

A Gentle Introduction to Cryptography



Outline of these lectures

- ♦ The general goals of cryptographic systems
- ♦ Vulnerabilities of cryptographic systems
- ♦ Two basic categories of cryptographic algorithms:
 - ♦ Symmetric
 - ♦ Asymmetric (public key)
- ♦ Methods for sharing keys (including Diffie-Hellman)



Outline (cont.)

- Methods for ensuring data integrity (hash algorithms)
- Methods for authentication (digital signatures)

The General Goals of Cryptography

- Confidentiality; assuring that only authorized parties are able to understand the data.
- Integrity; ensuring that when a message is sent over a network, the message that arrives is the same as the message that was originally sent.



Goals (cont.)

- Authentication; ensuring that whoever supplies or accesses sensitive data is an authorized party.
- Nonrepudiation; ensuring that the intended recipient actually received the message & ensuring that the sender actually sent the message.



Basic Terms

- Encryption: scrambling a message or data using a specialized cryptographic algorithm.
- Plaintext: the message or data before it gets encrypted.
- Ciphertext: the encrypted (scrambled) version of the message.
- Cipher: the algorithm that does the encryption.



Text

Plainteks (plain.txt):

Ketika saya berjalan-jalan di pantai,
saya menemukan banyak sekali kepiting
yang merangkak menuju laut. Mereka
adalah anak-anak kepiting yang baru
menetas dari dalam pasir. Naluri
mereka mengatakan bahwa laut adalah
tempat kehidupan mereka.

Cipherteks (cipher.txt):

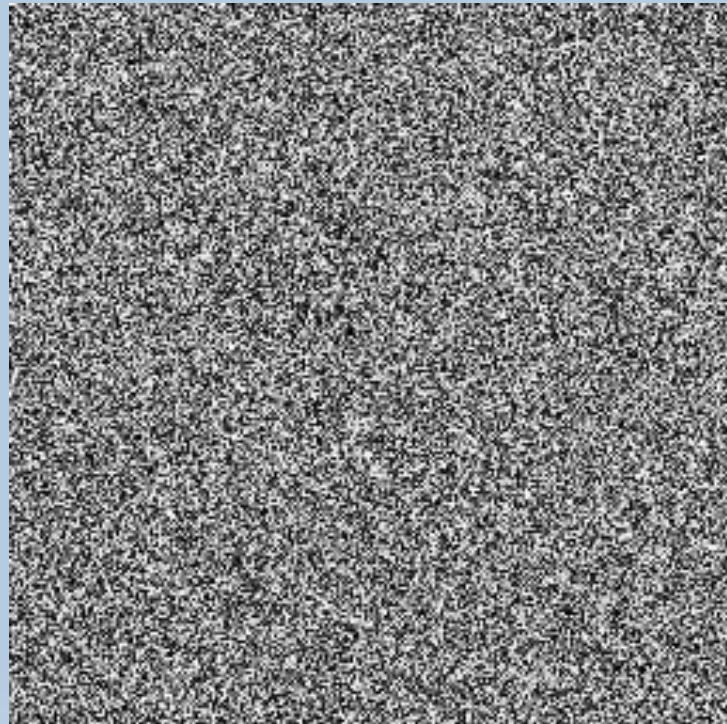
Ztâxzp/épêp/qtüyp{p}<yp{p}/sx/□}âpx;
□□épêp/|t}t|äzp}/qp}êpz/étzp{x/zt□xâx
}v□□ép}v/|tüp}vzpz/|t}äyâ/{päâ=/\tütz
p□□psp{pw/p}pz<p}pz/zt□xâx}v/ép}
v/qpüä□□|t}tâpé/spüx/sp{p|/□péxü=/]
p{äüx□□|ttüzp/|t}vpâpzp}/qpwâp/{päâ
/psp{pw□□ât|□pâ/ztwxsä□p}/|tützp=

Image

Plainteks (malu .bmp):



Cipherteks (malu .bmp):



Database

Plainteks (siswa.dbf):

NIM	Nama	Tinggi	Berat
000001	Elin Jamilah	160	50
000002	Fariz RM	157	49
000003	Taufik Hidayat	176	65
000004	Siti Nurhaliza	172	67
000005	Oma Irama	171	60
000006	Aziz Burhan	181	54
000007	Santi Nursanti	167	59
000008	Cut Yanti	169	61
000009	Ina Sabarina	171	62

Cipherteks (siswa2.dbf):

NIM	Nama	Tinggi	Berat
000001	tüp}vzpz/ t}äyä/{ää	äzp}	épêp
000002	□□ t}tâpé/spüx/sp	péxü=	ztwxsä□
000003	□□ât □â/ztwxsä□p}/	}/ tü	spüx/
000004	épêp/ t}t äzp}/qpêpz	qp}êpz	wxsä
000005	étzp{x/zt□xâx}v□□êp}	pää/psp	étzp{
000006	spüx/sp{p /□péxü=/]	xâx}v	ttüzp/
000007	Ztâxzp/épêp/qtüypp}<	äzp}	}äyä/{
000008	qpwâp/{pää/psp{pw□	Ztwxs	xâx}v□□
000009	}t äzp}/qp}êpz/ép{	qp}êp	äzp}/qp

Keterangan: hanya *field* Nama, Berat, dan Tinggi yang dienkrpsi.

Basic Terms (cont.)

- ♦ Decryption: the process of converting ciphertext back to the original plaintext.
- ♦ Cryptanalysis: the science of breaking cryptographic algorithms.
- ♦ Cryptanalyst: a person who breaks cryptographic codes; also referred to as “the attacker”.



More on Confidentiality

- Confidentiality means that only authorized parties are able to understand the data (authorized from the perspective of the party that encrypted the data).
- It is okay if unauthorized parties know that there is data. It is even okay if they copy the data, so long as they cannot understand it.



Authentication

- ♦ How can we know that a party that provides us with sensitive data is an authorized party?
- ♦ How can we know that the party that is accessing sensitive data is an authorized party?
- ♦ This is a difficult problem on the Internet.
- ♦ Two solutions are:
 - ♦ Passwords
 - ♦ Digital signatures

Integrity

- ♦ This involves ensuring that when a message (or any kind of data, including documents and programs) is sent over a network, the data that arrives is the same as the data that was originally sent. It is important that the data has not been tampered with.
- ♦ Technical solutions include:
 - ♦ Encryption
 - ♦ Hashing algorithms

Nonrepudiation

- ♦ Ensuring that the intended recipient actually got the message.
- ♦ Ensuring that the alleged sender actually sent the message.
- ♦ This is a difficult problem. How do we prove that a person's cryptographic credentials have not been compromised?



An Important Message

- ♦ In theory, some cryptographic algorithms seem to be EXTREMELY secure.
- ♦ Vulnerabilities arise when systems administrators do not deploy the encryption systems securely.
- ♦ A fundamental rule: DON'T CODE YOUR OWN CRYPTOGRAPH ALGORITHMS.
- ♦ Another rule: When using a cryptographic library, use the intuitive user interfaces provided with those libraries.



Message from Cryptlib Developer, Peter Gutman

“The major design philosophy behind the code [behind Cryptlib] is to give users the ability to build secure apps without needing to spend several years learning crypto. ... [T]he important point is that anybody should be able to employ them [important cryptographic algorithms] without too much effort. ...”




Standard Algorithms are Incredibly Secure

- Using a 128 bit key for a symmetric encryption algorithm, there are 2^{128} possible keys.
- Even with the computing resources of the US government, most of the software developers alive today will be dead before the government could break such an encryption [Viega and McGraw]



Incredibly secure (cont.)

- Most security experts believe that 256-bit keys are good for the lifetime of the universe (many billions of years).
 - The problem is that encryption is just one link in the chain of security. Encryption is a really strong link in that chain, but one weak link breaks the chain.
 - It is usually easier for the attacker to hack your machine and steal the plaintext than to break your cipher.
- 
- A decorative silhouette of a mountain range in shades of blue and purple, spanning the bottom of the slide.

A Simple Example

The plaintext:

0	1	0	0	0	0	1	1	0	1	0	0	0	0	1	0	1	0	0	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

The key:

1	1	0	1	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

The ciphertext

1	0	0	1	0	0	1	0	0	0	1	1	1	0	0	0	0	1	1	0	1	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

A Simple Encryption Example

ciphertext:

| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |

XOR'd with key

1	1	0	1	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

yields plaintext

0	1	0	0	0	0	1	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Common Types of Attacks

- ♦ Known cipher attacks: the attacker has the ciphertext and she tries to decrypt the message by generating all possible keys.
 - ♦ Rarely successful because the number of possible keys is enormous.
 - ♦ Also, the decrypted message (for certain types of data) may not be easy to recognize when it appears.



Common Types of Attacks (cont.)

- ♦ Known plaintext attack: the attacker has both the ciphertext and the plaintext.
 - ♦ Again, this is difficult because there are so many keys, but the plaintext information may make experimentation easier than in the previous case.
 - ♦ We are assuming that the attacker knows the algorithm that was used for the encryption.



Common Types of Attacks (cont.)

- Chosen plaintext attacks: The cryptanalyst introduces the plaintext into the system and then watches for how that plaintext will be encrypted.
 - The Allies used this approach in WWII by sending out false messages about allied troop movements.
 - Often the attacker will try to feed a planned sequence of messages that would reveal the most about the way in which the data is being encrypted.



Common Types of Attacks (cont.)

- Side channel attacks use seemingly incidental information that can reveal important information about the key being used.
- Viega and McGraw mention DPA (Differential Power Analysis) attacks on smart cards. A DPA attack analyzes the power output from a processor performing an encryption algorithm in order to get information about the key being used by that algorithm.



Symmetric Cryptography

- ♦ Symmetric algorithms are used for:
 - ♦ Confidentiality
 - ♦ Data integrity
- ♦ Even if an attacker captures the data, the attacker will not be able to manipulate it in any meaningful way.

Symmetric Cryptography (cont.)

- Symmetric algorithms use a single key shared by two communicating parties.
- The shared key must remain secret to ensure the confidentiality of the encrypted data.
- The shared key problem is the main technological challenge for this kind of encryption.
- We will discuss solutions to the key exchange problem a bit later.



Figure A-1 from Viega and McGraw


- The way symmetric encryption works is shown in Figure 1-A from Viega and McGraw.
- The message and the key are provided as input to the encryption algorithm.
- The output is the ciphertext, which can then be transferred over an insecure medium.



Figure 1-A (cont.)

- On the receiver end, the secret key and the ciphertext are inputs to the decryption algorithm.
- The output is the original plaintext.

Symmetric Cryptography (cont.)

- ♦ The secret key must be shared securely. Otherwise, the most sophisticated cryptographic algorithm is useless.
 - ♦ One method of distributing the key is using the sneaker-net.
 - ♦ Protocols exist for exchanging keys over an insecure medium, but care must be taken to assure a good authentication process.
 - ♦ Asymmetric cryptography is a common method for sharing keys.
- 

Symmetric Cryptography

- Two main categories of symmetric algorithms:
 - Block ciphers
 - Stream ciphers
- Most well-known and well-studied symmetric algorithms use block ciphers.
- Block ciphers break up the message into constant-size blocks and encrypt the code block by block.



Block Ciphers

- Typical block sizes are 64 bits or 128 bits.
- Messages are padded (with extra bits) to fit the block size.
- The simplest type of block ciphers work in ECB (Electronic Code Book) mode. In this mode, each block is encrypted separately, independent of the other blocks (like in our simple XOR example).



Block Ciphers (cont.)

- ❖ ECB block ciphers are not secure because given plaintext is always encoded in the same way.
- ❖ Thus, the attacker can look for common linguistic patterns.
- ❖ These patterns can help the attacker to figure out the algorithm and key being used.



Block Ciphers (cont.)

- In CBC (Cipher Block Chaining) mode, blocks are also encrypted one at a time, but the initial state for each block is dependent on the ciphertext of the previous block.
- Thus, the same text will be encrypted in many different ways. This makes it much more difficult for the cryptanalyst to crack the cipher.
- CBC mode is the default mode for many block ciphers.



Block Ciphers (cont.)

- A variety of block cipher modes exist (in addition to CBC) for making sure that repeated plaintext is encoded in different ways throughout the message.
- These modes are the default for the standard secure symmetric encryption algorithms (like DES).



Cipher Block Chaining (CBC) Mode

- By adding gibberish into the middle of the ciphertext, the attacker can interfere with the decryption of a CBC encrypted message.
- Two methods are used to defend against this kind of gibberish attack:
 - Encode the length of the message at the start of the message or elsewhere to help the receiver figure out if the message has been tampered with.
 - Use a cryptographic checksum (or hash) as a signature for your message.

Block Ciphers (cont.)

- The longer the message, the better chance the attacker has of breaking the encryption.
- Bruce Schneier says that a message would have to be at least 34 gigabytes in length for a 64-bit cipher before this would become a genuine risk.



Block Ciphers (cont.)

- Two factors influence the security of a symmetric block cipher:
 - The quality of the algorithm (e.g., ECB mode ciphers are less secure).
 - The length of the key (e.g., 64 bit blocks are questionable, but 128 bit blocks are considered more than adequate).
- There is a classic trade-off between efficiency and security.



Security is Hard to Prove

- Demonstrating how secure a cryptographic algorithm is remains an extremely hard problem.
- The best test seems to be years of experience and public exposure.
- The “one-time pad” method (which has been used in the military) is absolutely secure, but not very practical because the key changes with each communication.



Extended Quote from Lecture Notes

The two basic goals of a cryptographic algorithm are (a) to make life difficult for the attacker and (b) to produce algorithms that are efficient both in terms of space and time. An algorithm that is too inefficient to be used in practice is of little value even if it were proven to be highly secure ...



Quote (cont.)

It is fairly easy for the cryptography researcher to design an algorithm that is secure against all **KNOWN** forms of attack. It is far more difficult to design an algorithm that will be secure against types of attacks that are still **UNKNOWN**. It is nearly impossible to predict new attacks against block ciphers that will be manifesting in future years.




Quote (cont.)

For example, Viega and McGraw state that many people believe that the NSA has developed sophisticated attacks against block ciphers that they have not shared with the rest of the world.



Block Size

- A 64-bit cipher is considered too small for high security applications. According to Bruce Schneier (back in 1995), an organization such as the NSA could break a 64-bit key in under one minute.
 - A 256-bit key is believed to be secure enough that a computer made of all the matter in the universe computing for the entire lifetime of the universe would have an infinitesimal probability of finding a key by brute force.
- 
- A decorative graphic at the bottom of the slide showing a range of mountains in various shades of blue and purple, receding into the distance.

Quantum Computing

But, then there is always
Quantum Computing



Important Commercial Algorithms

- ♦ The most important symmetric algorithms from a commercial point of view are:
 - ♦ DES (Data Encryption Standard)
 - ♦ 3DES
 - ♦ AES (Advanced Encryption Standard)




DES

- This has been a US government standard for many years (although recently complimented with AES).
- It uses a 64-bit key (actually, only 56 bits are used for the encryption, the other 8 bits are parity bits), so it is no longer viable.
- Increased processing speeds (in recent years) are making brute force attacks on DES more viable.



3DES

- ♦ Then, came the idea of using DES twice on a given message.
 - ♦ A subtle form of attack was discovered which made 2DES no better than DES.
 - ♦ 3DES proved to have the properties that 2DES was supposed to have.
 - ♦ 3DES is a viable and popular symmetric block algorithm.
 - ♦ 3DES has one downside: it is inefficient.
- 
- A decorative graphic at the bottom of the slide showing a range of mountains in various shades of blue and purple, creating a layered, misty effect.

DES According to Denning



DES According to Denning

GET THE IDEA?



3DES

- Despite the fact that it is inefficient, 3DES is considered a very good (and it is a very popular) choice for encryption.
- Several good implementations of 3DES are easily downloaded off the Internet.

AES

- The NIST (National Institute of Standards and Technology) ran a competition for a new encryption standard.
- The winners were announced in October 2000. They were Joan Daemen and Vincent Rijmen. Their algorithm is called Rijndael or AES (Advanced Encryption Standard).
- AES is now an accepted federal standard and is widely available in open source form. Implementations are available in C++ and Java.

AES (cont.)

- 3DES still has the advantage that it has been studied (in DES) form for many years.
- The guestimate is that AES will be a viable encryption standard for the next 50 years, but there could be some surprises down the raod.




The Key Distribution Problem

- For symmetric ciphers, each pair of communicating agents needs a unique key.
- If there are lots of users, this creates a key management problem.
- Key derivation algorithms are used to generate a unique key for each communicating pair.
- If the master key for the key derivation algorithm is compromised, you've got a major problem.



Key Distribution (cont.)

- Some have even attacked derived keys in a key distribution system to get the master key.
 - Another approach is to use a key management system that generates session keys for each communication. Even for the same communicating pair, the session keys will change from session to session.
 - Kerberos, from MIT, is a highly regarded open source computer security product that supports symmetric key management.
- 
- A decorative graphic at the bottom of the slide showing a range of mountains in various shades of blue and purple, creating a layered, misty effect.

The Great Philosopher, Yogi Berra

“It is difficult to make predictions, especially about the future.”

