UC SANTA CRUZ

Math 134: Cryptography

Lecture 1

Eoin Mackall

January 6, 2024

University of California, Santa Cruz

Table of contents

Introduction

Introduction

Professor: Eoin Mackall
email: emackall@ucsc.edu

Syllabus

Course Overview

Structure: Lecture (MWF) and Discussion (T)

Course Overview

Structure: Lecture (MWF) and Discussion (T)

Objectives:

1. learn how to apply concepts from number theory to cryptography,
2. learn about symmetric-key and public-key cryptography (creation, uses, analyze security),
3. learn about additional cryptographic objects and their applications (hash functions, digital signatures)

Course Overview

Structure: Lecture (MWF) and Discussion (T)

Objectives:

1. learn how to apply concepts from number theory to cryptography,
2. learn about symmetric-key and public-key cryptography (creation, uses, analyze security),
3. learn about additional cryptographic objects and their applications (hash functions, digital signatures)

What do you get out of the course?

Course Overview

Structure: Lecture (MWF) and Discussion (T)

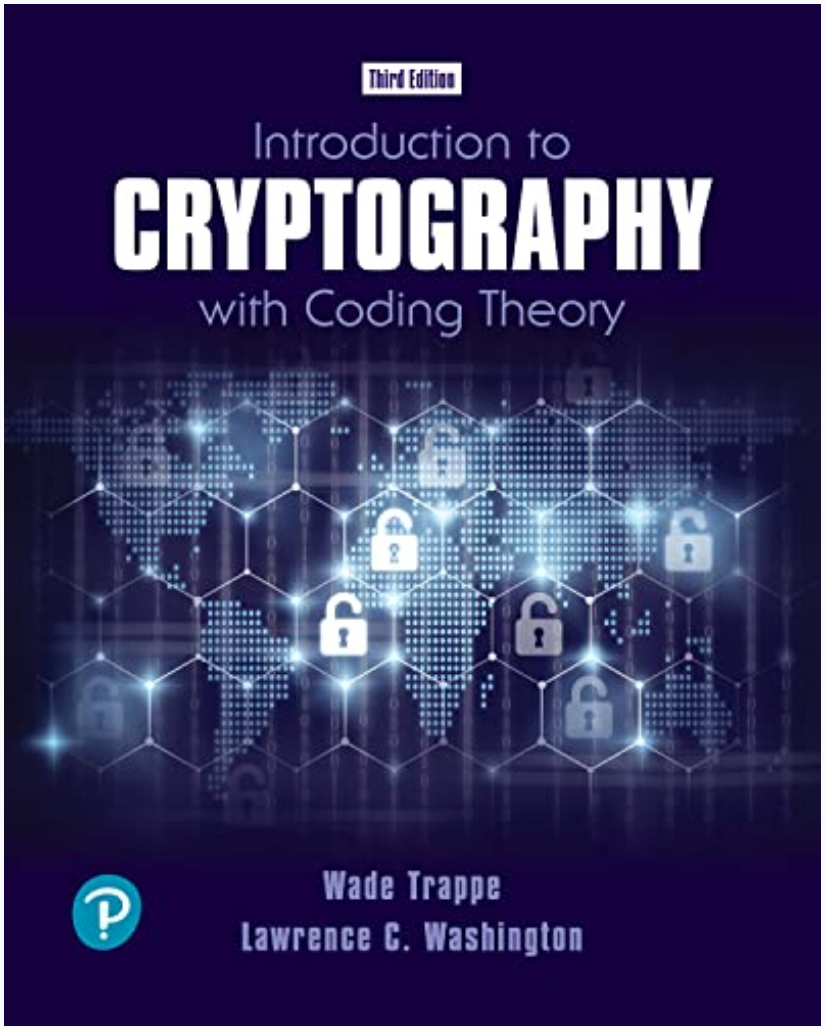
Objectives:

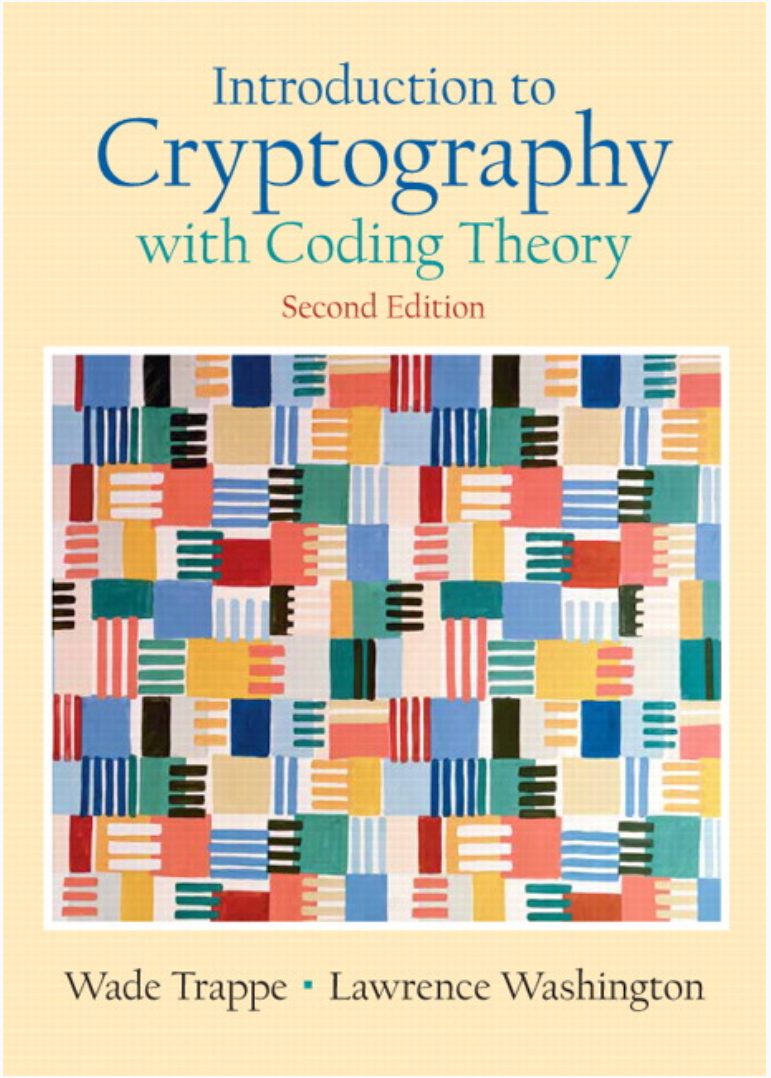
1. learn how to apply concepts from number theory to cryptography,
2. learn about symmetric-key and public-key cryptography (creation, uses, analyze security),
3. learn about additional cryptographic objects and their applications (hash functions, digital signatures)

What do you get out of the course?

1. Abstractly: as much as you put in
2. Concretely: IBM cryptography certification [optional]

Textbook(s)


Third Edition
Introduction to
CRYPTOGRAPHY
with Coding Theory
Wade Trappe
Lawrence C. Washington


Introduction to
Cryptography
with Coding Theory
Second Edition
Wade Trappe · Lawrence Washington

Grading

1. **Assigned Reading: 0%**
   Reading will be assigned weekly. It's important to actually do the readings.

Grading

1. **Assigned Reading: 0%**
   Reading will be assigned weekly. It's important to actually do the readings.
2. **Homework: 0%**
   Assigned weekly, but not collected.

Grading

1. **Assigned Reading: 0%**
   Reading will be assigned weekly. It's important to actually do the readings.
2. **Homework: 0%**
   Assigned weekly, but not collected.
3. **Midterm Exams(2): 60%**
   30% each, lowest midterm dropped

Grading

1. **Assigned Reading: 0%**
   Reading will be assigned weekly. It's important to actually do the readings.
2. **Homework: 0%**
   Assigned weekly, but not collected.
3. **Midterm Exams(2): 60%**
   30% each, lowest midterm dropped
4. **Final Exam(1): 20%**
   Cumulative exam

Grading

1. **Assigned Reading: 0%**
   Reading will be assigned weekly. It's important to actually do the readings.
2. **Homework: 0%**
   Assigned weekly, but not collected.
3. **Midterm Exams(2): 60%**
   30% each, lowest midterm dropped
4. **Final Exam(1): 20%**
   Cumulative exam
5. **Final Project(1): 20%**
   Report on chosen topic from syllabus list.

Cryptography

Ciphers (set-up)

First, we need to agree in what setting we are working.

Ciphers (set-up)

First, we need to agree in what setting we are working.

Imagine that two parties, say **A**lice and **B**ob, want to communicate through some channel.

Ciphers (set-up)

First, we need to agree in what setting we are working.

Imagine that two parties, say **A**lice and **B**ob, want to communicate through some channel.

- Alice and Bob will be sending messages written from some fixed *alphabet*.

Ciphers (set-up)

First, we need to agree in what setting we are working.

Imagine that two parties, say **A**lice and **B**ob, want to communicate through some channel.

- Alice and Bob will be sending messages written from some fixed *alphabet*. E.g. `This is a message.`

Ciphers (set-up)

First, we need to agree in what setting we are working.

Imagine that two parties, say **A**lice and **B**ob, want to communicate through some channel.

- Alice and Bob will be sending messages written from some fixed *alphabet*. E.g. `This is a message.`
- Messages are considered *plaintext*.

Ciphers (set-up)

First, we need to agree in what setting we are working.

Imagine that two parties, say **A**lice and **B**ob, want to communicate through some channel.

- Alice and Bob will be sending messages written from some fixed *alphabet*. E.g. `This is a message.`
- Messages are considered *plaintext*.
- A *cipher* is a function which converts *plaintext* to *ciphertext*.

Ciphers (set-up)

First, we need to agree in what setting we are working.

Imagine that two parties, say **A**lice and **B**ob, want to communicate through some channel.

- Alice and Bob will be sending messages written from some fixed *alphabet*. E.g. `This is a message.`
- Messages are considered *plaintext*.
- A *cipher* is a function which converts *plaintext* to *ciphertext*. E.g. a shift cipher on the english alphabet (shifting by 2) applied to the message above yields `Vjku ku c oguucig.`

Shift ciphers

Alice and Bob are sending messages using the English alphabet. They have the following idea:

- Choose beforehand a number $1 \leq k \leq 25$

Shift ciphers

Alice and Bob are sending messages using the English alphabet. They have the following idea:

- Choose beforehand a number $1 \leq k \leq 25$
- Label the letters of the alphabet from 0 to 25

Shift ciphers

Alice and Bob are sending messages using the English alphabet.
They have the following idea:

- Choose beforehand a number $1 \leq k \leq 25$
- Label the letters of the alphabet from 0 to 25
- For each of the numbers j from 0 to 25, compute $j + k \pmod{26}$

Shift ciphers

Alice and Bob are sending messages using the English alphabet. They have the following idea:

- Choose beforehand a number $1 \leq k \leq 25$
- Label the letters of the alphabet from 0 to 25
- For each of the numbers j from 0 to 25, compute $j + k \pmod{26}$
- For each letter of plaintext corresponding to the number j, replace with the letter corresponding to $j + k \pmod{26}$.

Shift ciphers

Alice and Bob are sending messages using the English alphabet. They have the following idea:

- Choose beforehand a number $1 \leq k \leq 25$
- Label the letters of the alphabet from 0 to 25
- For each of the numbers j from 0 to 25, compute $j + k \pmod{26}$
- For each letter of plaintext corresponding to the number j, replace with the letter corresponding to $j + k \pmod{26}$.
- Send the new ciphertext.

Shift ciphers

Example

We can implement a shift cipher with $k = 2$ by filling out the table below.

Plaintext	a	b	c	d	e	f	g	$\cdots$	x	y	z
j	0	1	2	3	4	5	6	$\cdots$	23	24	25
$j + k \pmod{26}$								$\cdots$			
Ciphertext								$\cdots$			

Shift ciphers

Example

We can implement a shift cipher with $k = 2$ by filling out the table below.

Plaintext	a	b	c	d	e	f	g	$\cdots$	x	y	z
j	0	1	2	3	4	5	6	$\cdots$	23	24	25
$j + k \pmod{26}$	2	3	4	5	6	7	8	$\cdots$	25	0	1
Ciphertext								$\cdots$			

Shift ciphers

Example

We can implement a shift cipher with $k = 2$ by filling out the table below.

Plaintext	a	b	c	d	e	f	g	$\cdots$	x	y	z
j	0	1	2	3	4	5	6	$\cdots$	23	24	25
$j + k \pmod{26}$	2	3	4	5	6	7	8	$\cdots$	25	0	1
Ciphertext	c	d	e	f	g	h	i	$\cdots$	z	a	b

Shift ciphers

Example

We can implement a shift cipher with $k = 2$ by filling out the table below.

Plaintext	a	b	c	d	e	f	g	$\cdots$	x	y	z
j	0	1	2	3	4	5	6	$\cdots$	23	24	25
$j + k \pmod{26}$	2	3	4	5	6	7	8	$\cdots$	25	0	1
Ciphertext	c	d	e	f	g	h	i	$\cdots$	z	a	b

The plaintext `This is a message` is converted to the ciphertext `Vjku ku c oguucig` with this cipher.

Shift ciphers

Example

We can implement a shift cipher with $k = 2$ by filling out the table below.

Plaintext	a	b	c	d	e	f	g	$\cdots$	x	y	z
j	0	1	2	3	4	5	6	$\cdots$	23	24	25
$j + k \pmod{26}$	2	3	4	5	6	7	8	$\cdots$	25	0	1
Ciphertext	c	d	e	f	g	h	i	$\cdots$	z	a	b

The plaintext `This is a message` is converted to the ciphertext `Vjku ku c oguucig` with this cipher.

Check

a goes to *c*, *e* goes to *g*, *m* goes to *o*, etc.

Shift ciphers

Suppose that we're given the ciphertext `Gcvrjv ivru kyv sffb` and we're told that this text was generated from a shift cipher with shift $k = 17$.

How do we get back the plaintext message?

Shift ciphers

Suppose that we're given the ciphertext `Gcvrjv ivru kyv sffb` and we're told that this text was generated from a shift cipher with shift $k = 17$.

How do we get back the plaintext message?

Ciphertext	a	b	c	d	e	f	g	$\cdots$	x	y	z
j	0	1	2	3	4	5	6	$\cdots$	23	24	25
$j - k \pmod{26}$								$\cdots$			
Plaintext								$\cdots$			

Shift ciphers

Suppose that we're given the ciphertext `Gcvrjv ivru kyv sffb` and we're told that this text was generated from a shift cipher with shift $k = 17$.

How do we get back the plaintext message?

Ciphertext	a	b	c	d	e	f	g	$\cdots$	x	y	z
j	0	1	2	3	4	5	6	$\cdots$	23	24	25
$j - k \pmod{26}$	9	10	11	12	13	14	15	$\cdots$	6	7	8
Plaintext								$\cdots$			

Shift ciphers

Suppose that we're given the ciphertext `Gcvrjv ivru kyv sffb` and we're told that this text was generated from a shift cipher with shift $k = 17$.

How do we get back the plaintext message?

Ciphertext	a	b	c	d	e	f	g	$\cdots$	x	y	z
j	0	1	2	3	4	5	6	$\cdots$	23	24	25
$j - k \pmod{26}$	9	10	11	12	13	14	15	$\cdots$	6	7	8
Plaintext	j	k	l	m	n	o	p	$\cdots$	g	h	i

Shift ciphers

Suppose that we're given the ciphertext `Gcvrjv ivru kyv sffb` and we're told that this text was generated from a shift cipher with shift $k = 17$.

How do we get back the plaintext message?

Ciphertext	a	b	c	d	e	f	g	$\cdots$	x	y	z
j	0	1	2	3	4	5	6	$\cdots$	23	24	25
$j - k \pmod{26}$	9	10	11	12	13	14	15	$\cdots$	6	7	8
Plaintext	j	k	l	m	n	o	p	$\cdots$	g	h	i

The ciphertext message `Gcvrjv ivru kyv sffb` is decoded to recover the plaintext message `Please read the book`.

Shift ciphers

More terminology:

Shift ciphers

More terminology:

- In this encryption scheme, we use the integer k to encode the plaintext. We would say that k is our *encryption key*.

Shift ciphers

More terminology:

- In this encryption scheme, we use the integer k to encode the plaintext. We would say that k is our *encryption key*.
- Here we also use the integer k to decode the ciphertext. We would say that k is our *decryption key*.

Shift ciphers

More terminology:

- In this encryption scheme, we use the integer k to encode the plaintext. We would say that k is our *encryption key*.
- Here we also use the integer k to decode the ciphertext. We would say that k is our *decryption key*.
- An encryption algorithm that uses the same key to both encrypt the paintext and decrypt the ciphertext is called a *symmetric-key cryptographic algorithm*.

Shift ciphers

More terminology:

- In this encryption scheme, we use the integer k to encode the plaintext. We would say that k is our *encryption key*.
- Here we also use the integer k to decode the ciphertext. We would say that k is our *decryption key*.
- An encryption algorithm that uses the same key to both encrypt the paintext and decrypt the ciphertext is called a *symmetric-key cryptographic algorithm*.
- In our example, the space of all possible keys for the algorithm was the set of integers $1 \leq k \leq 25$. This is called the *keyspace*.

Shift ciphers

Suppose that an adversary, say **E**ve, wants to “break” this cryptosystem. There are a number of avenues to do so.

Shift ciphers

Suppose that an adversary, say **E**ve, wants to “break” this cryptosystem. There are a number of avenues to do so.

Security

- **Known ciphertext**: If Eve can get ahold of valid ciphertext, a brute force method through all possible decryption keys will yield the plaintext message.

Shift ciphers

Suppose that an adversary, say **E**ve, wants to "break" this cryptosystem. There are a number of avenues to do so.

Security

- **Known ciphertext**: If Eve can get ahold of valid ciphertext, a brute force method through all possible decryption keys will yield the plaintext message.
- **Known plaintext**: If Eve knows how any plaintext alphabet letter is encoded to ciphertext, then she can solve for the encryption key k.

Shift ciphers

Suppose that an adversary, say **E**ve, wants to "break" this cryptosystem. There are a number of avenues to do so.

Security

- **Known ciphertext**: If Eve can get ahold of valid ciphertext, a brute force method through all possible decryption keys will yield the plaintext message.
- **Known plaintext**: If Eve knows how any plaintext alphabet letter is encoded to ciphertext, then she can solve for the encryption key k.
- **Chosen plaintext**: If Eve is allowed to choose plaintext that will be converted to ciphertext, then any character can be sent to give the key.

Shift ciphers

Suppose that an adversary, say **E**ve, wants to "break" this cryptosystem. There are a number of avenues to do so.

Security

- **Known ciphertext**: If Eve can get ahold of valid ciphertext, a brute force method through all possible decryption keys will yield the plaintext message.
- **Known plaintext**: If Eve knows how any plaintext alphabet letter is encoded to ciphertext, then she can solve for the encryption key k.
- **Chosen plaintext**: If Eve is allowed to choose plaintext that will be converted to ciphertext, then any character can be sent to give the key.
- **Chosen ciphertext**: If Eve can choose one piece of ciphertext to decode, then the negative of the result can be used to deduce the encryption key.