Hashing

- **Primary Goal:** Integrity Protection
- Guarding against modifications
- Hash functions *classify* an input based on its contents
  - Classification is repeated on the receiver side, if the results match, tampering unlikely
  - Simplest example: Even/odd parity
    - 1000 => parity 1; 0011 => parity 0; 1101 => parity 1
- **Digest:** A reduced form of the body of data.
  - Typically 100 – 1000 bits long.

Cryptographic Hash Functions

- Sender and Receiver agree on a hash function
  - Data and a hash value sent
  - Receiver separates the data and recomputes the hash value and compares with the received one
  - An intruder must not know the hash function!
- **NIST Standard:** SHA (Secure Hash Function)
  - Input length: $<2^{64}$ bits.
  - Digest: 160 bits.
Secure Hash Function (SHA)

1. Pad the message with random 1’s & 0’s
   - Add 64 bits indicating original message length
   - Total message length a multiple of 512
2. Process message in 16x32 bit long words
   - W(0), W(1), ..., W(15)
3. Expand W(1),..., W(15) to 80 words
   - W(t)=W(t-3)+W(t-8)+W(t-14)+W(t-16) [15<t<80]
4. Apply further transformations to each block and compute a message digest

SHA-1 algorithm

- Initialize hash value for this chunk:
  a := h0; b := h1; c := h2; d := h3; e := h4;
- Main loop:
  for i from 0 to 79
  if 0 ≤ i ≤ 19
     f := (b and c) or ((not b) and d)
     k := 0x5A827999
  else if 20 ≤ i ≤ 39
     f := b xor c xor d
     k := 0x6ED9EBA1
  else if 40 ≤ i ≤ 59
     f := (b and c) or (b and d) or (c and d)
     k := 0x8F1BBCDC
  else if 60 ≤ i ≤ 79
     f := b xor c xor d
     k := 0xCA62C1D6
  temp := (a leftrotate 5) + f + e + k + w(i)
  e := d; d := c; c := b leftrotate 30; b := a; a := temp;
- Add this chunk’s hash to result so far:
  h0 := h0 + a;  h1 := h1 + b;  h2 := h2 + c;  h3 := h3 + d;  h4 := h4 + e;
- digest = hash = h0 append h1 append h2 append h3 append h4 (expressed as big-endian)

Note: All variables are unsigned 32 bits and wrap modulo 2^32 when calculating
- Initialize variables:
  h0 := 0x67452301
  h1 := 0xEFCDAB89
  h2 := 0x98BADCFE
  h3 := 0x10325476
  h4 := 0xC3D2E1F0

SHA (2)

- It is unlikely that an intruder will manage to change ALL the right bits.
- Information is diffused throughout the message.
- A desired change causes an avalanche of changes.
- The goal of a good hash function is to generate different value of a digest for as many meaningful messages as possible.
Key Escrow and Clipper

Escrowed Encryption:

Plaintext → Encryption → Ciphertext → Decryption → Plaintext

K

Escrow Agency

Plaintext → Encryption → Ciphertext → Decryption → Plaintext

Clipper Chip (Fortezza)

- Clipper proposal:
  - Split every encryption key into halves.
  - Each half stored by a different agency.
  - Authorized court order may allow delivering key halves to law enforcement.
  - Only hardware embodiments of the algorithm sold.
  - Security by obscurity.

Clipper Message

M, k, n, A, LEAF

- 1. Intercept M + LEAF.
- 2. Using f, decrypt and obtain n (message ID), A.
- 3. Deliver n and the court order to escrow agencies.
- 4. Receive two halves of key u.
- 5. Decrypt the key k using u.
- 6. Decrypt M using key k.
Information Assurance

- Introduction
- Authentication and Identification
- Principles of Cryptography
- Modern Cryptography
- Secure Information Exchange

Communication Protocols

- Protocols separate the Process of accomplishing the task from the Mechanisms by which it is done
  - Specify the Rules of Behavior
  - Arbitrated protocols:
    - Actively involve a disinterested third party.
  - Adjudicated Protocols
  - Self-Enforcing Protocols

Symmetric Key Exchange

- Without a Server, the old key is used to encrypt and exchange the new key
- When a Repository (Key Server) is available:
  1. A->R: (A, B, Ia)
  2. R->A: E(Ka, (Ia, B, Kab, E(Kb, (Kab,A))))
     Only A can open this message
  3. A->B: E(Kb, (Kab, A))
     Only B can open this message
  4. A and B communicate using the SESSION key Kab
Assymetric Key Exchange (Without a Server)

- A has Ea and Da (Public and Private Keys)
- B has Eb and Db (Public and Private Keys)
- A can send Eb(M) to B
  - Typically slow. A symmetric session key desirable.
- A -> B: Eb(Kab)
  - No assurance that this message came from A!
- Therefore: A->B: Eb(Da(Kab))
  - Communication continues using Kab

Certificates

- Used for the Establishment of Trust
- Two parties can “trust” each other, if they “know” someone in common
- A Public Key and someone’s identity are bound together in a Certificate
  - Signed by someone certifying the accuracy of the binding
  - Need to trust the top authority

The Concept of Hierarchies
Practical Digital Certificates

- A certificate is a digital document, published by the Certificate Authority.
- Contains public keys (accessibility).
- Information about the public key owner.
- Information about the issuer of the certificate.
- Issuing authority's signature ensures authenticity.

Digital Signatures

- A Protocol that produces the same effect as a Real Signature
- Only the sender can make it
- Others can recognize it belongs to the sender
- Not forgeable
  - No one else can produce the same signature
- Authentic
  - Easy to check that the signature belongs to the sender, the signature firmly attached to the message
  - Not alterable (once sent it cannot be changed)
  - Not reusable (can be used only once)

Public Key Digital Signature

- Public Keys ideally suited for Signatures
- S sends a signed message M to R:
  - S -> R: Ds(M)
  - Not forgeable and Authentic: Only S could have sent this message.
  - Message is not private (use two encryptions).
  - Not alterable.
  - Time stamps can be used to guard against replays.
Practical Digital Signatures
- Uses Asymmetric Keys combined with a Hash
- Inserts Encrypted Hash into a file along with the Verification (Public) Key
- Provides integrity services for the data that contains the signature
- CAN provide evidence for non-repudiation

Digital Signatures
Data
Private signing key
ENCRIPTS
Completed Digital Signature

VERIFICATION
New Hash computed

Definition of PKI
- A Public Key Infrastructure (PKI) is an environment consisting of protocols, services and standards that support applications of public key cryptography.
- Typically a collection of Certificate Authorities (CA), associated repositories, and governing policies and policy bodies that are used to manage the distribution and trust relationship between digital certificates.
Motivation for PKI

- PKI provides evidentiary services that can prove non-repudiation
- PKI provides encryption services for protecting electronic information in transmission and storage states
- PKI provides the ability to affix digital signatures to electronic information, which can replace traditional paper based business practices
- PKI provides a mechanism to ensure data integrity
- PKI provides enhanced user identification by binding a certificate to an entity (person, computer, etc.)

Typical Uses for PKI

- Securing e-mail (S/MIME)
- Securing transactions between business partners and end clients over public networks, such as the Internet
- Securing stored data through folder-based encryption, encrypted databases
- Integrity for electronic data such as e-mails, financial exchanges, and mission critical business transactions through digital signatures
- User authentication and access control through certificate-based identification
- Non-repudiation - Digital Certificates validate their users’ identities, making it nearly impossible to later repudiate a digitally “signed” transaction, such as a purchase made on a web site

Why is PKI needed? (2)

“In its 1999 survey, the Computer Security Institute estimated the total financial losses by the 163 businesses it surveyed from computer security breaches at $123.7 million. …E-commerce has become so important that firms, including Sedgwick Group PLC (in cooperation with IBM), Lloyds of London, and Network Risk Management Services, are now offering ‘hacker insurance’ “

Source: Michael A. Vatis, Director of the National Infrastructure Protection Center (NIPC), Federal Bureau of Investigation, - Congressional Statement on NIPC Cyber Threat Assessment, October 1999
PKI Encryption

Combination of symmetric encryption for speed and asymmetric encryption for delivery

Recipient's Private Key
Decrypts symmetric key

Encrypted Message

Recipient's Public Key
Repository

Public Key

Symmetric Key

Encrypted Message

One time symmetric session key

Recipient's Public Key
Used to encrypt symmetric key

Encrypts Encrypted Message and key

Recipient's Public Key

Repository

Recipient

Registration Authority (RA)

Certificate Authority (CA)

Practical Key Distribution

1. Subscriber provides proof of identity to the RA
2. RA verifies identity and informs CA
3. CA generates and signs the subscriber's unique certificate
4. CA securely sends private certificate to subscriber and posts public certificate in the repository

X.509 Certificate Standard

<table>
<thead>
<tr>
<th>Version</th>
<th>Serial Number</th>
<th>Signature Algorithm</th>
<th>Issuer Name</th>
<th>Validity Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>JoeU001</td>
<td>SHA-1</td>
<td>c=US, o=ACME Inc.</td>
<td>03/01/2002</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subject</th>
<th>E-mail Address</th>
<th>Telephone Number</th>
<th>Business Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Doe</td>
<td><a href="mailto:jdoe@acme.com">jdoe@acme.com</a></td>
<td>555-555-5555</td>
<td>Security</td>
</tr>
</tbody>
</table>

CA Signature

John Doe's Public Certificate

3082a509dc3746b2198139fff59156736571237

John Doe's Unique Identifier

Validity Period

03/01/2002
Certificate Uses

- Encryption Pair
  - A Public Certificate is used to Encrypt
  - A Private Certificate is used to Decrypt
- Signature Pair
  - A Private Certificate is used to sign (Encrypt the Hash)
  - A Public Certificate is used to verify (Decrypt the Hash for comparison)

Certificate Verification

- Certificate of End Entity (i.e. User)

- Issuer’s Certificate (i.e. The CA)

- Certificate Revocation List (CRL)

PKI vulnerabilities

- Likelihood of misusing the private key.
  - Key management and key security are paramount.
    - If protected by a PIN, and the PIN gets compromised.
    - Private keys stored on local disks (security).
Secure Electronic Transaction Protocol

- SET was developed in 1996, as a part of Visa and MasterCard’s initiative to promote security on the Internet and build consumer trust in e-commerce.
- It is designed to provide an unrivalled level of security and authentication for Internet payments.

SET Definition

What is a SET?
- Secure Electronic Transaction (SET) is an Open Encryption and Security Specification (available at http://www.setco.org/) designed to protect Credit-Card Transactions on the internet.
- A wide range of companies were involved in developing the initial specification, including IBM, Microsoft, Netscape, RSA, Terisa and Verisign.

Objectives of SET

- Confidentiality: Encryption of information provides the confidentiality of information.
- Authentication: Digital signatures and Certificates provide the means to authenticate other participants in the transaction.
- Message Integrity: Digital signatures are used to detect tampering.
- Interoperability: An open set of protocols is used to provide interoperability between implementations of different vendors.
Participants in a SET

- Card holder: Consumer doing the purchase
- Issuer: Financial Institution that issues the card to the card holder.
- Merchant: Online Store providing the goods.
- Payment Gateway: System operated by the Acquirer, that handles the financial requests from the interacts with the Issuer on the Merchants behalf.
- Certificate Authority: Well-known web site that can generate and validate certificates.

SET in Action

1. Customer opens a master card or visa bank account.
2. Customer receives a digital certificate.
3. Third-party merchants also receives certificates.
4. Customer places an order over a web page, by phone, etc.
5. The customers browser received and confirms from the merchants certificate that the merchant is valid.
6. The browser sends the order information.
7. The merchant verifies the customer by checking the digital signature on the customers certificate.
SET in Action (2)

8. The merchant sends the order message to the bank.
9. The bank verifies the merchant and the message.
10. The bank digitally signs and sends authorization to the merchant who can then fill the order.