ITIS 3200

Class Notes

Sep 20 2018

Cryptography

* Introduction
  + Cryptography: process of making and using codes to secure transmission of information
  + Encryption: converting original message into a form unreadable by unauthorized individuals
  + Cryptanalysis: process of obtaining original message from encrypted message without knowing algorithms
  + Cryptology: science of encryption; combines cryptography and cryptanalysis
* Cryptography: Basic Terminology
  + Algorithm set of mathematical rules used in encryption and decryption
  + Cryptography science of secret writing that enables you to store and transmit data
  + Cryptanalysis practice of obtaining plaintext from ciphertext without a key or breaking the encryption. The breaking of codes.
  + Cryptology
  + Decipher act of transforming data into a readable format
  + Key secret sequence of bits and instructions

Plaintext input ->X-> Secret key shared by sender and recipient (K) Encryption algorithm (e.g., DES) -> Transmitted ciphertext (Y = E[K,X] -> Secret key shared by sender and recipient (K) Decryption algorithm (reverse of encryption algorithm). -> Plaintext output

Figure 2.1 Simplified Model of Symmetric Encryption

Attacking Symmetric Encryption

* Cryptanalytic Attacks
  + Rely on:
    - Nature of the algorithm
    - Some knowledge of the general characteristics of the plaintext
    - Some sample plaintext-ciphertext pairs
  + Exploits the characteristics of the algorithm to attempt to deduce a specific plaintext or the key being used
    - If successful all future and past messages encrypted with that key are compromised
* Brute – Force Attacks
  + Try all possible

Table 2.1

|  |  |  |  |
| --- | --- | --- | --- |
|  | DES | Triple DES | AES |
| Plaintext block size (bits) | 64 | 64 | 128 |
| Ciphertext block size (bits) | 64 | 64 | 128 |
| Key size (bits) | 56 | 112 or 168 | 128, 192, or 256 |

* DES = Data Encryption Standard
* AES = Advanced Encryption Standard

Comparison of Three Popular Symmetric Encryption Algorithms

Data Encryption Standard (DES)

* Until recently was the most widely used encryption scheme
  + FIPS PUB 46
  + Referred to as the Data Encryption Algorithm (DEA)

Table 2.2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Key Size (bits) | Cipher | Number of Alternative Keys | Time Required at 109 decryptions/s | Time Required at 1013 decryptions/s |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Triple DES (3DES)

* Repeats basic DES algorithm three times using either two or three.

Advanced Encryption Standard (AES)

* Needed a replacement for 3DES
  + 3DES was not reasonable for long term use.
* NIST called for proposals for a new AES in 1997
  + Should have a security strength equal to or better than 3DES

Practical Security Issues

* Typically, symmetric encryption is applied to a unit of data larger than a single 64bit or 128bit block
* Electronic codebook (ECB) mode is the simplest approach to multiple-block encryption
  + Each block of plaintext is encrypted using the same key
  + Cryptanalysts may be able to exploit regularities in the plaintext
* Modes of operation
  + Alternative techniques developed to increase the security of symmetric block encryption for large sequences
  + Overcomes the weaknesses of ECB

Block & Stream Ciphers

* Block Cipher
  + Processes the input one block of elements at a time
  + Produces an output block for each input block
  + Can reuse key

Message Authentication

* Protects against active attacks
* Verifies received message is authentic
  + Contents have not been altered
  + From authentic source
  + Timely and in correct sequence
* Can use conventional encryption
  + Only sender and receiver share a key

Message Authentication Without Confidentiality

* Message encryption by itself does not provide a secure form of authentication
* It is possible to combine authentication and confidentiality in a single algorithm by encrypting a message plus its authentication tag
* Typically, message authentication is provided as a separate function from message encryption
* Situations in which message authentication without confidentiality may be preferable include:
  + There are a number of applications in which the same message is broadcast to a number of destinations.

Message -> MAC algorithm (K) -> MAC -> Transmit -> MAC Algorithm

To be useful for message authentication, a hash function H

Electronic Codebook (ECB)

* The simplest
* Cipher Block Chaining (CBC)

Hash Functions

* Hash functions are extremely useful and appear in almost all information security applications.
* A hash function is a mathematical function that converts a numerical input value into another compressed numerical value. The input to the hash function is of arbitrary length but output is always of fixed length.
* Values returned by a hash function are called message digest or simply hash values. The following picture illustrated hash function

Message M

Features of Hash Functions

* Fixed Length Output (Hash Value)
  + Hash function coverts data of arbitrary length to a fixed length. This process is often referred to as hashing the data.
  + In general, the hash is much smaller than the input data, hence hash functions are sometimes called compression functions.
  + Since a hash is a smaller representation

Properties of Hash Functions

* Pre-Image Resistance
  + This property means that it should be computationally hard to reverse a hash function.
  + In other words, if a hash function h produced a hash value z, then it should be a difficult process to find any input value x that hashes to z.
  + This property protects against an attacker
* Second Pre-Image Resistance
  + This property means given an input and its hash, it should be hard to find a different input with the same hash. In other words, if a hash function h for an input x produces hash value h(x)
* Collision Resistance
  + This property means it should be hard to find two different inputs of any length that results in the same hash. This property is also referred to as collision free hash function.
  + In other words, for a hash function h, it is hard

Design of Hashing Algorithms

* At the heart of hashing is a mathematical function that operates on two fixed-size blocks of data to create.

Data Block -> Mathematical Hash Function -> Hash Value

Popular Hash Functions

* Message Digest (MD)
  + MD5
* Secure Hash Function

Applications of Hash Functions

* Password Storage
  + Hash functions provide protection to password storage.
  + Instead of storing password in clear, mostly all logon processes store the hash values of passwords in the file.
  + The Password file consists of a table of pairs which are in the form (user id, h(P)).

An intruder can only see the hashes of passwords, even if he accessed the password. He can neither logon using hash nor can he derive the password from hash value since hash function possesses the property of pre-image resistance.

Data Integrity Check

* Data integrity check is a most common application of the hash functions.

Public-Key Encryption Structure

* Publicly proposed by Diffie and Hellman in 1976
* Based on mathematical functions.
* Plaintext
  + Readable message or data that is fed into the algorithm as input.
* Encryption algorithm
  + Performs transformations on the plaintext
* Public and private key
  + Pair of keys, one for encryption, one for decryption.
* Ciphertext
  + Scrambled message produced as output
* Decryption key
  + Produces the original plaintext
* User encrypts data using his or her own private key
* Anyone who knows the corresponding public key will be able to decrypt the message

Table 2.3

* Application for Public-Key Cryptosystems

Requirements for public Key Cryptosystems

* Computationally easy to create key pairs
* Computationally easy for sender knowing public key to encrypt messages

Asymmetric Encryption Algorithms

* RSA (Rivest, Shamir, Adelman)

Digital Signatures

* NIST FIPS PUB 186-4 defines a digital signature as: “The result of a cryptographic transformation of data that, when properly implemented, provides a mechanism for verifying.

Random Numbers

* Uses include generation of:
  + Keys for public-key algorithms
  + Stream key for symmetric stream cipher
  + Symmetric key for use as a temporary

Random Number Requirements

* Randomness
  + Criteria:
    - Uniform distribution
      * Frequency of occurrence

Practical Application: Encryption of Stored Data

* Common to encrypt transmitted data
  + Much less common for stored data
    - Approaches to encrypt stored data:

Summary