Applied Cryptography Primer

Ning Shang, @syncomo

Version: 2017

Outline

1. Applied Cryptography and Security Protocols

2. Cryptographic Protocols

Outline

1. Applied Cryptography and Security Protocols

2. Cryptographic Protocols

Cryptography: Definition

Cryptography is the study of mathematics techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication. [1]
Cryptography is not the only means of providing information security, but rather one set of techniques.

[1] This is the definition of cryptography in the *Handbook of Applied Cryptography (HAC)*.

Goals of Cryptography

- The most fundamental problem cryptography addresses: ensure security of communication over insecure medium.
- Goals of cryptography: address the following areas in both theory and practice
  - Confidentiality, privacy, secrecy
  - Data integrity
  - Authentication
  - Non-repudiation

Encoding Vs. Encryption

- **Encoding** is about representation of a message
- **Encryption** is about hiding a message

A Taxonomy of Cryptographic Primitives

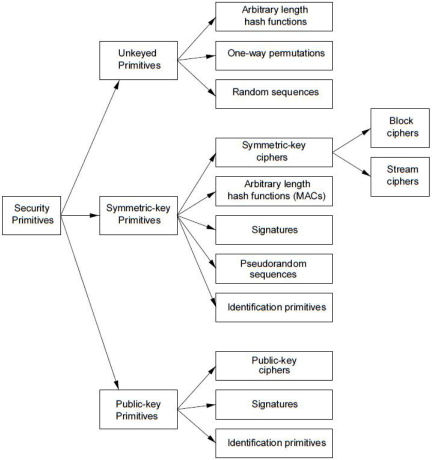


Figure: A taxonomy of cryptographic primitives, by the HAC

Encryption

The Figure below provides a simple model of a two-party communication using encryption.

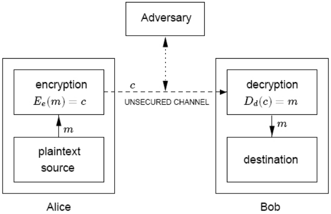


Figure: Schematic of a two-party communication using encryption

Intuitively,

- Symmetric (secret-key) encryption: $e = d$.
- Asymmetric (public-key) encryption: $e \neq d$.

Block Ciphers (Secret-Key)

- Block cipher algorithms operate on a block
  - DES uses 64-bit blocks, with 56-bit key
  - AES uses 128-bit blocks, with a key of length 128, 192, or 256 bits
- Security of block ciphers
  - When a random key is picked, the cipher should behave like a random mapping

Block Cipher Modes

- To encrypt a variable-length message using a block cipher, the data must first be divided into blocks.
- A block cipher mode specifies the process of encrypting each of these blocks.

Initialization Vector (IV)

An IV is a (usually random) fixed-size sequence used in most of the cipher modes

- IV does not need to be secret
- IV randomizes encryption
- IV shall not be used twice

Recommendation of block cipher and modes

General rules

- Use AEAD (authenticated encryption with associated data) whenever you can
- Consult an expert if you are unsure about the cipher or mode

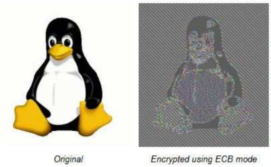


Figure: An image encrypted with ECB mode. Courtesy of "Wikipedia::Block cipher modes of operation".

Encryption: Secret-Key Vs. Public-Key

- Secret-key encryption
  - Secret key exchange is usually difficult
  - Fast
- Public-key encryption
  - Secret key exchange is not needed
  - Much slower than secret-key encryption algorithms
  - Most commonly used for transport of secret keys used for bulk data encryption by symmetric algorithms, and for encrypting small data items[2]

[2] E.g., credit card number and PINs.

Achieving Data Integrity and Data Origin Authentication

- Unkeyed approach: use cryptographic hash functions
  - Send the hash value of a message securely
- Symmetric approach: use Message Authentication Code (MAC)
- Asymmetric approach: use digital signatures
  - A digital signature is a mathematical scheme for demonstrating the authenticity of a digital message or document

Cryptographic Nonce and Replay Attacks

Nonce

- A time-variant number that is used only once
- Often used in authentication protocols

Nonce is to ensure that old communications cannot be replayed as new.

Key Management

- Initialization of system users within a domain
- Generation, distribution, and provisioning of keying materials
- Controlling the use of keying material
- Update, revocation, and destruction of keying material
- Storage, backup/recovery, and archival of keying material

Key Management (Cont'd)

Key management is the most difficult thing to get right in building a secure system.

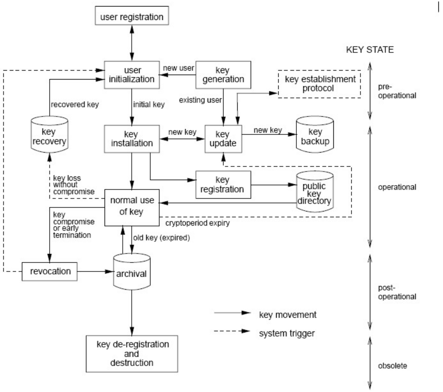


Figure: Key management life cycle.

Key Management is Hard

1. Key management is hard
2. Key management is really hard

Some key management bottom lines

- Stored secret keys must be secured so as to provide both confidentiality and authenticity
- Stored public keys must be secured such that the authenticity is verifiable
- Dependencies among keying material should be avoided.
    - Key management system should be able to “fail gracefully”, i.e., compromise of one key does not affect others
        - Oh Nine, Eff Nine: 09 F9 11 02 9D 74 E3 5B D8 41 56 C5 63 56 88 C0 This is an encryption key used for the DRM of HD DVDs and Blu-ray Discs, made public on many websites.

Cryptography DOs and DON'Ts

Do: Follow best practices and recommendations

- Use well established crypto libraries; use the API correctly: ask for help if you do not understand something
- Use a strong random number generator
- Use standard protocols
  - TLS, IPsec, OAuth
- Leverage a crypto expert

Cryptography DOs and DON'Ts (Cont'd)

Don't

- Don't roll your own
  - Crypto design is hard, and usually error-prone
  - Writing correct crypto code is hard
  - If you are in doubt, ask for help
- Don't use non-secure crypto algorithms for non-crypto purposes
  - Use of known bad or weak algorithms hurts a company's reputation
- Don't use unnecessary crypto/obfuscation
  - Better to use no crypto than poorly thought-out crypto
    - False sense of security to users
    - False sense of security to developers
    - Attackers will eventually figure out
    - Causes confusion

Outline

1. Applied Cryptography and Security Protocols

2. Cryptographic Protocols

Entity Authentication: What Is It?

- **Entity Authentication:** Binding of identity to subject[3]
- Basis of entity authentication
  - Something known.
    Passwords, ID numbers
  - Something possessed.
    National ID card, smart card
  - Something inherent.
    Biometrics, source IP, restricted area terminal

[3] Other names of entity authentication are *identification* and *identity verification*.

Entity Authentication: How It Works

This is how entity authentication works in general.

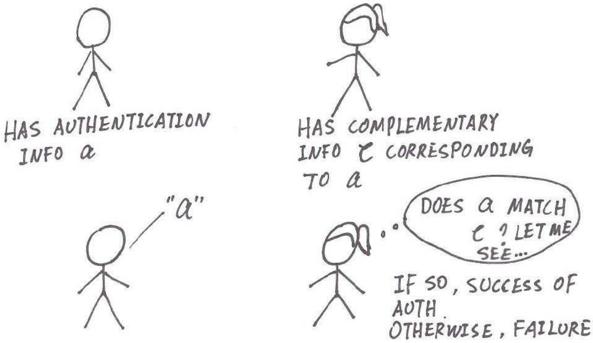

HAS AUTHENTICATION INFO a
HAS COMPLEMENTARY INFO c CORRESPONDING TO a
"a"
DOES a MATCH c ? LET ME SEE...
IF SO, SUCCESS OF AUTH. OTHERWISE, FAILURE

Figure: An illustration of entity authentication

Authentication Vs. Identity

- **Authentication:** binding of identity to subject
  - What is *identity*?
- **Principal:** unique entity
- **Identity:** specifies a principal

Representing Identity

- Randomly chosen
- User-chosen
- Hierarchical: disambiguate based on levels
  - File names in a file system
  - Email addresses
    - foobar@dputech.com
  - X.509v3: Distinguished Names
    - /O=DAPU/OU=InfoSec Department/CN=shangning

Validating Identity

- The problem: Does identity match principal?
- A solution: *certificates*
  - Certificate: Identity validated to belong to known principal
  - Certificate Authority (CA): Certificate Issuer
  - CA is trusted

CA : Certificates $\sim$ Public Security Department: National ID

Public Key Certificate

The term *certificate* refers to “a document that attests to the truth of something or the ownership of something.”
A public key certificate is a certificate attests to the legitimate ownership of a public key and attributes a public key to a principal.

- A digitally signed data structure that attests to the true ownership of a public key
- Identity validated to belong to a known principal
- A principal can be
  - A person
  - A hardware device or
  - Any other entity

Certificate Authority

A certificate authority (or certification authority) (CA)

- Issues, and possibly revokes, public key certificates
- Recognized and trusted by a community of users
- Obliged to verify a certificate applicant's credentials

A CA is responsible for claiming *"Yes, this person is who they say they are, and we, the CA, verify that."*

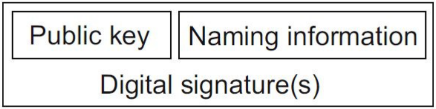
Examples of CAs that issue SSL certificates

- VeriSign (acquired Thawte and GeoTrust)
- GoDaddy
- Comodo

Public Key Infrastructure

- An infrastructure that can be used to issue, validate, and revoke public keys and public key certificates
- Consists of
  - Agreed-upon standards
  - CAs and structures among multiple CAs
  - Methods to discover and validate certificate paths
  - Operational and management protocols
  - Interoperable tools
  - Supporting legislation

Information Contained in a Public Key Certificate



Public key

Naming information

Digital signature(s)

Figure: A public key certificate comprising three pieces of information

A public key certificate comprises at least three pieces of information

- A public key
- Some naming information
  - Used to identify the owner of the public key certificate
  - Contains representation of identity
- One or more digital signatures
  - Ties the public key and the naming information together

X.509 Hierarchical Trust Model

X.509 hierarchical trust model

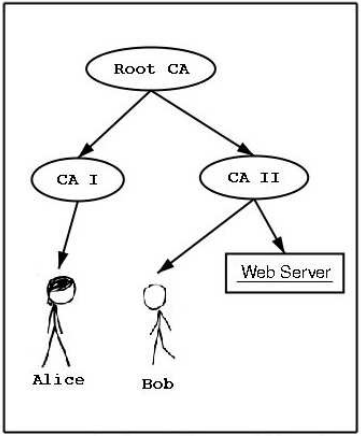


Figure: X.509 certificate chains

Trust Model in TLS and SSL

- As of today, the X.509 trust model is prevailing.

What are SSL and TLS?

- Protocols that provide end-to-end communication security over the Internet.
  - Confidentiality
  - Data integrity
  - Data origin authentication
  - Entity authentication
- **SSL:** Secure Socket Layer
  - Originally developed by Netscape in 1995 to provide secure and authenticated connections between browsers and servers
- **TLS:** Transport Layer Security
  - IETF made SSLv3 an open standard in 1999, and called it TLSv1

- There are security vulnerabilities in SSLv1 and SSLv2
- Use the latest available TSL version in product
- Do not use broken ciphers in SSL/TLS

SSL/TLS Features

- Two types of functions
  - Establish a secure connection between communicating parties
  - Use this connection to securely transmit higher layer protocol data from the sender to the recipient
- Either server-only authentication or server-client authentication is allowed
  - Server-only authentication: server sends certificate
  - Server-client authentication: client sends certificate as well
- SSL is not a single protocol: it is composed of a few subprotocols in two sublayers

SSL/TLS Connections and SSL Sessions

- **SSL connection**
  - A one-time transport of information between two peers
  - Connections are transient
  - Every connection is associated with a *session*
- **SSL session**
  - A session is created by the SSL handshaking protocol
  - Multiple connections can exist in one session
  - A session is characterized by a set of security parameters that apply to all the connections in the session
  - The concept of a session eliminates the need for negotiating the security parameters for each separate connection

SSL/TLS Implementations

- OpenSSL (BSD)
- BoringSSL (Google's fork of OpenSSL)
- SChannel (Microsoft)
- SharkSSL, mbedTLS (for embedded devices)
- Etc.