

# CIT 424

# Computer System Security

01/26/15  
Spring 2015



# Introduction to Cryptography

# Outline

- What is data encryption?
- Cryptanalysis
- Basic encryption methods
  - Substitution ciphers
  - Permutation ciphers

# Introduction to Encryption

- Much of computer security is about keeping secrets
- One method is to make the secret hard for others to read
- While (usually) making it simple for authorized parties to read

# Encryption

- Encryption is the process of hiding information in plain sight
- Transform the secret data into something else
- Even if the attacker can see the transformed data, he can't understand the underlying secret

# Encryption and Data Transformations

- Encryption is all about transforming the data
- One bit or byte pattern is transformed to another bit or byte pattern
- Usually in a reversible way

# Encryption Terminology

- Encryption is typically described in terms of sending a message
  - Though it's used for many other purposes
- The sender is  $S$
- The receiver is  $R$
- And the attacker is  $O$

# More Terminology

- *Encryption* is the process of making message unreadable/unalterable by  $O$
- *Decryption* is the process of making the encrypted message readable by  $R$
- A system performing these transformations is a *cryptosystem*
  - Rules for transformation sometimes called a *cipher*



# Plaintext and Ciphertext

- *Plaintext* is the original form of the message (often referred to as  $P$ )

Transfer \$100 to my  
savings account

- *Ciphertext* is the encrypted form of the message (often referred to as  $C$ )

Sqzmredq #099 sn lx  
rzuhmfr zbbntms

# Very Basics of Encryption Algorithms

- Most algorithms use a *key* to perform encryption and decryption
  - Referred to as  $K$
- The key is a secret
- Without the key, decryption is hard
- With the key, decryption is easy

# Terminology for Encryption Algorithms

- The encryption algorithm is referred to as  $E()$
- $C = E(K, P)$
- The decryption algorithm is referred to as  $D()$ 
  - Sometimes the same algorithm as  $E()$
- The decryption algorithm also has a key

# Symmetric and Asymmetric Encryption Systems

- Symmetric systems use the same keys for E and D :

$$P = D(K, C)$$

$$\text{Expanding, } P = D(K, E(K, P))$$

- Asymmetric systems use different keys for E and D:

$$C = E(K_E, P)$$

$$P = D(K_D, C)$$

# Characteristics of Keyed Encryption Systems

- If you change only the key, a given plaintext encrypts to a different ciphertext
  - Same applies to decryption
- Decryption should be hard without knowing the key

# Cryptanalysis

- The process of trying to break a cryptosystem
- Finding the meaning of an encrypted message without being given the key
- To build a strong cryptosystem, you must understand cryptanalysis

# Forms of Cryptanalysis

- Analyze an encrypted message and deduce its contents
- Analyze one or more encrypted messages to find a common key
- Analyze a cryptosystem to find a fundamental flaw

# Breaking Cryptosystems

- Most cryptosystems are breakable
- Some just cost more to break than others
- The job of the cryptosystem designer is to make the cost infeasible
  - Or incommensurate with the benefit extracted



# Types of Attacks on Cryptosystems

- Ciphertext only
- Known plaintext
- Chosen plaintext
  - Differential cryptanalysis
- Algorithm and ciphertext
  - Timing attacks
- In many cases, the intent is to guess the key

# Ciphertext Only

- No *a priore* knowledge of plaintext
- Or details of algorithm
- Must work with probability distributions, patterns of common characters, etc.
- Hardest type of attack

# Known Plaintext

- Full or partial
- Cryptanalyst has matching sample of ciphertext and plaintext
- Or may know something about what ciphertext represents
  - E.g., an IP packet with its headers

# Chosen Plaintext

- Cryptanalyst can submit chosen samples of plaintext to the cryptosystem
- And recover the resulting ciphertext
- Clever choices of plaintext may reveal many details
- Differential cryptanalysis iteratively uses varying plaintexts to break the cryptosystem
  - By observing effects of controlled changes in the offered plaintext

# Algorithm and Ciphertext

- Cryptanalyst knows the algorithm and has a sample of ciphertext
- But not the key, and cannot get any more similar ciphertext
- Can use “exhaustive” runs of algorithm against guesses at plaintext
- Password guessers often work this way
- *Brute force attacks* – try every possible key to see which one works

# Timing Attacks

- Usually assume knowledge of algorithm
- And ability to watch algorithm encrypting/decrypting
- Some algorithms perform different operations based on key values
- Watch timing to try to deduce keys
- Successful against some smart card crypto
- Similarly, observe power use by hardware while it is performing cryptography

# Basic Encryption Methods

- Substitutions
  - Monoalphabetic
  - Polyalphabetic
- Permutations

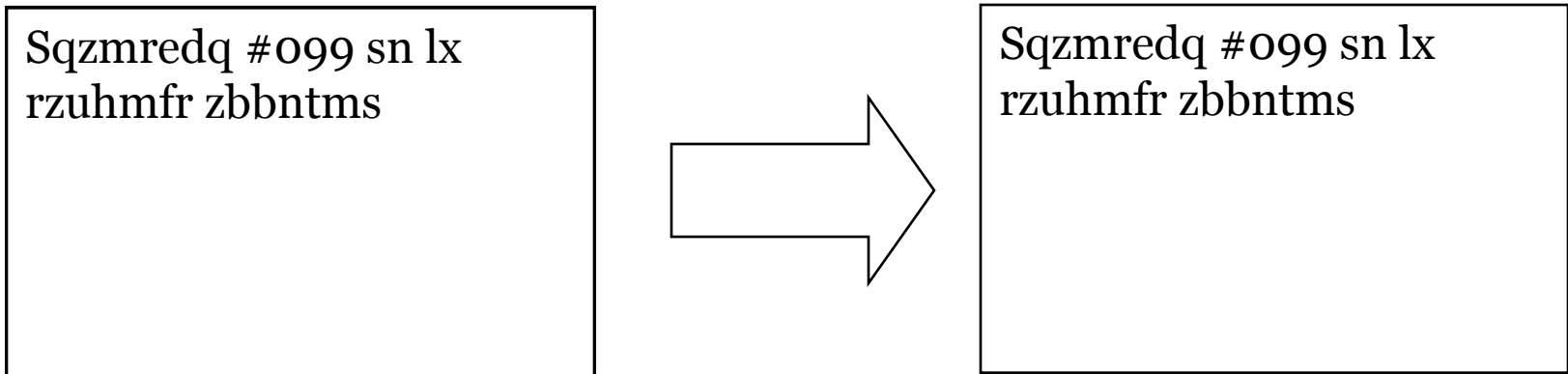
# Substitution Ciphers

- Substitute one or more characters in a message with one or more different characters
- Using some set of rules
- Decryption is performed by reversing the substitutions



# Example of a Simple Substitution Cipher

How did this transformation happen?



Every letter was changed to the  
“previous lower” letter

# Caesar Ciphers

- A simple substitution cipher like the previous example
  - Supposedly invented by Julius Caesar
- Translate each letter a fixed number of positions in the alphabet
- Reverse by translating in opposite direction

# Is the Caesar Cipher a Good Cipher?

- Well, it worked great 2000 years ago
- It's simple, but
- It's simple
- Fails to conceal many important characteristics of the message
- Which makes cryptanalysis easier
- Limited number of useful keys

# How Would Cryptanalysis Attack a Caesar Cipher?

- Letter frequencies
- In English (and other alphabetic languages), some letters occur more frequently than others
- Caesar ciphers translate all occurrences of a given plaintext letter into the same ciphertext letter
- All you need is the offset

# More On Frequency Distributions

- In most languages, some letters used more than others
  - In English, “e,” “t,” and “s” are common
- True even in non-natural languages
  - Certain characters appear frequently in C code
  - Zero appears often in numeric data

# Cryptanalysis and Frequency Distribution

- If you know what kind of data was encrypted, you can (often) use frequency distributions to break it
- Especially for Caesar ciphers
  - And other simple substitution-based encryption algorithms

# Breaking Caesar Ciphers

- Identify (or guess) the kind of data
- Count frequency of each encrypted symbol
- Match to observed frequencies of unencrypted symbols in similar plaintext
- Provides probable mapping of cipher
- The more ciphertext available, the more reliable this technique

# Example

- With ciphertext “Sqzmredq #099 sn lx rzuhmfr zbbntms”
- Frequencies -

[illegible]



# Applying Frequencies To Our Example

a	0		b	2		c	0		d	1		e	1
f	1		g	0		h	1		i	0		j	0
k	0		l	1		m	3		n	2		o	0
p	0		q	2		r	3		s	3		t	1
u	1		v	0		w	0		x	1		y	0
z	3												

- The most common English letters are typically “e,” “t,” “a,” “o,” and “s”
- Four out of five of the common English letters in the plaintext map to these letters

# Cracking the Caesar Cipher

- Since all substitutions are offset by the same amount, just need to figure out how much
- How about +1?
  - That would only work for  $a \Rightarrow b$
- How about -1?
  - That would work for  $t \Rightarrow s$ ,  $a \Rightarrow z$ ,  $o \Rightarrow n$ , and  $s \Rightarrow r$
  - Try it on the whole message and see if it looks good

# More Complex Substitutions

- Monoalphabetic substitutions
  - Each plaintext letter maps to a single, unique ciphertext letter
- Any mapping is permitted
- Key can provide method of determining the mapping
  - Key could be the mapping

# Are These Monoalphabetic Ciphers Better?

- Only a little
- Finding the mapping for one character doesn't give you all mappings
- But the same simple techniques can be used to find the other mappings
- Generally insufficient for anything serious

# Codes and Monoalphabetic Ciphers

- Codes are sometimes considered different than ciphers
- A series of important words or phrases are replaced with meaningless words or phrases
- E.g., “Transfer \$100 to my savings account” becomes
  - “The hawk flies at midnight”

# Are Codes More Secure?

- Frequency attacks based on letters don't work
- But frequency attacks based on phrases may
- And other tricks may cause problems
- In some ways, just a limited form of substitution cipher
- Weakness based on need for codebook
  - Can your codebook contain all message components?

# Superencipherment

- First translate message using a code book
- Then encipher the result
- If opponent can't break the cipher, great
- If he can, he still has to break the code
- Depending on several factors, may (or may not) be better than just a cipher
- Popular during WWII (but the Allies still read Japan's and Germany's messages)