Key Management and Distribution

Raj Jain
Washington University in Saint Louis
Saint Louis, MO 63130
Jain@cse.wustl.edu

Audio/Video recordings of this lecture are available at:
http://www.cse.wustl.edu/~jain/cse571-11/
Overview

1. Distribution of Private Keys
2. Distribution of Public Keys
3. Public Key Infrastructure: PKI and PKIX
4. X.509 Certificates
5. Certificate revocation

Key Distribution

- Symmetric schemes require parties to share a secret key
  - n Parties \(\Rightarrow\) n(n-1)/2 pairs \(\Rightarrow\) n(n-1)/2 keys

- Public key schemes require parties to acquire valid public keys. How to trust a public key?

- Once “master” secret keys are setup, they are used only to exchange “session” secret keys.
  - Session keys are used for a short time and then discarded.

\[
\begin{align*}
\text{I am A. My nonce is } N_1 & \\
\text{ID}_A \, || \, N_1 & \\
& \text{B} \\
\text{Let’s use session key } K_s. \text{ My nonce is } N_2. & \\
E(K_m, [K_s \, || \, \text{ID}_A \, || \, \text{ID}_B \, || \, f(N_1) \, || \, N_2]) & \\
\text{I am able to decrypt using the key. Here is a fn of your Nonce.} & \\
E(K_s, f(N_2)) &
\end{align*}
\]
Key Distribution Using KDC

- Central authority, called “Key Distribution Center” (KDC)
- Everyone has a shared secret key with KDC

- Hierarchies of KDC’s required for large networks
Public key cryptosystems are inefficient

- So almost never used for direct data encryption
- Rather used to encrypt secret keys for distribution

This scheme is subject to **man in the middle attack**
Secret Key Distribution with Confidentiality and Authentication

Hi I am A. My nonce is $N_1$.

$E(PU_B, [N_1 \ || \ ID_A])$

Here is my nonce. Prove that you have private key for A

$E(PU_A, [N_1 \ || \ N_2])$

Here is your nonce.

$E(PU_B, N_2)$

Now that we know each other. Let’s use session key $K_S$.

$E(PU_B, E(PRA, K_S))$
Hybrid Key Distribution

- Retain use of private-key KDC
- Shares secret master key with each user
- Distributes session key using master key
- Public-key used to distribute master keys
  - Especially useful with widely distributed users
- Rationale
  - Performance
  - Backward compatibility
Distribution of Public Keys

1. Public announcement: Forgery possible
2. Publicly available directory: Message can be tampered with.
3. Public-key authority: users know public key for the directory

A

I want to talk B. Time is T₁
Request || T₁
Here is the certified public key for B
E(PRᵦ, [PUᵦ || Request || T₁])
Hi, I am A. My nonce is N₁.
E(PUᵦ, [IDᵦ || N₁])

Authority

Time is T₂. What’s A’s public key?
Request || T₂
Here is the certified public key for A
E(PRᵦ, [PUᵦ || T₂])

B

Here is my nonce. Prove that you are A.
E(PUᵦ, [N₁ || N₂])
I was able to decrypt. Here is your nonce.
E(PUᵦ, N₂)

Requires real-time access to directory when keys are needed
Public-Key Certificates

- Certificates allow key exchange without real-time access to public-key authority
- A certificate binds **identity** to **public key**
- All contents **signed** by a trusted Public-Key or Certificate Authority (CA)
- Can be verified by anyone who knows the public-key authorities public-key

Hi I AM A

ID_A

Here is your certificate

\[ C_A = E(PR_{auth}, [T_{1} || ID_A || PU_A]) \]

Hi I AM B

ID_B

Here is your certificate

\[ C_B = E(PR_{auth}, [T_{2} || ID_B || PU_B]) \]

Here is my certificate with my public key.

C_A

Here is my certificate with my public key

C_B
PKI, PKIX, and X.509

- **PKI**: Infrastructure to find public keys
  - S/MIME, PGP, SSL use asymmetric cryptography and make use of PKI
  - Certificate authorities
  - Standards for certificates
- **X.509**: ISO standard for Certificate formats
- **PKIX** is the IETF group on PKI
- PKIX adopted X.509 and a subset of its options
- PKIX is a "Profile" of X.509
- TLS, IPSec, SSH, HTTPS, Smartcard, EAP, CableLabs, use X.509
### Root Certificates

![Certificate Manager](image)

You have certificates on file that identify these certificate authorities:

<table>
<thead>
<tr>
<th>Certificate Name</th>
<th>Security Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>ValiCert, Inc.</td>
<td>Builtin Object Token</td>
</tr>
<tr>
<td>VeriSign, Inc.</td>
<td>Builtin Object Token</td>
</tr>
<tr>
<td>Verisign Class 3 Public Primary Certific...</td>
<td>Builtin Object Token</td>
</tr>
<tr>
<td>Verisign Class 3 Public Primary Certific...</td>
<td>Builtin Object Token</td>
</tr>
<tr>
<td>Verisign Class 4 Public Primary Certific...</td>
<td>Builtin Object Token</td>
</tr>
<tr>
<td>Verisign Class 2 Public Primary Certific...</td>
<td>Builtin Object Token</td>
</tr>
<tr>
<td>Verisign Class 1 Public Primary Certific...</td>
<td>Builtin Object Token</td>
</tr>
</tbody>
</table>
Sample X.509 Certificate

Internet Explorer

Certificate Information

This certificate is intended for the following purpose(s):

• Ensures the identity of a remote computer
• Proves your identity to a remote computer
• Protects e-mail messages
• Ensures software came from software publisher
• Protects software from alteration after publication
• All issuance policies

Issued to: VeriSign Class 3 Public Primary Certification Authority - G5

Issued by: VeriSign Class 3 Public Primary Certification Authority - G5

Valid from 11/7/2006 to 7/16/2036

OK
<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>V3</td>
</tr>
<tr>
<td>Serial number</td>
<td>18 da d1 9e 26 7d e8 bb 4a 21…</td>
</tr>
<tr>
<td>Signature algorithm</td>
<td>sha1RSA</td>
</tr>
<tr>
<td>Issuer</td>
<td>VeriSign Class 3 Public Primary …</td>
</tr>
<tr>
<td>Valid from</td>
<td>Tuesday, November 07, 2006…</td>
</tr>
<tr>
<td>Valid to</td>
<td>Wednesday, July 16, 2036 6:…</td>
</tr>
<tr>
<td>Subject</td>
<td>VeriSign Class 3 Public Primary …</td>
</tr>
<tr>
<td>Public key</td>
<td>RSA (2048 Bits)</td>
</tr>
<tr>
<td>version</td>
<td></td>
</tr>
<tr>
<td>Serial number</td>
<td>18 da d1 9e 26 7d e8 bb 4a 21…</td>
</tr>
<tr>
<td>Signature algorithm</td>
<td>sha1RSA</td>
</tr>
<tr>
<td>Issuer</td>
<td>VeriSign Class 3 Public Primary …</td>
</tr>
<tr>
<td>Valid from</td>
<td>Tuesday, November 07, 2006…</td>
</tr>
<tr>
<td>Valid to</td>
<td>Wednesday, July 16, 2036 6:…</td>
</tr>
<tr>
<td>Subject</td>
<td>VeriSign Class 3 Public Primary …</td>
</tr>
<tr>
<td>Public key</td>
<td>RSA (2048 Bits)</td>
</tr>
</tbody>
</table>
X.509 Certificates

- Issued by a Certification Authority (CA), containing:
  - Version V (1, 2, or 3)
  - Serial number SN (unique within CA) identifying certificate
  - Signature algorithm identifier AI
  - Issuer X.500 name CA)
  - Period of validity TA (from - to dates)
  - Subject X.500 name A (name of owner)
  - Subject public-key info Ap (algorithm, parameters, key)
  - Issuer unique identifier (v2+)
  - Subject unique identifier (v2+)
  - Extension fields (v3)
  - Signature (of hash of all fields in certificate)

- Notation CA<<A>> denotes certificate for A signed by CA
CA Hierarchy

- CA's must form a hierarchy
- Each CA has certificates for clients (forward) and parent (backward)
- Each client trusts parents certificates
- Enable verification of any certificate from one CA by users of all other CAs in hierarchy
X.509 Version 3

- Additional information is needed in a certificate
  - Email/URL, Policy details, Usage constraints
- Rather than explicitly naming new fields defined a general extension method
- Extensions consist of:
  - Extension identifier
  - Criticality indicator
  - Extension value
X.509 Extensions

- Authority Key Identifier: Serial # of CA's key
- Subject Key Identifier: Uniquely identifies the subject's key. Serial # or hash.
- Key Usage: Allowed usage - email, business, ...
- Private Key Usage Period: Timestamps for when key can be used (similar to validity)
- Certificate Policies
- Policy Mappings: from Issuer's domain to subject's domain
- Subject Alt Name: Alternative name. DNS.
- Subject Directory Attributes: Other attributes
Certificate Revocation

- May need to revoke before expiry, for example,
  a. User's private key is compromised
  b. User is no longer certified by this CA
  c. CA's certificate is compromised
- CA’s maintain list of revoked certificates
  - Certificate Revocation List (CRL)
- Users should check certificates with CA’s CRL
  - Too much traffic on the net $\Rightarrow$ Not used
- **On-Line Revocation Server (OLRS):**
  - On-line Certificate Status Protocol (OCSP) [RFC 2560]
  - Provides current information
  - Also allows chaining of OCSP responders
# Entrusted Certificates

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>V3</td>
</tr>
<tr>
<td>Serial number</td>
<td>75 0e 40 ff 97 f0 47 ed f5 56 c...</td>
</tr>
<tr>
<td>Signature algorithm</td>
<td>md5RSA</td>
</tr>
<tr>
<td>Issuer</td>
<td>VeriSign Commercial Software ...</td>
</tr>
<tr>
<td>Valid from</td>
<td>Tuesday, January 30, 2001 7:...</td>
</tr>
<tr>
<td>Valid to</td>
<td>Thursday, January 31, 2002 6...</td>
</tr>
<tr>
<td>Subject</td>
<td>Microsoft Corporation, Microso...</td>
</tr>
<tr>
<td>Public key</td>
<td>RSA (1024 Bits)</td>
</tr>
<tr>
<td>Basic Constraints</td>
<td>Subject Type=End Entity, Pat...</td>
</tr>
<tr>
<td>Key Usage</td>
<td>Digital Signature, Key Encipher...</td>
</tr>
<tr>
<td>Authority Key Identifier</td>
<td>KeyID=7b 96 e4 d1 43 fd 68 9...</td>
</tr>
<tr>
<td>Basic Constraints</td>
<td>Subject Type=End Entity, Pat...</td>
</tr>
<tr>
<td>SpcFinancialCriteria</td>
<td>Financial Information=Availabl...</td>
</tr>
<tr>
<td>Key Usage Restriction</td>
<td>[1]Cert PolicyId=1.3.6.1.4.1....</td>
</tr>
<tr>
<td>SpcSpAgencyInfo</td>
<td>Policy Information:URL=https:...</td>
</tr>
<tr>
<td>Thumbprint algorithm</td>
<td>sha1</td>
</tr>
<tr>
<td>Thumbprint</td>
<td>7d 7f 44 14 cc ef 16 8a df 6b f...</td>
</tr>
<tr>
<td>Friendly name</td>
<td>Fraudulent, NOT Microsoft</td>
</tr>
<tr>
<td>Extended Error Information</td>
<td>Revocation Status : The certifi...</td>
</tr>
</tbody>
</table>
1. **Master keys** are used to exchange **session keys**. Session keys are used for a short duration and then discarded.

2. Secret keys are distributed via a **KDC** or via public keys.

3. Public keys are distributed via **X.509 based PKI**. Browsers have a built-in list of **root CAs**.

4. **PKIX** is a profile of the X.509 PKI standard.

5. X.509 uses **X.500** names. DNS names in Alternate Name field.

6. Certificate Revocation Lists (**CRLs**) are used to revoke a certificate. On-line certification Status Protocol (**OCSP**) can be used to check revocation.
Homework 14A

- Study the root certificates in your Internet Explorer
  - Find the certificate for “Thawte Premium Server CA”
  - a. What is the X.500 name of the CA?
  - b. What version of X.509 does this CA use?
  - c. What are the uses of the public key in this certificate?
  - d. What signature algorithm is used to sign this certificate?
  - e. What are the last 4 bytes of the public key
Homework 14B

You will receive a signed email from the TA with his digital certificate. Import this certificate in your contacts list. (Use help feature on your email software for details. See instructions for Outlook and Gmail). Now send an encrypted signed email to TA with the subject line of “CSE571S Encrypted Signed Mail Homework 14B”

You will need a certificate for yourself too.
Lab Homework 14B (Cont)

Getting your Certificate:

- Use **Internet Explorer** to request and collect a free email certificate from:
  

- After you have collected the certificate, in Internet Explorer go to Tools → Internet Options → Contents → Certificates → Personal

- Select your certificate and export it to a file.
  
  Select “Yes – Export the private key” click next
  
  Select “Include all certificates in the certification path”
  
  Select “Enable strong protection”
  
  Do not select “Delete the private key if the export is successful”
  
  Save it with a password of your choice.

- Import this certificate in Outlook as follows:
  
  Tools → Options → Security → Import/Export

- Browse to your certificate file and add it.
Lab Homework 14B (Cont)

- If you use Firefox, use the following procedure to request and collect a free email certificate from:
  

- After you have collected the certificate, in Firefox go to Tools → Options → Advanced → Encryption → View Certificates → Your Certificates

- Select your certificate and backup to a file. Save it with a password of your choice.

- Import this certificate in Outlook as follows: Tools → Options → Security → Import/Export

- Browse to your certificate file and add it. Note: You have to use the same browser to collect the certificate from Comodo that you used to request the certificate.
Importing Other’s Certificates in Outlook:

- In Outlook, open the signed message received from TA. In the message window, right click on the name in the “From field” and select "save as outlook contact"
- This will open a new contact window. In that window, click on the "certificates" tab.
- You will see the certificate listed there.
- Save this contact in your contacts list.
- When you reply or send email to this contact, you can enable the security options for encryption and signatures by: 
  View → Options → Security Options
  Select Encrypt Message or Add Digital Signature or both
  Select Security Settings: <Automatic>
Lab Homework 14B (Cont)

Gmail Instructions:

- The certificate will show up as an attachment name smime.p7s
- Download and save this attachment on your computer.
- Transfer this file to the computer where you have an outlook email.
- Manually create a new contact entry in outlook with proper name and email address.
- Open this contact entry. Go to certificate panel and import. Select all files *.* and select the file smime.p7s
- Save and close the entry.
- To send an email with your Gmail address in the from field, you will need to create a new email account in Outlook with the corresponding Gmail address in the from field. Outlook allows email security. Gmail does not.
Lab Homework 14B (Cont)

Sending Encrypted and Signed Messages w Outlook:

- You can reply to the TA’s email with a signed encrypted message. Content of the reply is not important.

- Before sending the message, on the message window, Select View → Options → Security Settings
  Select encryption and signature
  Now send the message.