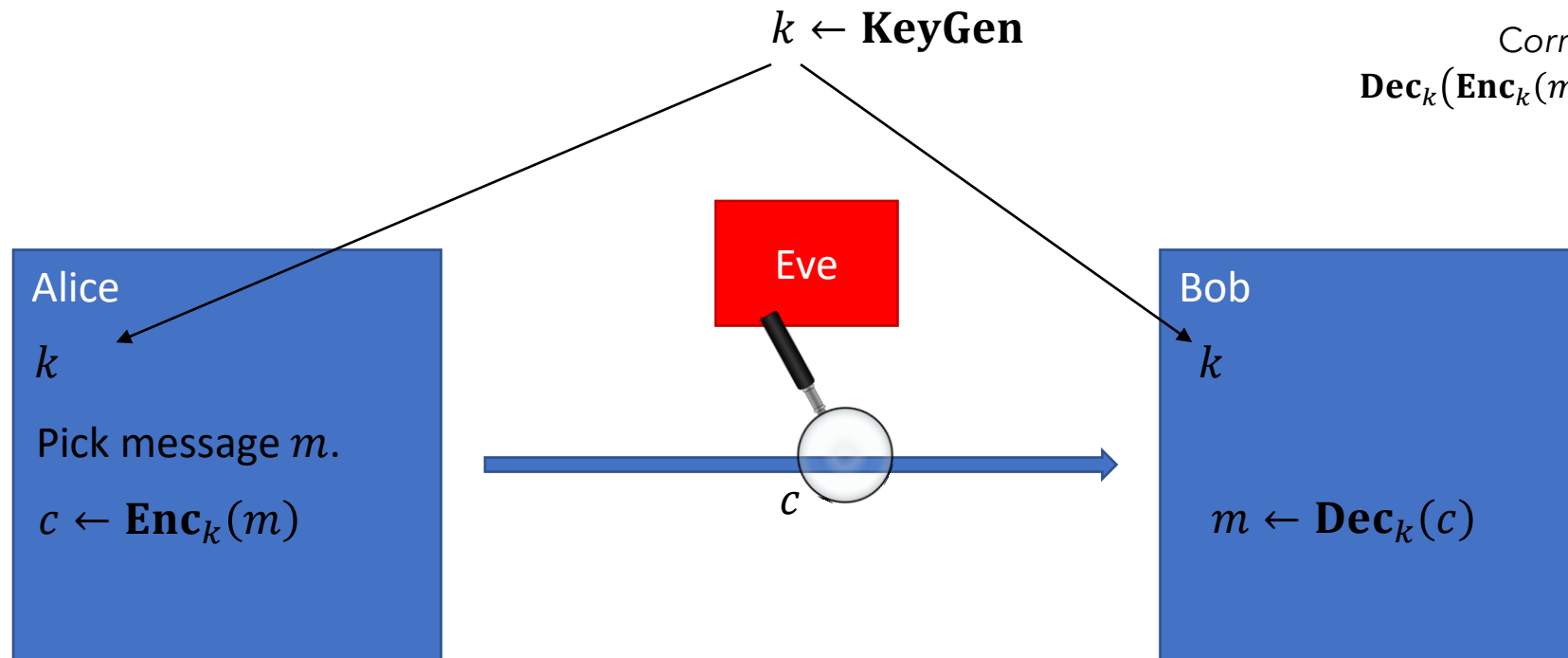


ENCRYPTION SCHEMES

Generic approach to this task:

- generate key via some algorithm: $k \leftarrow \mathbf{KeyGen}$
- encrypt via some algorithm: $c \leftarrow \mathbf{Enc}_k(m)$
- decrypt via some algorithm: $m \leftarrow \mathbf{Dec}_k(c)$

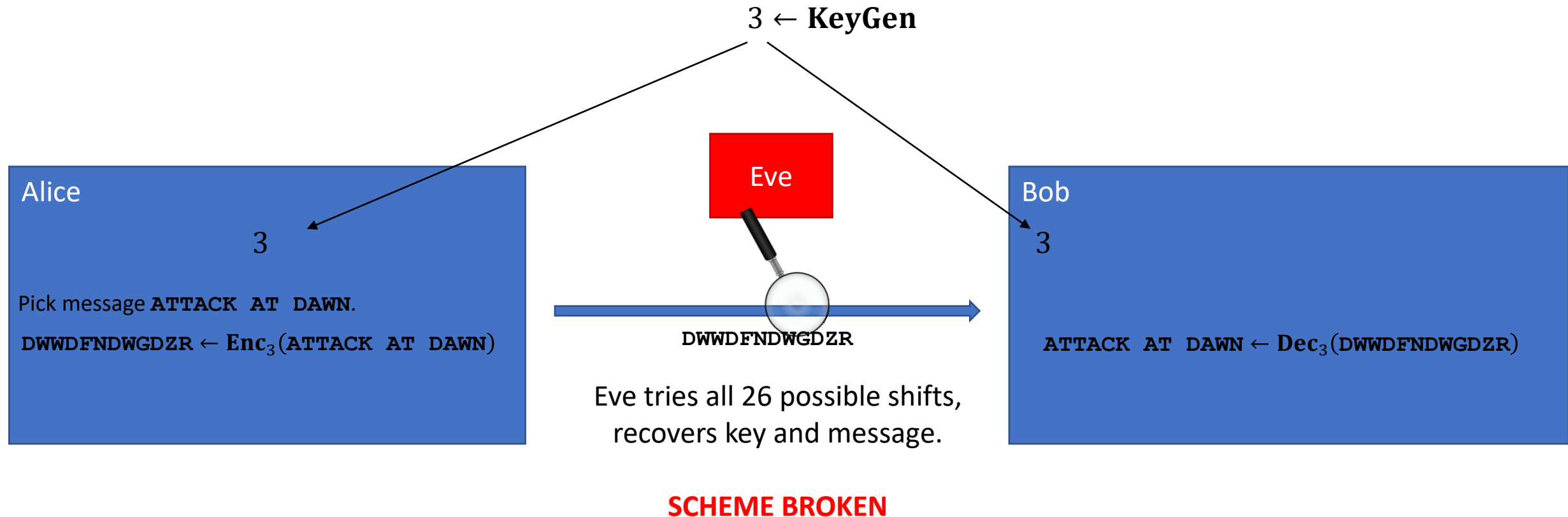
The triple $(\mathbf{KeyGen}, \mathbf{Enc}, \mathbf{Dec})$ is called an *encryption scheme*.



ENCRYPTION SCHEMES

Examples

Let's look at our initial Caesar's cipher example.



ENCRYPTION SCHEMES: ONE-TIME PAD

Examples: one-time pad (Vernam cipher, ~1882)

- *Key generation* : sample uniformly random $k \in \{0,1\}^n$
- *Encryption* : $\mathbf{Enc}_k(m) = m \oplus k$
- *Decryption* : $\mathbf{Dec}_k(c) = c \oplus k$;

(note 1: messages are interpreted as bitstrings.)

(note 2: key length = message length = ciphertext length = n .)

Bitwise XOR (+ mod 2):

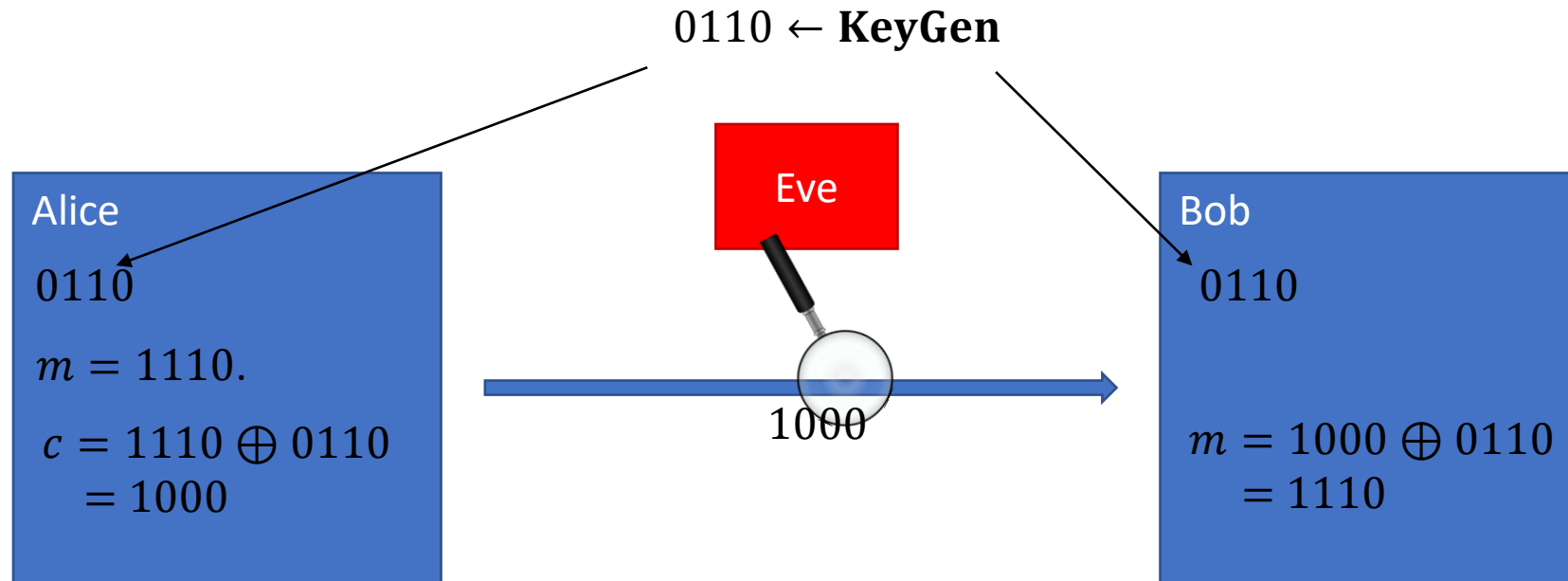
$$0 \oplus 0 = 0$$

$$0 \oplus 1 = 1$$

$$1 \oplus 1 = 0$$

Check correctness:

$$\mathbf{Dec}_k(\mathbf{Enc}_k(m)) = (m \oplus k) \oplus k = m$$



ENCRYPTION

Is the one-time pad (OTP) secure?

What does it mean to be secure?

- impossible to recover the key?

Consider this scheme:

- **KeyGen** outputs a random string $k \in \{0,1\}^n$.
- **Enc** $_k(m) = m$.

totally insecure!

- impossible to recover message?

Consider a scheme like this:

$$\text{Enc}_k(m) = \underbrace{m_1 m_2 m_3 m_4}_{\text{first 4 bits leak}} \underbrace{*****}_{\text{rest are secret (somehow)}}.$$

first 4 bits leak

rest are secret (somehow)

Or something more insidious...

... like leaking the parity of m ?

More generally: what do we mean by “impossible to recover”?

A LITTLE PROBABILITY

Random variables

- outcome of some random experiment; denoted with capital letters: X, Y, M, C, \dots ;
- comes with a probability distribution; denoted with calligraphic letters: $\mathcal{X}, \mathcal{Y}, \mathcal{M}, \mathcal{C}, \dots$;
- possible values (or samples) denoted with lowercase letters: x, y, m, c, \dots ;
- **event**: a subset of the sample space of some random experiment.

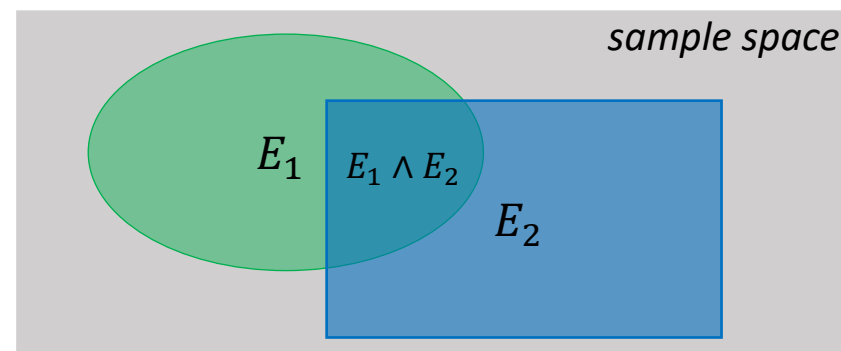
Examples

Let X be uniformly random on $\{0,1\}^n$. Then $\Pr[X = x] = 2^{-n}$ for all $x \in \{0,1\}^n$.

$\begin{array}{cc} RV & value \\ \downarrow & \downarrow \\ \underbrace{\Pr[X = x]}_{event} & \end{array}$

Let X be uniformly random on $S = \{0,1,2,3,4\}$. Then $\mathbf{E}[X] = \sum_{s \in S} \Pr[X = s] \cdot s = \frac{1}{5}(0 + 1 + 2 + 3 + 4) = 2$.

Let E_1, E_2 be events. Then $\Pr[E_1|E_2] := \frac{\Pr[E_1 \wedge E_2]}{\Pr[E_2]}$.



ENCRYPTION SECRECY: CANDIDATE I

Secrecy: a good attempt.

"The adversary never learns anything *new* about the plaintext by looking at the ciphertext."

This is called *semantic security*. A very informal way to state it:

Definition 1. (very informal) An encryption scheme is **semantically secret** if, for all choices of:

- adversary A ,
- message m ,
- "prior information" function g , and
- "target information" function f ,

the following property holds:

$$\Pr[f(m) \leftarrow A(g(m), \mathbf{Enc}_k(m))] = \Pr[f(m) \leftarrow A(g(m))].$$

"Look, I studied the ciphertext carefully and learned something interesting about the plaintext!"

"Actually, you could have learned it without looking at the ciphertext at all!"

Super complicated! And we haven't even properly formalized it...

ENCRYPTION SECRECY: CANDIDATE II

Secrecy: “perfect secrecy” (KL p.29)

Definition 2. An encryption scheme (**KeyGen**, **Enc**, **Dec**) is **perfectly secret** if, for every plaintext distribution \mathcal{M} , every plaintext m , and every ciphertext c ,

$$\Pr[M = m \mid C = c] = \Pr[M = m].$$

“The probability that the plaintext is some particular m , if you DID see the ciphertext.”

“The probability that the plaintext is some particular m , if you DID NOT see the ciphertext.”

What does the notation mean? This is the random experiment:

- Sample a uniformly random key $k \leftarrow \mathbf{KeyGen}$;
- Get a sample from the random variable M with distribution \mathcal{M} ;
- Run encryption \mathbf{Enc}_k on the sample; the result is the random variable C ;

Sounds like semantic secrecy, but without all the baggage. Good enough?

ENCRYPTION SECRECY: CANDIDATE III

Secrecy: what about this one?

Definition 3. An encryption scheme (**KeyGen**, **Enc**, **Dec**) is **perfectly secret** if, for every plaintext distribution \mathcal{M} , every plaintext pair m, m' , and every ciphertext c ,

$$\Pr_k[\mathbf{Enc}_k(m) = c] = \Pr_k[\mathbf{Enc}_k(m') = c]$$

Something like: "If the key is secret, then the distribution of ciphertexts is independent of the message."

Looks pretty good too. Is it right?

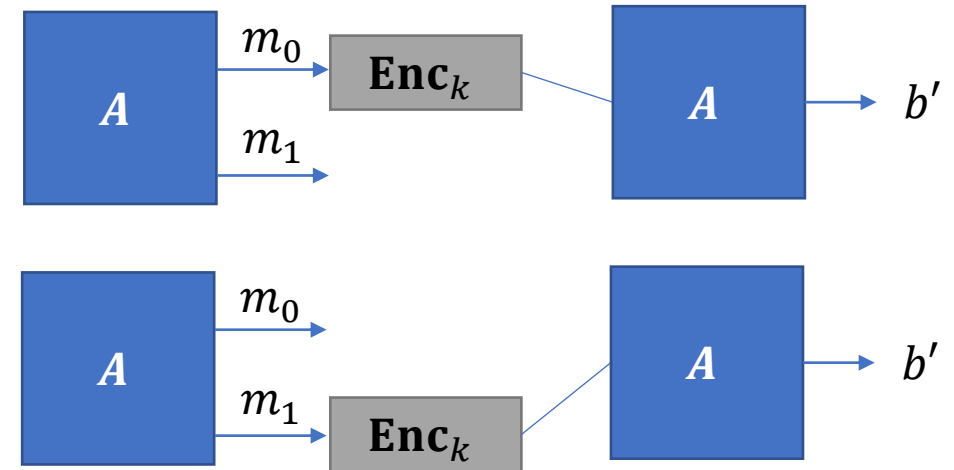
ENCRYPTION SECRECY: CANDIDATE IV

Secrecy: let's do it with an experiment (or game, if you like.)

Indistinguishability experiment (IND).

1. we sample a key $k \leftarrow \mathbf{KeyGen}$;
2. adversary (Eve) \mathbf{A} outputs two messages m_0, m_1 ;
3. we flip a uniform coin $b \leftarrow \{0,1\}$;
4. we give \mathbf{A} the ciphertext $c \leftarrow \mathbf{Enc}_k(m_b)$;
5. \mathbf{A} outputs a bit b' .

We say \mathbf{A} wins if $b = b'$.



Definition 4. An encryption scheme $(\mathbf{KeyGen}, \mathbf{Enc}, \mathbf{Dec})$ has **perfectly indistinguishable ciphertexts** if, for every adversary \mathbf{A} ,

$$\Pr_k[\mathbf{A} \text{ wins IND}] = \frac{1}{2}.$$

ENCRYPTION SECRECY

Surprise: (I know, not really...)

Theorem 1. Definitions 1-4 are all equivalent. In particular,

semantic secrecy \Leftrightarrow perfect secrecy \Leftrightarrow perfectly indistinguishable ciphertexts.

- proof is not very hard; some parts in book, others in homework;
- studying how the proofs work is worthwhile.

This is awesome:

- each definition comes with some natural intuition: a secure scheme *should* satisfy it;
- that they are all equivalent is an indication that we are on to a *good notion*;
- the definitions are reasonably different in form; as a result, they will be useful in different situations;
- some have an explicit adversary, others do not!
- you can pick which one to use depending on context.

ENCRYPTION SECRECY OF ONE-TIME PAD

Example: one-time pad.

Which definition should we use? Let's do **Definition 3**: $\Pr_k[\mathbf{Enc}_k(m) = c] = \Pr_k[\mathbf{Enc}_k(m') = c]$.

Simple argument:

- $k \leftarrow \{0,1\}^n$ is a uniformly random bitstring.
- for any fixed x , observe that $x \oplus k$ is also uniformly random in $\{0,1\}^n$.
- in particular, $\Pr_k[x \oplus k = c] = 2^{-n}$ for any $c \in \{0,1\}^n$.
- but this holds *for any fixed* x . In particular, it holds for both m and m' from the setup in Definition 3.

It follows that

$$\Pr_k[\mathbf{Enc}_k(m) = c] = \frac{1}{2^n} = \Pr_k[\mathbf{Enc}_k(m') = c]$$

So the one-time pad is *perfectly secret*, and (by **Theorem 1**) all those other great things too.

So we have *perfectly secure, unbreakable encryption*! Is the course over?



End of Lecture 1.