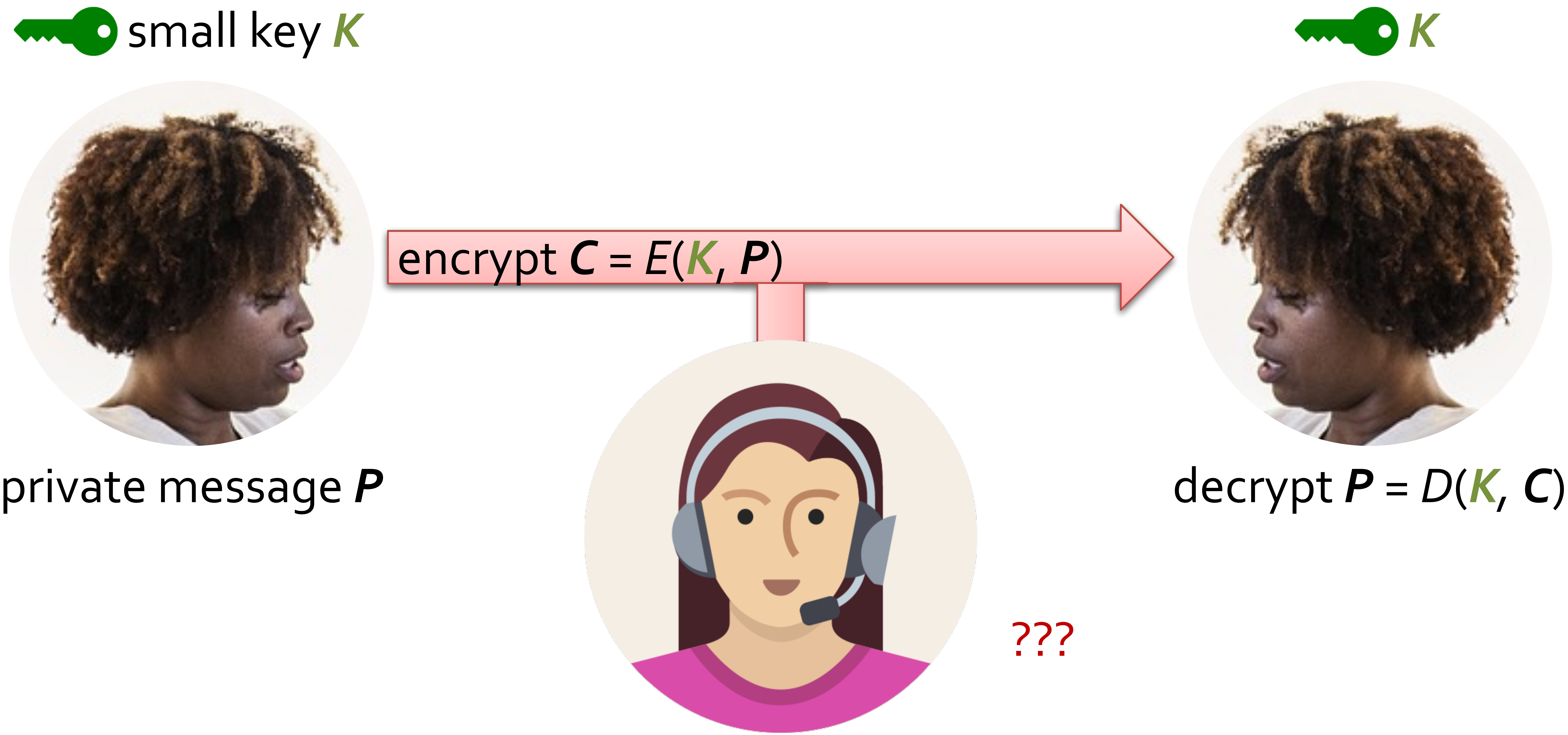
Lecture 2: Unpredictability

- Pick up a syllabus if you didn't get one last time
- Exam dates: 2/20, 3/31, and 5/5 (mark on your calendar now!)
- Homework 1 has been posted on Piazza, due on Monday 1/27
- Textbook reading: *The Block Cipher Companion*, chapter 1
- Summer teaching opportunity for high school outreach program

“Cryptography is about **communication** in the presence of an **adversary**.”

–Prof. Ron Rivest, MIT

(For now) protecting data confidentiality at rest



“Confidentiality xor authenticity is **not possible**.
If you don't have both, often you don't have either.”

–Prof. Matthew Green, Johns Hopkins

Course outline

1. Protecting data confidentiality at rest
2. Attacking data confidentiality at rest → Exam 1
3. Adding data integrity
4. Protecting data in transit → Exam 2
5. Protecting data during use
6. Designing symmetric ciphers

How can Alice encode messages so Eve can't read them?

Substitute each character

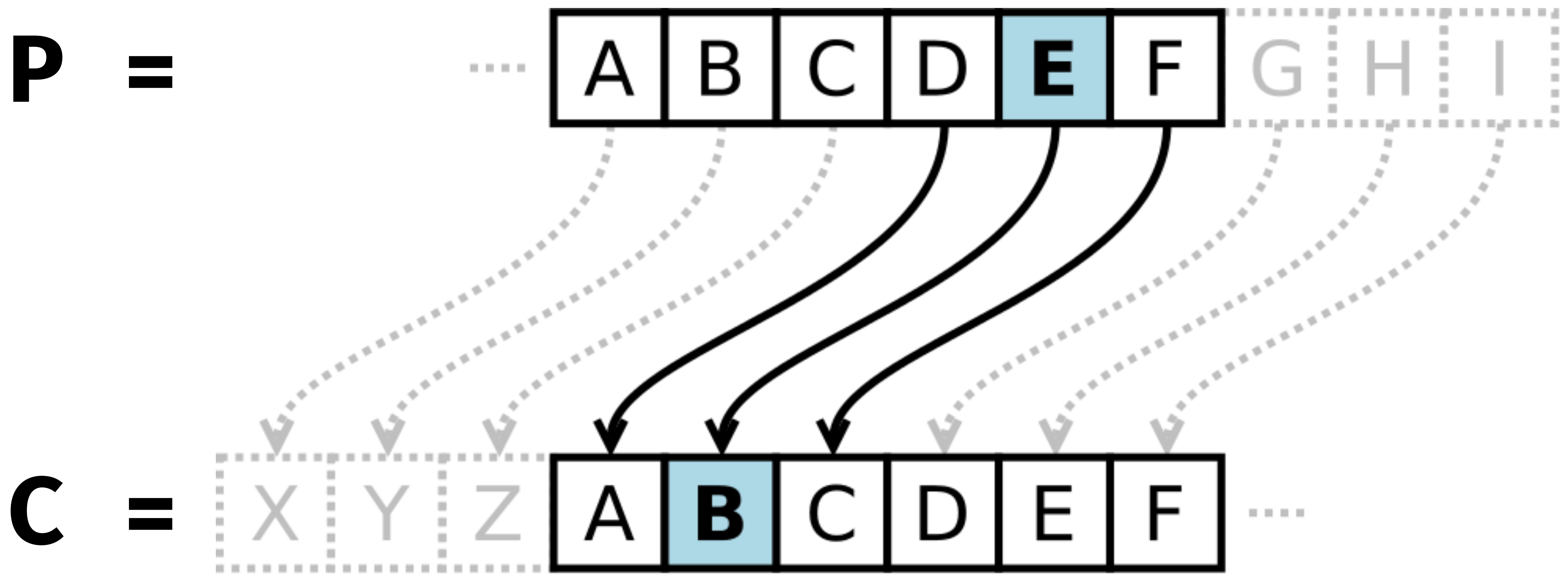


Image source: Wikipedia

Substitute each word / block

Caesar cipher

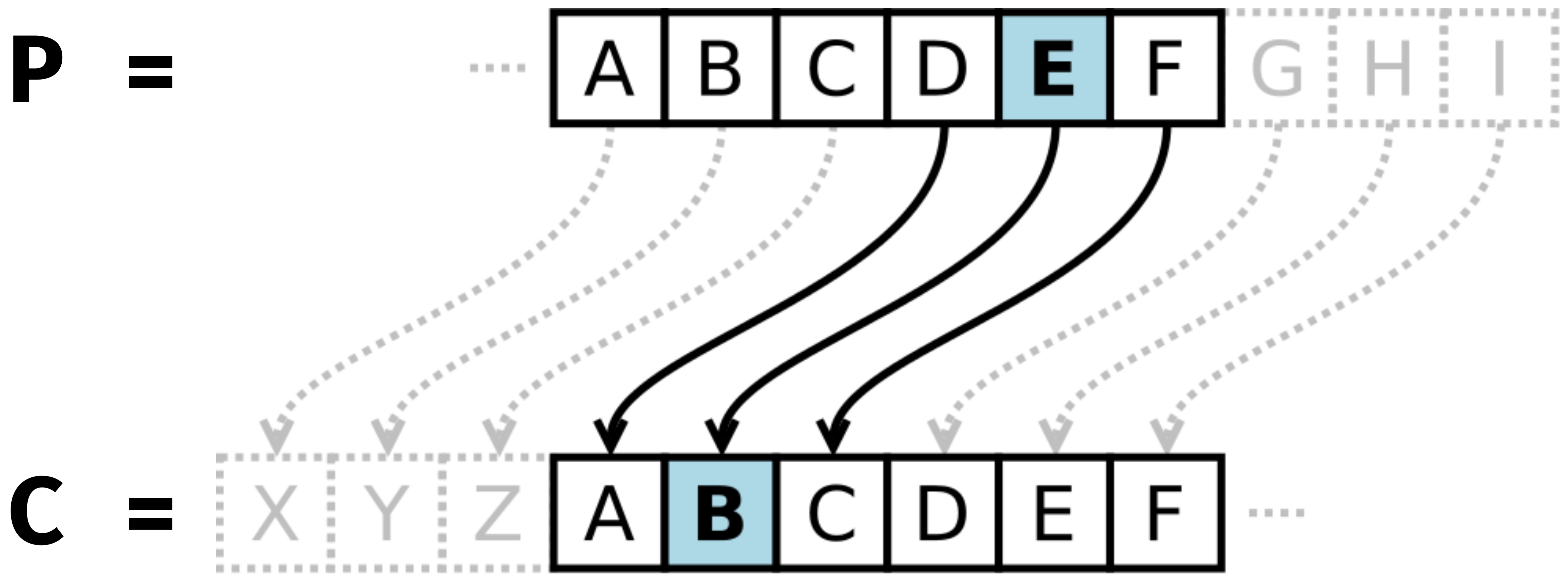


Image source: Wikipedia

- Encipher one character at a time
- Figure of cipher with key $K = 3$
  - `one` ↦ `lkb`
  - `two` ↦ `qtl`
- Problems?
  - If Eve observes C for a known P (even 1 known character), then she learns K
  - `three` ↦ `qeobb`
  - Reuse leads to a "frequency attack"

cryptograms.org

AKYHM JOO WGKYDVDGKC UHOO-GMPJKDEHY LDGOHKWH CHHFC VG NDF VNH CNGMVHCV YDCVJKWH SHVUHHK VUG BGDKVC.

Check It! Reset Hint me!

Stats

A – 1
B – 1
C – 7
D – 7
E – 1
F – 2
G – 8
H – 13
J – 3
K – 8
L – 1
M – 3
N – 3
O – 5
P – 1
S – 1
U – 3
V – 9
W – 3
Y – 4

One time pad

⊕	0	1
0	0	1
1	1	0

- Fix Caesar cipher by giving each character its own key
- XOR function measures whether 2 inputs are identical
- OTP "masks" private message by applying Caesar cipher independently to each bit
- XOR is a "lossless" function, so it is invertible (XOR is its own inverse)
- Drawbacks?
  - Key length == length of private message
  - No integrity: easy to manipulate enciphered text

```
    message  0110 0110 1001
XOR key      0011 1100 1110
    cipher   0101 1010 0111

    cipher   0101 1010 0111
XOR key      0011 1100 1110
    message  0110 0110 1001
```


Slowenisch
Rumänisch
Collins Gem
Russian
COLLINS
SPANISH
Französisch - Deutsch
Deutsch - Französisch
ROMANIAN-ENGLISH
ENGLISH-ROMANIAN
spanisch deutsch
deutsch spanisch
PONS
Praxis
Klett

Source:
www.simonandschuster.com/books/The-Klingon-Dictionary/Marc-Okrand/Star-Trek/9780671745592

A crypto "Manhattan project"

Plain word	Code word
aba	nrq
abs	mbk
ace	ybd
act	wxv
add	jen
ado	hhg
aft	uxv
age	zmx
ago	dgs
aha	ase
aid	ktf
⋮	⋮
zip	cyu
zoo	dux

3-letter words 3 characters from random.org

- Codebook = random-looking permutation $R : \{0,1\}^\mu \to \{0,1\}^\mu$
  - Why is it important that all codewords are distinct?
  - So Alice can decipher her message
- Suppose society expends an enormous effort to make one public codebook R (and inverse)
- Can Alice protect her messages by encoding $\mathrm{P} \to R(\mathrm{P})$?

“If an adversary Eve has not **explicitly queried** a [perfect codebook] ***R*** on some point *X*, then the value of ***R***(*X*) is **completely random**… at least as far as Eve is concerned.”

–*Jon Katz and Yehuda Lindell,* Introduction to Modern Cryptography

Properties of a public codebook *R*

Plain word	Code word
aba	nrq
abs	mbk
ace	ybd
act	wxv
add	jen
ado	hhg
aft	uxv
age	zmx
ago	dgs
aha	ase
aid	ktf
⋮	⋮
zip	cyu
zoo	dux

3-letter words

3 characters from random.org

- Privacy: *R* unintelligible to Eve?
  - No, Eve can use the codebook too
- Usability: *R* is simple for Alice?
  - No, *R* is too large for Alice to carry

New plan: everybody gets a codebook

Randomness ⇒ Unpredictability ⇒ Secrecy

Privacy of a personal codebook $Y^* = \Pi(X^*)$ for Alice?

X	Y
aba	nrq
abs	mbk
ace	ybd
act	wxv
add	jen
ado	hhg
aft	uxv
age	zmx
ago	dgs
aha	ase
aid	ktf
⋮	⋮
zip	cyu
zoo	dux

Question: can Eve recover Alice’s private data X^* given

1. Only Y^* (i.e., if Alice only uses Π once)
2. Y^*, plus many known (X_i, Y_i) pairs, say chosen at random
3. Above, plus many (X_i, Y_i) pairs for X_i that Eve chose
   - Hmm, let’s enforce the restriction that $X_i \neq X^*$ (for now)
4. Above, plus Eve can choose the X_i one at a time, and adapt her choices based on the Y_i responses she receives
5. Above, plus Eve can also decipher Y_i of her choice

Upshot: whether a cipher is “secure” depends on Eve’s powers, and we want ciphers that withstand a strong Eve

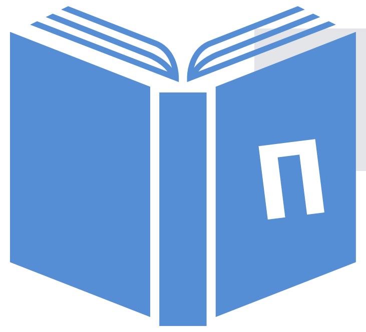
Auguste Kerckhoffs' principles to protect communication

1. The system must be practically, if not mathematically, ***indecipherable***
2. It should ***not require secrecy***, and it should not be a problem if it falls into enemy hands
3. It must be possible to communicate and ***remember the key*** without using written notes, and correspondents must be able to ***change or modify it at will***
4. It must be applicable to telegraph communications
5. It must be portable, and should not require several persons to handle or operate
6. Given the circumstances in which it is to be used, the system must be easy to use and should ***not be stressful to use*** or require its users to know and comply with a long list of rules

Source: A. Kerckhoffs, *La Militaire*, 1883

Does a private codebook *Π* satisfy Kerckhoffs' principles?

X	*Y*
aba	nrq
abs	mbk
ace	ybd
act	wxv
add	jen
ado	hhg
aft	uxv
age	zmx
ago	dgs
aha	ase
aid	ktf
⋮	⋮
zip	cyu
zoo	dux

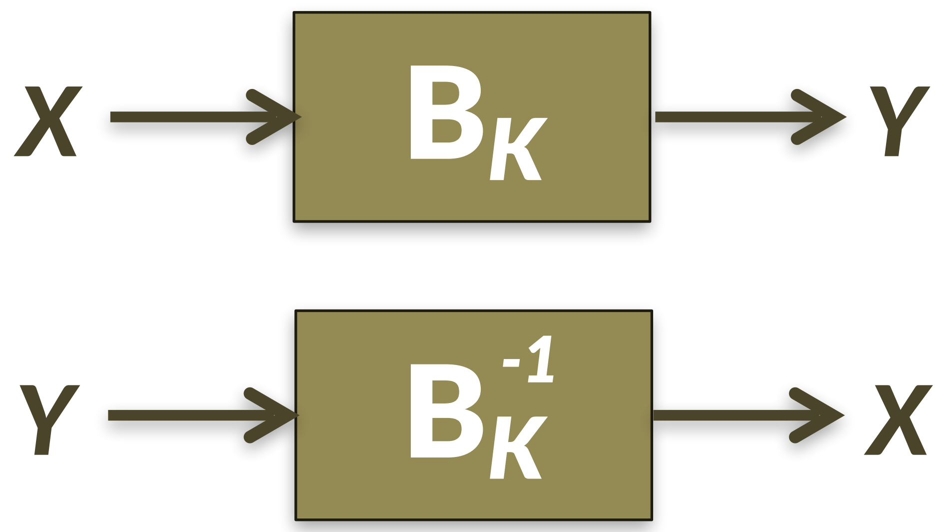


- Easy to remember *Π*? ⇒ ✘ Codebook is huge
- Easy to change *Π*? ⇒ ✘ Creating *Π* is very difficult
- Stress-free to use *Π*? ⇒ ✘ Slow to search a big table, and cannot repeat input *X*

Let's ignore frequency attacks for now.
Focus on addressing the other usability goals.

What we want: a block cipher

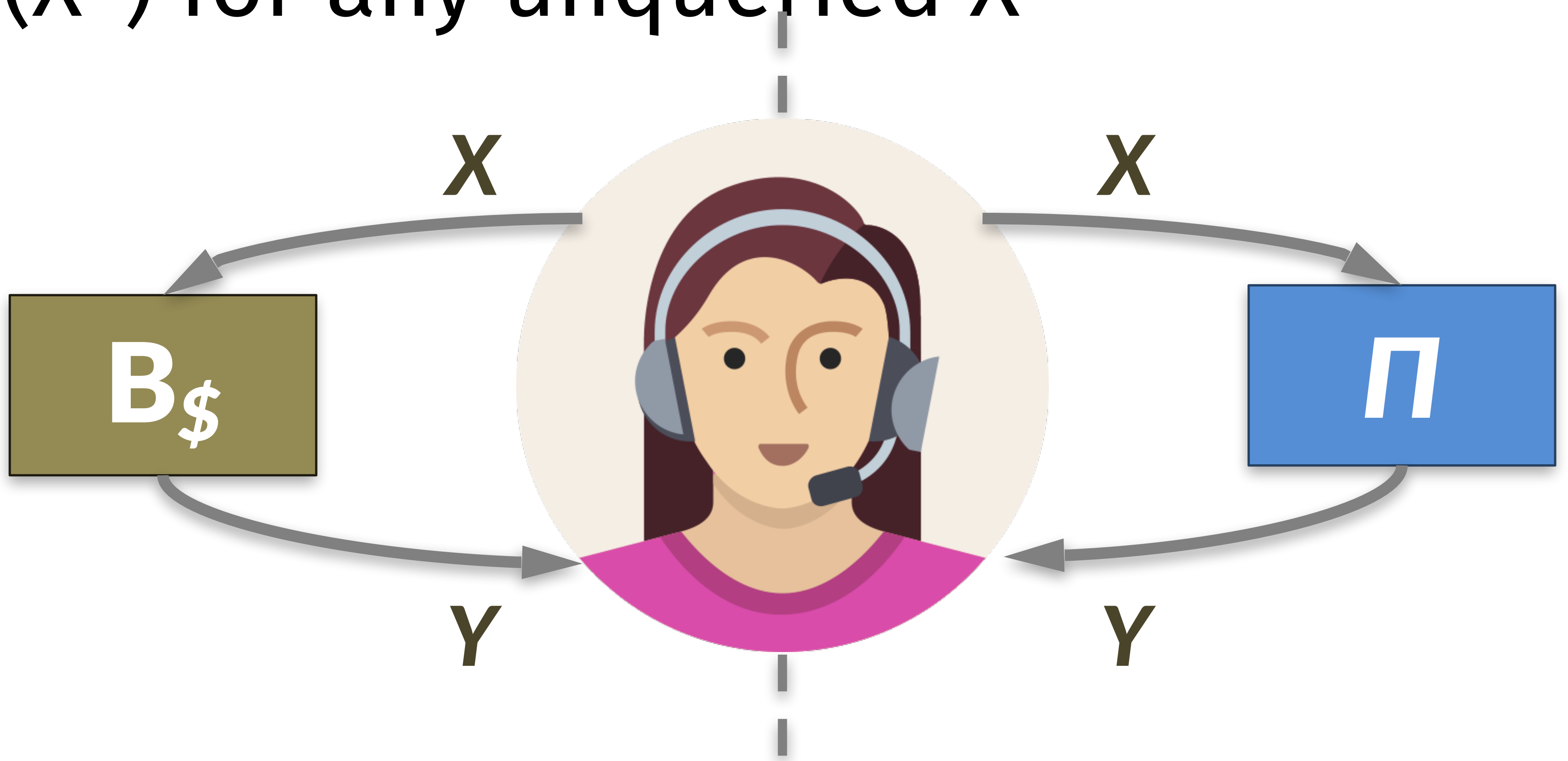
- Family of invertible permutations (i.e., codebooks), indexed by a secret key
- Forward direction called *enciphering*
- Backward direction called *deciphering*



- Design goals
  1. **Simple** - built from native CPU operations like XOR, cyclic shifts, and small table lookups so they are really fast to compute (think: throughput of 3-4 GB/sec)
  2. **Makes no sense** - its design looks unpredictable (aka pseudorandom)
  3. **Simple to see why it makes no sense** - we have simple, convincing arguments that the cipher is unpredictable (remember Schneier's law!)

Pseudorandomness

- Goal: rows of truth table are "independent"
  - Suppose Eve adaptively makes q queries to codebook B_K using a randomly chosen key K
  - We call B **pseudorandom** if Eve has a very small chance to predict B_K(X*) for any unqueried X*



- Upshot
  - Good usability: Alice gets to use the simple cipher B_K
  - Good privacy: (almost) as hard for Eve to understand as Π

X	*Y*
aba	nrq
abs	mbk
ace	ybd
act	wxv
add	jen
ado	hhg
aft	uxv
age	zmx
ago	dgs
aha	ase
aid	ktf
⋮	⋮
zip	
zoo	

Strong pseudorandomness

B_K is *strongly pseudorandom* if every resource-bounded Eve can distinguish the real cipher from Π with very small probability ε

- Here, we provide Eve with access to both enciphering and deciphering
- In this class, we will only be concerned with *strong* pseudorandomness