PKI

Tom Longstaff

CERT Coordination Center
Software Engineering Institute
Carnegie Mellon University
Pittsburgh PA 1521

The CERT Coordination Center is sponsored by the Advanced Research Projects Agency (ARPA). The Software Engineering Institute is sponsored by the U.S. Department of Defense.

Reading:

Text Chapters 7, 8, 11 (7-8 were previously assigned - just review if you've already read them)

Intro to crypto

Like access control in that crypto controls access to information

Unlike access control, crypto is applied to the data itself, not just an attribute on the data

Think of crypto as a tool like access control to protect access and use of information

Types of Encryption Systems

Two basic types:

- Shared (or symmetric) key encryption
- Public (or asymmetric) key encryption

Shared: use of a single key for both encryption and decryption that both parties must *share*

- Tends to be more efficient
- Used for block ciphers

Public: different keys used for encryption and decryption

- Most popular form is based on RSA or Diffie Helman
- More computational intensive (uses exponentiation)
- Frequently used for symmetric key exchange

PKI

Mechanism to distribute and trust public keys

Two types in common use: Hierarchical and the Web of Trust

Modified Hierarchical combines distinct Hierarchical PKIs with cross-realm authentication

Common use of PKI refers to Hierarchical, but also covers Web of Trust and Modified Hierarchical

Using Public Key for Signatures

A digital signature is a cryptographically strong hash of a longer data set

- E.g., MD5 used by tripwire and others to verify the integrity of the information

If you create a digital signature to a document, then encrypt it with your *private* key, anyone can verify two properties of this information:

- Integrity (through the MD5 checksum)
- Source (only the owner of the private key could have encrypted the signature)

A digital signature, signed with a private key on a public key becomes a *trust verifier* for that key

Carnegie Mellon University
CERT Coordination Center

Signed Keys

Example:

Alice has an asymmetric key pair - creates an MD5 checksum of the key and encrypts it with her private key

Alice:
K{public},(K[MD5]{public})K{private}

Bob (K') verifies the authenticity of the key, then encrypts the MD5 with his private key as well

K{pubic},(K[MD5]{public})K{private} },(K[MD5]{public}) K'{private}

And so on

7

Carnegie Mellon University
CERT Coordination Center

Other properties in signed keys

In addition to the MD5, other properties may be included in the private encrypted part of the public key record

- **Level of trust**
- **Relationship with the key owner**
- **Link to other CA information**

K{pubic},(K[MD5]{public},owner,email)K{private} },(K[MD5]{public},moderate trust,email,authoritative CA) K'{private}

8

Web of trust

From these building blocks, a web of trust can be built:

- **Two users cross-sign each other's public keys**
- **Alice signs Bob who signs Charlie who signs Dain who signs Alice**
- **A particularly trusted user signs many keys**

If you have a small number of individuals you trust, you can build a bridge to a new recipient

This is the principle behind pgp

Other pgp attributes

Trust of a key you are signing

Trust of a key you receive

Key rings

Key servers

PGP designed to sign static documents, not live transactions but the PKI built up with PGP can be used to exchange a session key for a live block cypher

- **E.g., pgp phone.**

Carnegie Mellon University
CERT Coordination Center

Key and signature revocation

What if a private key is compromised in the web of trust?

First of all, need a mechanism to distribute this information

Secondly, need to invalidate all signatures under this key

May be able to limit the extent of revocation based on date of the revocation certificate

11

Carnegie Mellon University
CERT Coordination Center

Building up a hierarchy of keys

In a hierarchical PKI, you need a root certificate who's security is above reproach

- **Why?**

ROOTPublicKey,(ROOTPublicKey[MD5])ROOT PrivateKey

CAPublicKey,(CAPublicKey[MD5])CAPrivateKey,(CAPublicKey[MD5])ROOTPrivateKey

UserPublicKey,(UserPublicKey[MD5])UserPrivateKey,(UserPublicKey[MD5])CAPrivateKey

12

Carnegie Mellon University
CERT Coordination Center

Distribution of Hierarchical Public Keys

The root public key must be widely distributed in a variety of paths to everyone in the hierarchy

- **Why multiple paths?**
- **What is the primary vulnerability here?**

If the root key is secure, the system can be consistent

Root key is used to sign all revocation certificates for Cas

Root servers *do not need to sign keys lower in the hierarchy*

- **Why not?**

13

Carnegie Mellon University
CERT Coordination Center

One versus Multiple Hierarchies

What are the problems with a single root server for all PKI systems?

If you want to trust users across hierarchies, you need *cross-realm certification*

Combines Web-of-Trust with Hierarchical PKI

Means that some root or CA public key is signed by one in the other hierarchy

14

Carnegie Mellon University
CERT Coordination Center

Problems in cross-realm certification

- Naming
- Different policies for inclusion in the hierarchy
- Different uses of keys
- Compatibly of algorithms and key records

15

Carnegie Mellon University
CERT Coordination Center

What does this have to do with operating system security architectures?

- Application-level architecture
- Trust of users within the operating system
- Basic tool for linking users with processes
- Kerberos and related systems make use of these concepts to implement OS trust

16