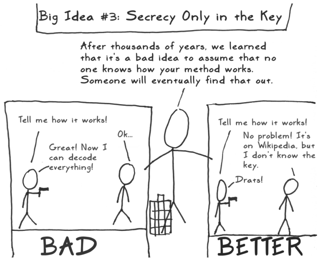
CPSC 418/MATH 318 Introduction to Cryptography

Course Technicalities, Symmetric Cryptosystems

Renate Scheidler

Department of Mathematics & Statistics
Department of Computer Science
University of Calgary

Week 1


Big Idea #3: Secrecy Only in the Key
After thousands of years, we learned that it's a bad idea to assume that no one knows how your method works. Someone will eventually find that out.
Tell me how it works!
Ok...
Great! Now I can decode everything!
BAD
Tell me how it works!
No problem! It's on Wikipedia, but I don't know the key.
Drats!
BETTER

Outline

1. A Real-World Problem
2. Course Technicalities
3. Overview of Cryptography
   - Cryptography Within Information Security
4. Symmetric Cryptography
5. Cryptanalysis
   - Cryptographic Attacks

Motivation

Cryptography (from the Greek) — 'hidden writing'

What would you like to see in a secure electronic assignment submission system? Want submission:

- confidential so no one can steal it (confidentiality)
- protected so no one can alter it (data integrity)
- authentic so no one can impersonate creator (entity authentication)
- safe from intrusion on disk (access control)
- safe from denial by instructor or TA (non-repudiation)

What we will do: This course will work toward solutions for ensuring all of these. Examples of complete systems at the end of the course.

What we won't do: Hacking, bitcoin, cipher puzzles, invent cryptosystems, . . .

Course Details

Course web page: `//cpsc.ucalgary.ca/~rscheidl/crypto`

- Course info, slides, handouts, course schedule, other resources
- Link can be found on the D2L page (combined page for CPSC 418 and MATH 318, but NOT used)

Delivery:

- Lectures and tutorials are delivered *in person*
- Office Hours: MWF right after class or by appointment
- Tutorials (for CPSC 418, optionally for MATH 318):

  Each tutorial has an individual time slot; the Wed 18:00 time is common to all 5 tutorials

  Consult the course web page for schedule and content

  MATH 318 students are welcome to attend subject to space

To-do for this week: read through the "home" and "about" tabs on the course website.

Course Materials

- Recommended textbook (entirely optional):

  D. R. Stinson & M. B. Paterson
  Cryptography — Theory and Practice
  4th edition, CRC Press, 2019

  Older editions of Stinson's book are obsolete and missing modern material!

- Slides, handouts, practice problems, LaTeX templates, tutorial materials (on course web page)
- Other sources (see "references" page)

Software Tools

Course resources: course web page

Discussion forum: *Piazza*, `//piazza.com`

- You should all be enrolled (let me know if you aren't)
- Activate your account asap
- Sign up at
  `//piazza.com/ucalgary.ca/winter2024/cpsc418math318`
- Use your U of C e-mail address and surname as they appear on D2L

Assignment submission: *Gradescope*, `//gradescope.ca`

- Create an account on the *Canadian site*
- Your user name is your U of C e-mail address
- Add the course to your account using the code **95V468**

For day-to-day use, you only need the course web page and Piazza.

Homework

3 assignments, each worth 10%, each consisting of

- Written problems common to CPSC 418 and MATH 318
- Programming problems for CPSC 418 only
  (can be done by MATH 318 students for limited bonus credit)
- Written problems for MATH 318 only
  (can be done by CPSC 418 students for limited limited bonus credit)

An additional set of *challenge problems* for extra credit

Extra credit policy is explained on the "assignments" tab of the course page.

Homework (cont'd)

- Assignments and challenge problems posted on the Piazza "Resources" tab
- Tentative assignment due dates: Feb. 2, Mar. 1, Apr. 8;
  Challenge problems also due Apr. 8
- All work must be done *individually*
- All written work must be done in LaTeX
- All CPSC 418 programming problems must be done in Python.

To-do for next week: *thoroughly and carefully* read the "assignments" tab on our course web page

To-do for the week after next: practice LaTeX

Exams

Midterm Exam, worth 30%: Friday Mar. 15, 18:00-19:30 pm, in person

Final Exam, worth 40%: during final exam period, in person

Exam details:

- Closed book
- Mix of multiple choice, short answer and long answer
- All work must be done *individually*
- Study guides and practice exams will be posted in advance

Course Content

Rough schedule (see "schedule" tab on course webpage):

- 5 weeks: Encryption via conventional cryptography (what it is, what it does, techniques, attacks)
- 1 week: Cryptographic key agreement
- 1.5 weeks Data integrity via conventional cryptography
- 4 weeks: Public Key Cryptography (encryption, signatures)
- 1 week: Cryptography in practice and real-life use examples, plus other topics (time-permitting)

More about this course under "about" tab on the course web page

CPSC 418 is part of the Computer Science BSc concentration (area of specialization) in Information Security

Basic Terminology

Historically, cryptography is the art of sending messages in secret, or disguised form.

Definition 1 (encrypt, encipher)

To render a message unintelligible to everyone except the intended recipient.

Definition 2 (decrypt, decipher)

To transform an encrypted message back into its unencrypted form.

More Terminology

Definition 3 (plaintext)

The message or data to be encrypted.

Definition 4 (ciphertext)

The message after encryption.

Definition 5 (cipher, cryptosystem)

A particular method of encryption, capable of handling arbitrary messages.

An Old Example

Example 6 (Caesar Cipher)

Substitute each plaintext letter with the third subsequent letter of the alphabet, wrapping from Z to A; *i.e.* $A \to D$, $B \to E$, $\cdots$, $Z \to C$:

Plaintext: `I came, I saw, I conquered.`

Ciphertext: `L FDPH, L VDZ, L FRQTXHUHG.`

Example of a class of ciphers knows as *shift ciphers*:

- shift every letter by another letter a fixed position down in the alphabet (with "wrap-around" at "Z").

2000 years old: According to Suetonius ("Lives of the Caesars"), Julius Caesar used this cipher during his campaign in Gaul (modern day France) to send encrypted dispatches back to Rome.

An Old Example

Example 7 (Shift Cipher)

Can you crack the code?

Ciphertext: `GTB YNJX FWJ HTTQ.`

Plaintext: `Bow ties are cool.`

Any good strategies?

Who Uses Cryptography?

Historic users:

- Governments (military, diplomatic service)
- A few private citizens (illicitly, e.g. for secret love letters, conspiracies)

Modern users (since invention of computers):

- Everyone (using a computer, smart phone, credit card, ATM, the internet, . . .)

Cryptography is ubiquitous! Examples:

- E-commerce, online banking/shopping/auctioning, storage of sensitive data, cloud computing, and much more
- Personal computers, mobile phones, online communication and conferencing, chip cards, medical devices, cars, sensors, and more.

Modern cryptography does MUCH more than just hiding messages!

Information Security

How is cryptography related to information security?

Definition 8 (information security)

Measures to protect information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction.

Cryptography provides *some* such measures

- important foundational part of complete security systems
- addresses mainly technological questions
- does *not* do it all!

Security Objectives

Cryptography provides services that can achieve *security objectives*.

Services provided by modern cryptography:

- Data confidentiality (data only readable to legitimate parties)
- Data integrity (data has not been modified)
- Non-repudiation (protection against denial by one of the parties in a communication)
- Authentication (communicating entity is the one claimed)
- Access Control

Security Mechanisms

Encryption is just one of many *security mechanisms* that achieve one or more of the above security objective.

Cryptographic security mechanisms discussed in this course include:

- Encryption systems — for confidentiality and limited data integrity
- Hash functions, Message Authentication Codes (MACs) — for data integrity
- Digital signatures — for data origin authentication and non-repudiation
- Authentication exchange/protocol — for entity authentication and access control

Cryptography provides many security mechanisms, but not all

- Necessary, but not sufficient for information security
- See Anderson "Why cryptosystems fail" (1993, but still relevant today; link under "references").

Security Attacks

Security mechanisms are designed to detect, prevent, or recover from a *security attack*, *i.e.* an action that compromises the security of information owned by an organization.

In cryptography, we distinguish between

- *passive* attacks – listening, eavesdropping on information without interaction with the system
- *active* attacks – interacting with the system, modifying information (for impersonation, replaying messages, changing contents, or denial of service)

Successful cryptographic protocols typically combine several mechanisms to guard against as many different attacks as possible (especially active ones).

Modern Terminology

Definition 9

Cryptography – the study of mathematical techniques for providing information security services

Cryptanalysis – the study of mathematical techniques for attempting to defeat cryptographic security mechanisms

Cryptology – combined fields of cryptography and cryptanalysis

Cryptographic primitive – tool that represents a cryptographic security mechanism

Cryptographic protocol – an algorithm (sequence of steps) to be undertaken by two or more entities to achieve a specific security objective

Will cover primitives/protocols for all security mechanisms listed above.

Great reference: *Handbook of Applied Cryptography* (see "references")

Terminology for Ciphers

Definition 10

Message space $\mathcal{M}$ – set of all possible plaintext messages

Ciphertext space $\mathcal{C}$ – set of all possible encrypted messages

Key space $\mathcal{K}$ – the finite set of possible keys

Encryption transformation – a left invertible map $E_K : \mathcal{M} \to \mathcal{C}$, indexed by some key $K \in \mathcal{K}$

Decryption transformations – the left inverse map D_K of E_K, so $D_K(E_K(M)) = M$ for all plaintexts $M \in \mathcal{M}$.

Note: $D_K(E_K(M)) = M$ implies that $D_K \circ E_K = I$ is the *identity transformation* on $\mathcal{M}$.

Note: The fact that E_K is left-invertible is equivalent to E_K is an *injective* (*i.e.* one-to-one) map.

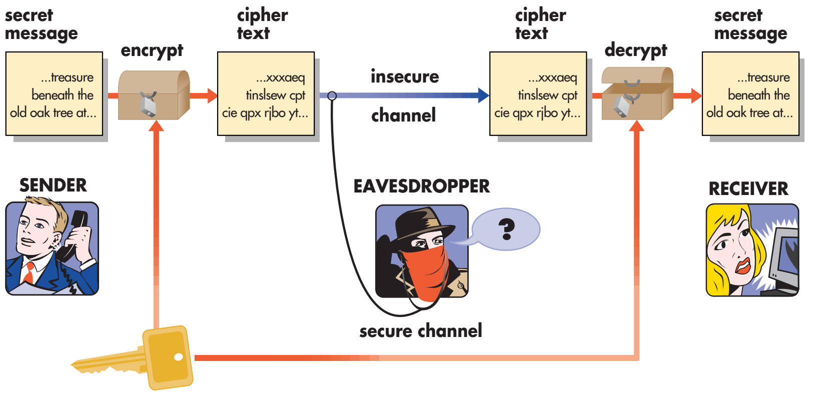
The Idea of Encryption and Decryption

In cryptography, two communicating parties are usually called *Alice* and *Bob* and adversaries are called *Eve* (short for "eavesdropper").

Idea:

- A transmitter (Bob) generates a plaintext $M \in \mathcal{M}$, to be communicated to a legitimate receiver (Alice) over an insecure channel.
- To prevent an eavesdropper (Eve) from learning the contents of M, Bob chooses a key $K \in \mathcal{K}$ and encrypts M with E_K to produce the ciphertext $C = E_K(M)$.
- C is sent along the insecure channel. When Alice obtains C, she deciphers it by applying D_K to C to obtain $M = D_K(C)$.

Conventional Cryptosystem


secret message
...treasure beneath the old oak tree at...
encrypt
cipher text
...xxxaeq tinslsew cpt cie qpx rjbo yt...
insecure channel
cipher text
...xxxaeq tinslsew cpt cie qpx rjbo yt...
decrypt
secret message
...treasure beneath the old oak tree at...
SENDER
EAVESDROPPER
?
RECEIVER
secure channel

Remarks

Encryption functions are our first example of a cryptographic primitive

- could easily formalize the above description to create a cryptographic protocol.

Note that Bob must somehow communicate the secret key to Alice without Eve obtaining it, *i.e.* over a secure channel (more on that later).

The assumption is that the workings of E_K and D_K are not secret, but K is secret. So only Alice can decrypt, but no one else can.

Example: Shift Cipher

Description:

- $\mathcal{M} = \mathcal{C} = \{A, B, \ldots, Z\}$.
- Keys represent shifts by a position between 0 and 25.
- Encryption is a forward circular shift of a plaintext letter by K
- Decryption is the corresponding backward circular shift of a ciphertext letter by K.

Example, cont.

More formally, first assign each letter a numerical equivalent as follows.

0	1	2	3	...	25
A	B	C	D	...	Z

With that, we have $\mathcal{M} = \mathcal{C} = \mathcal{K} = \mathbb{Z}_{26}$ (the integers modulo 26).

Encryption: $E_K(M) \equiv M + K \pmod{26}$ (remainder between 0 and 25).

Decryption: $D_K(C) \equiv C - K \pmod{26}$ (remainder between 0 and 25).

For the Caesar cipher, $K = 3$.

Problems with the Shift Cipher

Reveals redundancies and patterns such as repeated letters, frequent letters and letter combinations (see frequency handout on “handouts” page)

- Vulnerable to statistical attacks and frequency analysis

Very small key space: $|\mathcal{K}| = 26$

- Easily falls to a *brute force* key search attack that simply tries each key in turn

How small is “small?”

- With modern technology, $2^{128} \approx 10^{38}$ (one hundred trillion trillion trillion) is safe
- $2^{80} \approx 10^{24}$ (one trillion trillion) is questionable
- The number of keys in the first modern commercial cipher (the *Data Encryption Standard*, invented in the 1970s by IBM) is $2^{56} \approx 10^{17}$

Symmetric Cryptosystems

We are now in a position to formally define a cryptosystem:

Definition 11 (Symmetric Cryptosystem)

A symmetric cryptosytem consists of the following:

- A finite non-empty set $\mathcal{M}$ called the *plaintext (message) space*,
- A finite non-empty set $\mathcal{C}$ called the *ciphertext space*,
- A finite non-empty set $\mathcal{K}$ called the *key space*
- A single-parameter family $\{E_K\}_{K \in \mathcal{K}}$ of injective transformations

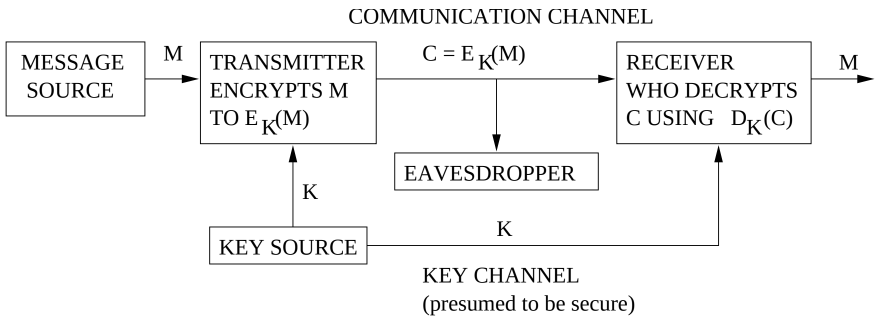
$$E_K : \mathcal{M} \to \mathcal{C} \quad \text{via} \quad M \mapsto C := E_K(M)$$

called *encryption functions*. The left inverse of E_K, denoted D_K, is called the corresponding *decryption function*. That is;

$$D_K(E_K(M)) = M \quad \text{for all } M \in \mathcal{M} \text{ and } K \in \mathcal{K} \,.$$

Schematic of a Symmetric Cryptosystem

AKA *conventional* or *private key* cryptosystems.



COMMUNICATION CHANNEL

MESSAGE SOURCE

M

TRANSMITTER ENCRYPTS M TO $E_K(M)$

$C = E_K(M)$

RECEIVER WHO DECRYPTS C USING $D_K(C)$

M

EAVESDROPPER

K

KEY SOURCE

K

KEY CHANNEL
(presumed to be secure)

Key Channel

In order for the encryption to be secure, *key channels* must be absolutely secure, as must the channel from the source to the transmitter.

In the real world, this usually means expensive.

For example, the keys to the Moscow-Washington hotline are transmitted by means of highly paid couriers, who fly there and back every week.

But If We Already Have a Secure Channel...?

It would be nice to dispense with the key channel. Why bother encrypting when we have a secure channel already?

- *Time-shifting, convenience* – you have access to a secure channel now, but would like to use it later, when the channel may not be available.
- *Speed, bandwidth* – the secure channel may be slow or of a limited bit rate.
- *Cost* – the secure channel may be expensive; *e.g.* hand-delivered by courier.
- *Feasibility* – the secure channel may be impractical; *e.g.* Alice and Bob meet in person before securely communicating.

Goals of an Attacker

We can now refine our notions of attacks on cryptosystems

Goals of an attacker:

- Deduce the key or portions thereof
- Deduce one or more plaintexts or portions thereof
- Modify a message
- Replay a message
- Impersonate (*i.e.* masquerade as) another entity

The first two are passive attacks, the last three active attacks.

Passive Attacks on Cryptosystems

Depends on what adversary has available and what they can do.

- *Ciphertext Only Attack* (COA) – adversary has only ciphertext, but no plaintext.
- *Known Plaintext Attack* (KPA) – adversary has some plaintext and corresponding ciphertext.

Note: These two attacks are *passive*: adversary does not interact with the system and has no control over the text she is given.

Active Attacks on Cryptosystems

- *Chosen Plaintext Attack* (CPA) – adversary chooses some plaintext (independently of the ciphertext she wishes to decrypt) and obtains the corresponding ciphertext.
- *Adaptive* CPA – adversary's choice of plaintext may depend on on the ciphertext she wishes to decrypt and on ciphertexts received from previous requests.
- *Chosen Ciphertext Attack* (CCA1) – adversary chooses some ciphertext (independently of the ciphertext she wishes to decrypt) and obtains the corresponding plaintext.
- *Adaptive* Chosen Ciphertext Attack (CCA2) – adversary's choice of ciphertext may depend on the ciphertext she wishes to decrypt and on plaintexts received from previous requests. She is not allowed to chose the ciphertext she wishes to decrypt.

CCA may refer to CCA1 or CCA2.

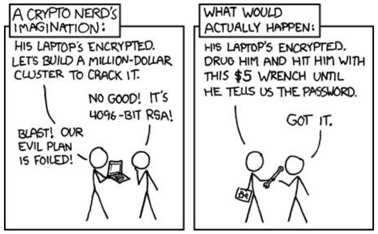
Note: These attacks are *active*: adversary interacts with the system.

More on Attacks

Note: A good/secure cryptosystem should be be secure against adaptive CCA's (as strong as possible)

Some attacks that basic cryptography cannot protect against:

- *Side Channel Attacks* – adversary exploits some aspect of the cryptosystem's implementation to extract the key (power/timing/radiation analysis)
- *Clandestine Attacks* – adversary bribes, blackmails, threatens, steals, or beats the key out of the recipient


A CRYPTO NERD'S IMAGINATION:
HIS LAPTOP'S ENCRYPTED. LETS BUILD A MILLION-DOLLAR CLUSTER TO CRACK IT.
NO GOOD! IT'S 4096-BIT RSA!
BLAST! OUR EVIL PLAN IS FOILED!
WHAT WOULD ACTUALLY HAPPEN:
HIS LAPTOP'S ENCRYPTED. DRUG HIM AND HIT HIM WITH THIS $5 WRENCH UNTIL HE TELLS US THE PASSWORD.
GOT IT.