Using Cryptography
CS 239
Computer Security
February 7, 2007

Outline

- Digital signatures
- Digital hashes
- Legal and political issues in crypto
  - Key recovery systems

Digital Signature Algorithms

- In some cases, secrecy isn't required
- But authentication is
- The data must be guaranteed to be that which was originally sent
- Especially important for data that is long-lived

Desirable Properties of Digital Signatures

- Unforgeable
- Verifiable
- Non-repudiable
- Cheap to compute and verify
- Non-reusable
- No reliance on trusted authority
- Signed document is unchangeable

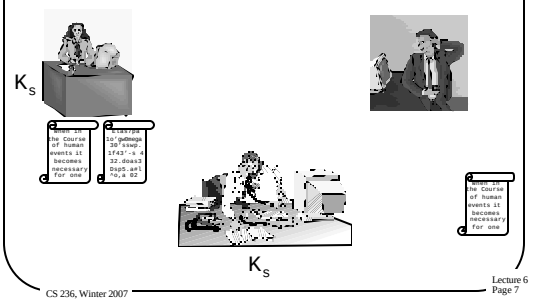
Encryption and Digital Signatures

- Digital signature methods are based on encryption
- The basic act of having performed encryption can be used as a signature
  - If only I know *K*, then *C=E(P,K)* is a signature by me
  - But how to check it?

Signatures With Shared Key Encryption

- Requires a trusted third party
- Signer encrypts document with secret key shared with third party
- Receiver checks validity of signature by consulting with trusted third party
- Third party required so receiver can't forge the signature

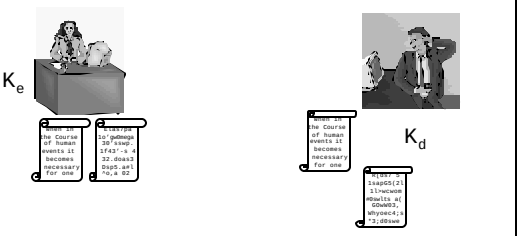
For Example,



Signatures With Public Key Cryptography

- Signer encrypts document with his private key
- Receiver checks validity by decrypting with signer's public key
- Only signer has the private key
  - So no trusted third party required
- But receiver must be certain that he has the right public key

For Example,



Problems With Simple Encryption Approach

- Computationally expensive
  - Especially with public key approach
- Document is encrypted
  - Must be decrypted for use
  - If in regular use, must store encrypted and decrypted versions

Secure Hash Algorithms

- A method of protecting data from modification
- Doesn't actually prevent modification
- But gives strong evidence that modification did or didn't occur
- Typically used with digital signatures

Idea Behind Secure Hashes

- Apply a one-way cryptographic function to data in question
- Producing a much shorter result
- Attach the cryptographic hash to the data before sending
- When necessary, repeat the function on the data and compare to the hash value

Secure Hash Algorithm (SHA)

- Endorsed by NIST
- Reduces input data of up to 2^{64} bits to 160 bit digest
- Doesn't require secret key
- Broken in 2005

What Does "Broken" Mean for SHA-1?

- A crypto hash matches a digest to a document
- It's bad if two documents match the same digest
- It's very bad if you can easily find a second document with a matching hash
- The crypto break finds matching hashes in 2^{63} operations

How Bad Is That?

- We can do things in 2^{63} operations
  - Though it's not trivial
- But the second "document" might be junk
- So relevant if that is a reasonable attack
- NIST isn't panicking
  - But is recommending phasing out SHA-1 by 2010
  - NIST just announced a competition for a new secure hash standard

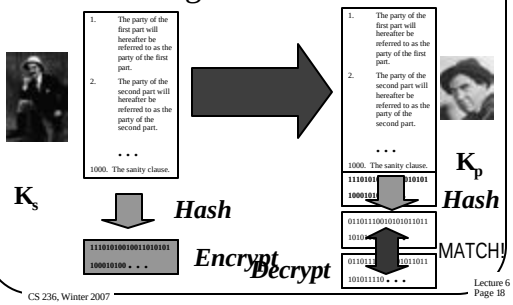
Use of Cryptographic Hashes

- Must assume opponent also has hashing function
- And it doesn't use secret key
- So opponent can substitute a different message with a different hash
- How to prevent this?
- And what (if anything) would secure hashes actually be useful for?

Hashing and Signatures

- Use a digital signature algorithm to sign the hash
- But why not just sign the whole message, instead?
- Computing the hash and signing it may be faster than signing the document
- Receiver need only store document plus hash

Checking a Document With a Signed Hash



The Birthday Attack

- How many people must be in a room for the chances to be greater than even that two of them share a birthday?
- Answer is 23
- The same principle can be used to attack hash algorithms

Using the Birthday Attack on Hashes

- For a given document, find a different document that has the effect you want
- Trivially alter the second document so that it hashes to the same value as the target document
  - Using an exhaustive attack

How Hard Is the Birthday Attack?

- Depends on the length of the hash
  - And the quality of the hashing algorithm
- Essentially, looking for hashing collisions
- So long hashes are good
  - SHA produces 2^{80} random hashes
  - But 2005 attack finds collisions in 2^{63} operations
  - Not for chosen plaintext, however

Legal and Political Issues in Cryptography

- Cryptography is meant to help keep secrets
- But should all secrets be kept?
- Many legal and moral issues

Societal Implications of Cryptography

- Criminals can conceal communications from the police
- Citizens can conceal taxable income from the government
- Terrorists can conceal their activities from governments trying to stop them

Problems With Controlling Cryptography

- Essentially, it's mostly algorithms
- If you know the algorithm, you can have a working copy easily
- At which point, you can conceal your secrets from anybody
  - To the strength the algorithm provides

Governmental Responses to Cryptography

- They vary widely
- Some nations require government approval to use cryptography
- Some nations have no laws governing cryptography at all
- The US laws less restrictive than they used to be

The US Government Position on Cryptography

- All forms of cryptography are legal to use in the US
- **BUT**
  - Some minor restrictions on exporting cryptography to other countries
- The NSA used to try to keep a lid on cryptographic research

US Restrictions on Cryptographic Exports

- Rules changed in 2000
- Greatly liberalizing cryptographic exports
- Almost all cryptography is exportable
- Exception is for government use by a handful of countries
  - Those the US government currently doesn't like

Cryptographic Source Code and Free Speech

- US government took Phil Zimmermann to court over PGP
- Court ruled that he had a free-speech right to publish PGP source
- Eventually, appeals courts also found in favor of Zimmermann

Other Nations and Cryptography

- Generally, most nations have few or no restrictions on cryptography
- A group of treaty signatories have export restrictions similar to US's
- Some nations have stronger restrictions
  - China, Russia, Vietnam, a few others
- A few have laws on domestic use of crypto
  - E.g., Australia, UK, India have laws that demand decryption with court order

Key Recovery Cryptosystems

- An attempt to balance:
  - Legitimate societal security needs
    - Which require strong encryption
  - And legitimate governmental and law enforcement needs
    - Which require access to data
- How can you have strong encryption and still satisfy governments?

Idea Behind Key Recovery

- Use encryption algorithms that are highly secure against cryptanalysis
- But with mechanisms that allow legitimate law enforcement agency to:
  - Obtain any key with sufficient legal authority
  - Very, very quickly
  - Without the owner knowing

Proper Use of Data Recovery Methods

- All encrypted transmissions (or saved data) must have key recovery methods applied
- Basically, the user must cooperate
  - Or his encryption system must force him to cooperate
  - Which implies everyone must use this form of cryptosystem

Methods to Implement Key Recovery

- Key registry method
  - Register all keys before use
- Data field recovery method
  - Basically, keep key in specially encrypted form in each message
  - With special mechanisms to get key out of the message

Problems With Key Recovery Systems

- Requires trusted infrastructures
- Requires cooperation (forced or voluntary) of all users
- Requires more trust in authorities than many people have
- International issues
- Performance and/or security problems with actual algorithms

The Current Status of Key Recovery Systems

- Pretty much dead (for widespread use)
- US tried to convince everyone to use them
  - Skipjack algorithm, Clipper chip
- Very few agreed
- US is moving on to other approaches to dealing with cryptography
- Some businesses run key recovery internally
  - More to avoid losing important data when keys lost than for any other reason