

Introduction to Cryptography

Lecture 1

Benny Pinkas

Administrative Details

- Grade
 - Exam 75%
 - Homework 25% (might include programming)
- Office hours: after the class, by appointment
- Email: benny@pinkas.net
- Web page: <http://www.pinkas.net/course.html>

- Goal: Learn the basics of modern cryptography
- Method: introductory, applied, precise.

Bibliography

- Textbook:
 - *Cryptography Theory and Practice, Second edition* by D. Stinson.
- Optional:
 - *Handbook of Applied Cryptography*, by A. Menezes, P. Van Oorschot, S. Vanstone. (Free!)
 - *Introduction to Cryptography Applied to Secure Communication and Commerce*, by Amir Herzberg. (Free!)
 - *Applied Cryptography*, by B. Schneier.

Course Outline

- Course Outline
 - Data secrecy: encryption
 - Symmetric encryption
 - Asymmetric (public key) encryption
 - Data Integrity: authentication, digital signatures.
 - Required background in number theory
 - Cryptographic protocols

Encryption

```

    graph LR
      Alice((Alice)) <--> Bob((Bob))
      Eve((Eve)) --- Link
  
```

- Two parties: Alice and Bob
- Reliable communication link
- Goal: send a message m while hiding it from Eve

October 17, 2004 Introduction to Cryptography, Benny Pinkas page 5

Authentication / Signatures

```

    graph LR
      Alice((Alice)) <--> Bob((Bob))
      Eve((Eve)) --- Link
  
```

- Goal:
 - Enable Bob to verify that Eve did not change messages sent by Alice
 - Enable Bob to prove to others the origin of messages sent by Alice
- (We'll discuss these issues in later classes)

October 17, 2004 Introduction to Cryptography, Benny Pinkas page 6

Encryption

- Message space $\{m\}$
- Encryption key k_1 , decryption key k_2
- Key generation algorithm
- Encryption function E
- Decryption function D

} Define the encryption system

```

    graph LR
      plaintext --> Enc[Encryption E_{k1}]
      Enc -- ciphertext --> Dec[Decryption D_{k2}]
      Dec -- plaintext --> plaintext
  
```

- For every message m
 - $D_{k_2}(E_{k_1}(m)) = m$
 - i.e., the decryption of the encryption of m is m
- Symmetric encryption $k = k_1 = k_2$

October 17, 2004 Introduction to Cryptography, Benny Pinkas page 7

Security Goals

- (1) No adversary can determine m
or, even better,
- (2) No adversary can determine any information about m

- Suppose $m = \text{"attack on Sunday, October 17, 2004"}$.
- The adversary can at most learn that
 - $m = \text{"attack on S**day, Oct**er 17, 2004"}$
 - $m = \text{"***** ** *U***** ***** *** *****"}$.
- Here, goal (1) is satisfied, but not goal (2)

October 17, 2004 Introduction to Cryptography, Benny Pinkas page 8

Adversarial Model

- Adversary Knows encryption and decryption algorithms E and D , and message space.
- Kerckhoff's Principle (1883):
 - The only thing Eve does not know is the secret key k
 - The design is public
 - Allows public scrutiny of the design
 - No need to replace system if the design is exposed -> no need to keep the design secret
 - Same design can be used for multiple applications
 - Focus on securing the key
 - Examples
 - Security by obscurity, Intel's HDCP ☹
 - DES, AES, SSL ☺

October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 9

Adversarial Power

- Types of attacks:
 - Ciphertext only attack – ciphertext known to the adversary (eavesdropping)
 - Known plaintext attack – plaintext and ciphertext are known to the adversary
 - Chosen plaintext attack – the adversary can choose the plaintext and obtain its encryption (e.g. he has access to the encryption system)
 - Chosen ciphertext attack – the adversary can choose the ciphertext and obtain its decryption
- Assume restrictions on the adversary's capabilities, but not that it is using specific attacks or strategies.

October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 10

Breaking the Enigma

- German cipher in WW II
- Kerckhoff's principle
- Known plaintext attack
- (somewhat) chosen plaintext attack



October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 11

Caesar Cipher

- A shift cipher
- Plaintext: "ATTACK AT DAWN"
- Ciphertext: "DWWDFN DW GDZQ"
- Key: $k \in_{\mathbb{R}} \{0, 25\}$. (In this example $k=3$)
- More formally:
 - Key: $k \in_{\mathbb{R}} \{0 \dots 25\}$, chosen at random.
 - Message space: English text (i.e., $\{0 \dots 25\}^{|m|}$)
 - Algorithm: ciphertext letter = plaintext letter + $k \bmod 26$
- Kerckhoff's principle
- Not a good idea

October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 12

Brute Force Attacks

- Brute force attack : adversary tests all key space and checks which key decrypts the message
- Caesar cipher: $|\text{key space}| = 26$
- Need a large key space
- Usually, the key is a bit string chosen uniformly at random from $\{0,1\}^{|k|}$. Implying $2^{|k|}$ equiprobable keys.
- How long should k be?
- The adversary should not be able to do $2^{|k|}$ decryption trials

October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 13

Adversary's computation power

- Theoretically
 - Adversary can perform $\text{poly}(|k|)$ computation
 - Key space = $2^{|k|}$
- Practically
 - $|k| = 64$ is too short
 - $|k| = 80$ starts to be reasonable
 - Why? (what can be done by 1000 computers in a year?)
 - $2^{55} = 2^{20}$ (ops per second)
 - $\times 2^{20}$ (seconds in two weeks)
 - $\times 2^5$ (\approx fortnights in a year) (might invest more than a year..)
 - $\times 2^{10}$ (computers in parallel)
- All this, assuming that the adversary cannot do better than a brute force attack

October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 14

Monoalphabetic Substitution cipher

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Y	A	H	P	O	G	Z	Q	W	B	T	S	F	L	R	C	V	M	U	E	K	J	D	I	X	N

- Plaintext: "ATTACK AT DAWN"
- Ciphertext: "YEEYHT YE PYDL"
- More formally:
 - Plaintext space = ciphertext space = $\{0..25\}^{|m|}$
 - Key space = 1-to-1 mappings of $\{0..25\}$ (i.e., permutations)
 - Encryption: map each letter according to the key
- Key space = $26! \approx 4 \times 10^{28} \approx 2^{95}$. (Large enough.)
- Still easy to break

October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 15

Breaking the substitution cipher

- Plaintext has a lot of structure
 - Known letter distribution in English (e.g. $\text{Pr}("e") = 13\%$).
 - Known distribution of pairs of letters ("th" vs. "jj")



October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 16

Cryptanalysis of a substitution cipher

- QEFPP FP QEB CFOPQ QBUQ
- QEFPP FP QEB CFOPQ QBUQ
- TH TH T T T
- THFP FP THB CFOPT TBUT
- THIS IS TH I ST T T
- THIS IS THB CIOST TBUT
- THIS IS THE I ST TE T
- THIS IS THE FIRST TEXT

The Vigenere cipher

- Plaintext space = ciphertext space = $\{0..25\}^{|m|}$
- Key space = $|k|$ letter strings $\{0..25\}^{|k|}$
- Generate a pad by repeating the key until it is as long as the plaintext (e.g., "SECRETSECRETSEC...")
- Encryption algorithm: add the corresponding characters of the pad and the plaintext
 - THIS IS THE PLAINTEXT TO BE ENCRYPTED
 - SECR ET SEC RETSECRET SE CR ETSECRETSE
- $|Key\ space| = 26^{|k|}$. ($k=17$ implies $|key\ space| \approx 2^{80}$)
- Each plaintext letter is mapped to $|k|$ different letters

Attacking the Vigenere cipher

- Known plaintext attack (or rather, known plaintext distribution)
 - Guess the key length $|k|$
 - Examine every $|k|$ 'th letter, this is a shift cipher
 - THIS IS THE PLAINTEXT TO BE ENCRYPTED
 - SECR ET SEC RETSECRET SE CR ETSECRETSE
 - Attack time: $|k| \times |k| \times$ time of attacking a shift cipher⁽¹⁾
 - Chosen plaintext attack:
 - Use the plaintext "aaaaaa..."
- ⁽¹⁾ Can't assume English plaintext. Can assume known letter frequency

Perfect Cipher

- For a *perfect cipher*, it holds that given ciphertext C ,
 - $Pr(\text{plaintext} = P | C) = Pr(\text{plaintext} = P)$
 - i.e., knowledge of ciphertext does not change the a priori distribution of the plaintext
 - Probabilities taken over key space and plaintext space
- One Time Pad: (for a one bit plaintext)
 - Plaintext $p \in \{0,1\}$
 - Key $k \in_R \{0,1\}$ (i.e. $Pr(k=0) = Pr(k=1) = 1/2$)
 - Ciphertext = $p \oplus k$
- What happens if we know a priori that $Pr(\text{plaintext}=1)=0.8$?

The one-time-pad is a perfect cipher

- ciphertext = plaintext \oplus k
- $Pr(\text{ciphertext} = 1)$
- = $Pr(\text{plaintext} \oplus \text{key} = 1)$
- = $Pr(\text{key} = \text{plaintext} \oplus 1) = 1/2$
- $Pr(\text{plaintext} = 1 | \text{ciphertext} = 1)$
- = $Pr(\text{plaintext} = 1 \ \& \ \text{ciphertext} = 1) / Pr(\text{ciphertext} = 1)$
- = $Pr(\text{plaintext} = 1 \ \& \ \text{ciphertext} = 1) / 1/2$
- = $Pr(\text{ciphertext} = 1 | \text{plaintext} = 1) \cdot Pr(\text{plaintext} = 1) / 1/2$
- = $Pr(\text{key} = 0) \cdot Pr(\text{plaintext} = 1) / 1/2$
- = $1/2 \cdot Pr(\text{plaintext} = 1) / 1/2$
- = $Pr(\text{plaintext} = 1)$

October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 21

The one-time-pad

- Plaintext = $p_1 p_2 \dots p_m \in \Sigma^m$ (e.g. $\Sigma = \{0,1\}$, or $\Sigma = \{A \dots Z\}$)
- key = $k_1 k_2 \dots k_m \in_R \Sigma^m$
- Ciphertext = $c_1 c_2 \dots c_m$, $c_i = p_i \oplus k_i$
- Essentially a shift cipher with a different key for every character
- Shannon [47,49]:
 - An OTP is a perfect cipher, unconditionally secure. ☺
 - As long as the key is a random string, of the same length as the plaintext. ☺
 - Cannot use
 - Shorter key (e.g., Vigenere cipher)
 - Key which is not random

October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 22

What we've learned today

- Introduction
- Kerckhoff's Principle
- Some classic ciphers
 - Brute force attacks
 - Required key length
 - A large key does not guarantee security
- Perfect ciphers

October 17, 2004

Introduction to Cryptography, Benny Pinkas

page 23