Lecture 06

Private-key Encryption

Definitions and security of the one-time pad

Two requirements, one goal

Correctness

Bob recovers the original message when he decrypts Alice's ciphertext using their secret key.

Privacy

The ciphertext gives an eavesdropper no additional information about Alice's message.

Main result: the one-time pad achieves both

Over any finite group, a uniformly sampled secret key, independent of the message, makes the one-time pad **correct and private** for one encryption.

Security in this course = correctness + privacy.

Lecture 06, pp. 2, 5–12 [Maj26].

How to use this supplement

Start with the lecture

The distributed 17-slide Lecture 06 deck remains the required minimum. We follow its order: definitions, construction, correctness, privacy, and examples.

What this companion adds

The short algebra and probability arguments are written out.
Worked numerical instances and review questions illustrate the lecture's four message spaces.

Roadmap. Specify the experiment; separate correctness from privacy; prove the group-based one-time pad; instantiate it for bits and letters.

Baseline: [Maj26]. Textbook pointers at the end are optional reading.

Private-key encryption: three algorithms

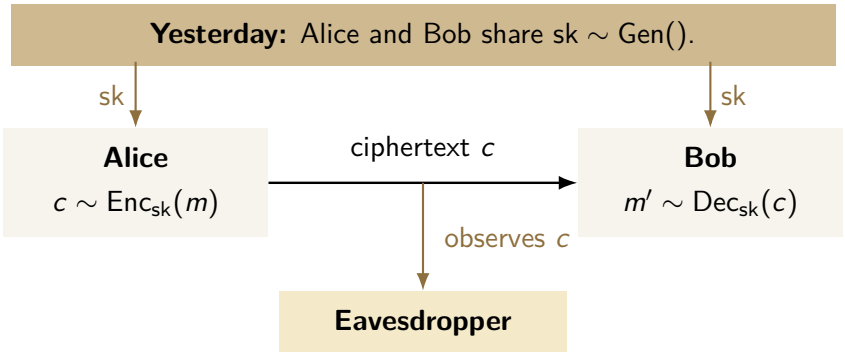
A private-key encryption scheme specifies Gen, Enc, and Dec.

Algorithm	**Input**	**Output**
Key generation $\mathsf{Gen}()$	No input in this lecture	Secret key sk
Encryption $\mathsf{Enc}_{\mathsf{sk}}(m)$	Key sk, message m	Ciphertext c
Decryption $\mathsf{Dec}_{\mathsf{sk}}(c)$	Key sk, ciphertext c	Recovered message m'

Encryption and decryption *may* be randomized. The one-time pad will be a special case in which both are deterministic.

Lecture 06, pp. 3–4, 8 [Maj26].

The story: Alice sends, Bob recovers



Today the ciphertext is public. The shared secret key is not.

Lecture 06, p. 4 [Maj26].

Values, random variables, and sampling

Lowercase and uppercase

	Value	Variable
Message	m	M
Secret key	sk	SK
Ciphertext	c	C
Recovered	m'	M'

Read the notation

$\mathsf{sk} \sim \mathsf{Gen}()$: sample according to the distribution output by Gen.
$c := \mathsf{Enc}_{\mathsf{sk}}(m)$: assign the result of a deterministic computation.
$\mathsf{sk} \xleftarrow{\$} G$: sample uniformly from the finite set G.

The message may have any distribution. The key is sampled **independently** of it; this independence is used in the privacy proof.

Lecture 06, pp. 4–5, 8, 13 [Maj26].

Correctness: recover the original message

For any fixed message m, run the experiment

$$\mathsf{SK} \sim \mathsf{Gen}(), \qquad C \sim \mathsf{Enc}_{\mathsf{SK}}(m), \qquad M' \sim \mathsf{Dec}_{\mathsf{SK}}(C).$$

Correctness requirement

The output must be m with probability 1:

$$\Pr[M' = m] = 1 \qquad \text{for every message } m.$$

Equivalently, for any message distribution in the encryption experiment,

$$\Pr[M' = M] = 1.$$

The probability includes all randomness in key generation, encryption, and decryption. The target here is certainty, not merely a high success probability.

Lecture 06, p. 5 [Maj26]; the fixed-message quantifier is made explicit.

Privacy: the ciphertext does not update the message law

Before observing the ciphertext

The **prior** probability of a message:

$$\Pr[M = m].$$

After observing $C = c$

The **posterior** probability:

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

Privacy requirement

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

Require this for every message distribution independent of the key, every m, and every c with $\Pr[C = c] > 0$.

Lecture 06, p. 6 [Maj26]. Explicit quantifiers: cf. [BS23, p. 11].

Dropping either requirement makes the task trivial

Correct, but not private

Send the message in the clear:

$$\mathrm{Enc}_{\mathsf{sk}}(m) = m, \qquad \mathrm{Dec}_{\mathsf{sk}}(c) = c.$$

Bob always recovers m.
For a fair bit message,

$$\Pr[M = 0 \mid C = 0] = 1 \neq \tfrac{1}{2}.$$

Private, but not correct

Send the same symbol every time:

$$\mathrm{Enc}_{\mathsf{sk}}(m) = 0.$$

Observing $C = 0$ changes nothing.
But a decoder given $(\mathsf{sk}, 0)$ cannot always recover two distinct messages.

The nontrivial goal is to achieve **correctness and privacy simultaneously**.

Lecture 06, p. 7 [Maj26]; the fair-bit calculation is a worked illustration.

Recall: the group facts we will use

Let $(G, \circ)$ be a finite group. Write e for its identity and $\mathrm{inv}(x)$ for the inverse of x.

Group property	**Identity available for the proof**
Associativity	$(x \circ y) \circ z = x \circ (y \circ z)$
Identity element	$x \circ e = e \circ x = x$
Inverse element	$x \circ \mathrm{inv}(x) = \mathrm{inv}(x) \circ x = e$

Keep the order of the factors. These laws do not require $x \circ y = y \circ x$. We will not assume commutativity.

Group notation recalled for Lecture 06, pp. 8, 13 [Maj26].

One-time pad on a finite group

Messages, secret keys, and ciphertexts all belong to G.

Key generation	**Encryption**	**Decryption**
$\mathsf{Gen}()$:	$\mathsf{Enc}_{\mathsf{sk}}(m)$:	$\mathsf{Dec}_{\mathsf{sk}}(c)$:
$\mathsf{sk} \xleftarrow{\$} G.$	$c := m \circ \mathsf{sk}.$	$m' := c \circ \mathsf{inv}(\mathsf{sk}).$
Return sk.	Return c.	Return m'.

Only key generation is randomized. Sample the key independently of the message. Once sk is fixed, encryption and decryption are deterministic.

Finite G is explicit here because the lecture samples uniformly from G.

Lecture 06, pp. 8, 11, 13 [Maj26].

Correctness: cancel the key on the right

Proof idea. Use associativity, then the inverse and identity laws.

For every $m, \mathsf{sk} \in G$,

$$\begin{aligned} \mathsf{Dec}_{\mathsf{sk}}(\mathsf{Enc}_{\mathsf{sk}}(m)) &= (m \circ \mathsf{sk}) \circ \mathsf{inv}(\mathsf{sk}) & \\ &= m \circ (\mathsf{sk} \circ \mathsf{inv}(\mathsf{sk})) & \text{associativity} \\ &= m \circ e & \text{inverse} \\ &= m. & \text{identity} \end{aligned}$$

Conclusion

Decryption always returns the original message: $\Pr[M' = M] = 1$.

Independent check. This is an identity for every key, not an average-case statement. Neither uniformity of the key nor commutativity was used.

Expanded proof of Lecture 06, p. 9 [Maj26].

Privacy: the proof roadmap

Our target is the lecture's posterior identity:

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

1. Rewrite

Use conditional probability and sum over all possible messages.

2. Use the claim

Replace each joint probability by the message probability times $1/|G|$.

3. Prove the claim

A message and a ciphertext determine exactly one compatible key.

We retain the lecture's order: first deduce privacy from **Claim 1**, then prove Claim 1. The claim is stated explicitly before it is used.

Lecture 06, pp. 10–13 [Maj26].

Privacy I: rewrite the posterior

Fix $m, c \in G$, with $\Pr[C = c] > 0$.

$$\Pr[M = m \mid C = c] = \frac{\Pr[M = m, C = c]}{\Pr[C = c]}$$

$$= \frac{\Pr[M = m, C = c]}{\sum_{x \in G} \Pr[M = x, C = c]}.$$

Why the denominator is a sum

The events $\{M = x, C = c\}$, as x ranges over G, are disjoint and together make up the event $\{C = c\}$.

Next: the joint-probability claim will simplify both numerator and denominator.

Privacy II: isolate the counting claim

Claim 1 (proved shortly)

For every $x, y \in G$,

$$\Pr[M = x, C = y] = \Pr[M = x] \cdot \frac{1}{|G|}.$$

Substituting Claim 1 into the previous expression gives

$$\Pr[M = m \mid C = c] = \frac{\Pr[M = m] \cdot |G|^{-1}}{\sum_{x \in G} \Pr[M = x] \cdot |G|^{-1}}.$$

The same nonzero factor $|G|^{-1}$ appears in numerator and denominator. Cancel it.

Lecture 06, p. 11 [Maj26].

Privacy III: finish the calculation

Since $\sum_{x \in G} \Pr[M = x] = 1$, canceling the common factor from the previous slide gives

$$\Pr[M = m \mid C = c] = \frac{\Pr[M = m]}{\sum_{x \in G} \Pr[M = x]} = \frac{\Pr[M = m]}{1} = \Pr[M = m] .$$

What remains?

Correctness is already proved. Once Claim 1 is established, privacy follows from this calculation, and the one-time pad is secure in the course's sense.

No uniformity assumption was made about M. The distribution of the message may be highly biased.

Lecture 06, p. 12 [Maj26].

Exactly one key connects a message to a ciphertext

Proof idea. Solve $x \circ z = y$ by multiplying on the *left* by $\mathrm{inv}(x)$.

The unique compatible key

For every $x, y \in G$, the equation $x \circ z = y$ has exactly one solution:

$$z = \mathrm{inv}(x) \circ y.$$

Existence

$$\begin{aligned} x \circ (\mathrm{inv}(x) \circ y) &= (x \circ \mathrm{inv}(x)) \circ y \\ &= e \circ y = y. \end{aligned}$$

Uniqueness

If $x \circ z = y$, then

$$\begin{aligned} \mathrm{inv}(x) \circ y &= \mathrm{inv}(x) \circ (x \circ z) \\ &= e \circ z = z. \end{aligned}$$

Worked proof of the homework fact on p. 13 [Maj26]; existence is also explicit.

Proof of Claim 1: from events to key probabilities

Fix $x, y \in G$, and set $z := \mathsf{inv}(x) \circ y$.

$$\begin{aligned} \Pr[M = x, C = y] &= \Pr[M = x, \mathsf{SK} = z] && \text{unique compatible key} \\ &= \Pr[M = x]\ \Pr[\mathsf{SK} = z] && \text{independence} \\ &= \Pr[M = x] \cdot \frac{1}{|G|}. && \text{uniform key} \end{aligned}$$

The proof is complete

Claim 1 gives privacy. Together with correctness, this proves security for one encryption.

This argument still works when $\Pr[M = x] = 0$: no division by a message probability is used.

Lecture 06, p. 13 [Maj26].

Independent check: compute the ciphertext distribution

For a possible message m, the map $\mathsf{sk} \mapsto m \circ \mathsf{sk}$ is a bijection. Independence keeps the key uniform after conditioning on $M = m$.

$$\Pr[C = c \mid M = m] = \frac{1}{|G|} \qquad \text{when } \Pr[M = m] > 0.$$

Average over the message distribution; every ciphertext has positive probability:

$$\Pr[C = c] = \sum_{m:\, \Pr[M=m]>0} \Pr[M = m] \, \frac{1}{|G|} = \frac{1}{|G|}.$$

Return to the posterior

$$\Pr[M = m \mid C = c] = \frac{\Pr[M = m] \,/ |G|}{1/|G|} = \Pr[M = m] \,.$$

A second derivation of the calculation on pp. 10–13 [Maj26].

Example I: encrypting one bit

Use $(G, \circ) = (\mathbb{Z}_2, + \bmod 2)$. Write $\oplus$ for addition modulo 2.

The algorithms

Choose an independent, uniform key $\mathsf{sk} \in \{0, 1\}$.

$$c = m \oplus \mathsf{sk}, \qquad m' = c \oplus \mathsf{sk}.$$

Each bit is its own inverse: $\mathsf{sk} \oplus \mathsf{sk} = 0$.

m	sk	$c = m \oplus \mathsf{sk}$
0	0	0
0	1	1
1	0	1
1	1	0

For either message, a uniform key gives ciphertext 0 or 1 with probability $1/2$ each. The ciphertext distribution does not depend on the message.

Lecture 06, p. 14 [Maj26]; the truth table makes the example explicit.

Worked illustration: a biased message stays biased

Let $\Pr[M = 0] = 3/4$ and $\Pr[M = 1] = 1/4$. The independent key is a fair bit.

Joint probabilities

	$C = 0$	$C = 1$	Prior
$M = 0$	$3/8$	$3/8$	$3/4$
$M = 1$	$1/8$	$1/8$	$1/4$
Total	$1/2$	$1/2$	1

After observing $C = 1$

$$\Pr[M = 0 \mid C = 1] = \frac{3/8}{1/2} = \frac{3}{4}.$$

The message remains just as biased as it was before the ciphertext was observed.

Check the other column: observing $C = 0$ also leaves the probability of $M = 0$ equal to $3/4$.

Companion illustration of Claim 1 and Example I [Maj26, pp. 11, 14].

Example II: encrypting an n-bit string

Use $G = \{0,1\}^n$ with coordinate-wise addition modulo 2:

$$(x_1, \ldots, x_n) \oplus (y_1, \ldots, y_n) = (x_1 + y_1 \bmod 2, \ldots, x_n + y_n \bmod 2).$$

One uniform n-bit key

$$\mathsf{sk} \xleftarrow{\$} \{0,1\}^n, \qquad c = m \oplus \mathsf{sk}.$$

Decrypt with the same bitwise operation:

$$m' = c \oplus \mathsf{sk}.$$

Worked example: $n = 6$

m	101101
sk	011010
$c = m \oplus \mathsf{sk}$	110111
$c \oplus \mathsf{sk}$	101101

Lecture 06, p. 15 [Maj26]; the six-bit calculation is a worked instance.

Example III: encrypting an alphabet symbol

Represent the 26 letters by residues modulo 26:

$$\texttt{A} \leftrightarrow 0, \quad \texttt{B} \leftrightarrow 1, \quad \ldots, \quad \texttt{Z} \leftrightarrow 25.$$

Addition modulo 26

$\mathsf{sk} \xleftarrow{\$} \mathbb{Z}_{26}, \qquad c = m + \mathsf{sk} \pmod{26}.$

Decrypt by subtraction:

$$m' = c - \mathsf{sk} \pmod{26}.$$

Worked example

Message T: $m = 19$. Choose $\mathsf{sk} = 11$.

$$c = 19 + 11 \equiv 4 \pmod{26}.$$

So the ciphertext symbol is E.

$$4 - 11 \equiv 19 \pmod{26}.$$

Worked instance of Lecture 06, p. 16 [Maj26]. Polynomial homework preview: final slide.

Example IV: encrypting an n-symbol word

Use $G = \mathbb{Z}_{26}^n$, with coordinate-wise addition modulo 26:

$$c_i = m_i + \mathrm{sk}_i \pmod{26}, \qquad m_i' = c_i - \mathrm{sk}_i \pmod{26}.$$

Worked example: three letters

	First	Second	Third	Word
Message	2	0	19	`CAT`
Secret key	5	14	22	
Ciphertext	7	14	15	`HOP`
After subtraction	2	0	19	`CAT`

Sample the key independently and uniformly from all 26^n vectors, not as one common shift applied to every coordinate.

Lecture 06, p. 17 [Maj26]; the three-letter calculation is a worked instance.

One construction, four message spaces

Messages	**Group G**	**Operation**	$\lvert G\rvert$
One bit	$\mathbb{Z}_2$	Addition modulo 2	2
n-bit strings	$\{0,1\}^n$	Bitwise XOR	2^n
One letter	$\mathbb{Z}_{26}$	Addition modulo 26	26
n-letter words	$\mathbb{Z}_{26}^n$	Coordinate-wise addition modulo 26	26^n

The same theorem handles all four

Use a uniform, independent key from the whole group. Decryption removes that key; each message–ciphertext pair has exactly one compatible key.

Lecture 06, pp. 8–17 [Maj26].

Check your understanding

1. Recover the compatible key

In the one-time pad over $\mathbb{Z}_{26}$, a message $m = 7$ gives ciphertext $c = 3$. What is the key? Check both encryption and decryption.

2. Keep the message distribution arbitrary

For the one-bit pad with a uniform, independent key, suppose $\Pr[M = 0] = 4/5$. What is $\Pr[M = 0 \mid C = 1]$?

3. Locate the assumptions

Where do uniformity and independence of the key enter the privacy proof? Does the correctness proof use either one? Does either proof require commutativity?

Companion practice on Lecture 06, pp. 9–14, 16 [Maj26]; not assigned homework.

Answers: the same two proof moves

1. Solve for the key

The key is $\mathsf{sk} = 3 - 7 \equiv 22 \pmod{26}$.
Check: $7 + 22 \equiv 3$ and $3 - 22 \equiv 7 \pmod{26}$.

2. The posterior equals the prior

$$\Pr[M = 0 \mid C = 1] = \frac{(4/5)(1/2)}{1/2} = \frac{4}{5}.$$

3. Algebra versus probability

Independence factors the joint event $\{M = x, \mathsf{SK} = z\}$; uniformity makes $\Pr[\mathsf{SK} = z] = 1/|G|$. Correctness needs neither. Neither proof requires commutativity.

Solutions to the preceding companion practice.

Takeaways

Correctness is an algebraic identity

$$(m \circ \mathsf{sk}) \circ \mathsf{inv}(\mathsf{sk}) = m.$$

Bob knows the key and removes it on the right.

Privacy is a probability identity

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

Every candidate message has exactly one compatible key, sampled with the same probability.

The theorem. On any finite group, the one-time pad with a uniform key independent of the message is correct and private for one encryption.
The message distribution need not be uniform. The group need not be commutative.

Lecture 06, pp. 5–13 [Maj26].

Reading guide

Required baseline: the distributed Lecture 06

Pages 3–7: syntax, correctness, privacy, and the two trivial schemes.

Pages 8–13: the group-based pad and its proof.

Pages 14–17: the four message-space examples.

Optional comparison: Boneh–Shoup, Version 0.6

Basic ciphers and examples: §2.1.1, pp. 4–7; Example 2.1, p. 6; Example 2.4, p. 7.

The bit-string pad: Theorem 2.2, p. 9.

Posterior probabilities and independence: §2.1.2, p. 11; Theorem 2.4 and its proof, pp. 11–12.

The book uses different notation. Textbook references use printed page numbers.

[Maj26, BS23]. The proofs here do not assume an external theorem.

References and the homework preview

[BS23] Dan Boneh and Victor Shoup.
A Graduate Course in Applied Cryptography.
Author-maintained textbook, 2023.
Version 0.6, January 2023. Sections 2.1.1–2.1.2.
URL: `https://toc.cryptobook.us/book.pdf`.

[Maj26] **Hemanta K. Maji.**
Lecture 06: Private-key encryption (Definition & Security of one-time pad).
CS 35500, Purdue University. Instructor's lecture slides, 2026.
Revised lecture deck, 17 slides; uploaded as `06(1).pdf`.

Homework preview: a multiplication-type construction

Page 16 previews polynomials of degree less than 3 with coefficients in $\mathbb{Z}_3$:

$$a_0 + a_1 X + a_2 X^2, \qquad a_0, a_1, a_2 \in \mathbb{Z}_3.$$

There are $3^3 = 27$ such polynomials; the homework will define the operation.