COMP 3704 Computer Security

Christian Grothoff
christian@grothoff.org

http://grothoff.org/christian/

Protocols

- "A **protocol** is a series of steps, involving two or more parties, designed to accomplish a task."
- Everyone involved must know the steps in advance and agree to follow it.
- The protocol must be complete and unambiguous.
- For cryptographic protocols, it should not be possible to do more or learn more than what is specified in the protocol.

Dramatis Personae[1]

- Alice, Bob, Carol and Dave
- Eve – Eavesdropper
- Mallory – Malicious active attacker
- Trent – Trusted arbitrator
- Walter – Warden
- Peggy – Prover
- Victor – Verifier

[1] More at `http://en.wikipedia.org/wiki/Alice_and_Bob`

Efficiency

- Number of steps in protocol
- Size of messages
- Conflict resolution cost:
  1. Involvement of trusted party (arbitrated protocols)
  2. Resolution by trusted party on dispute (adjudicated protocols)
  3. Self-enforcing protocols

Attack Personae

- Eavesdroppers
- Passive cheaters
- Active cheaters
- Real-world adversaries – Mallory

Example: Symmetric Cryptography

1. Alice and Bob agree on a cryptosystem
2. Alice and Bob agree on a key
3. Alice encrypts plaintext with key
4. Alice sends ciphertext to Bob
5. Bob decrypts ciphertext and reads it

One-Way (hash) Functions

- Easy to compute $f(x)$, hard to compute $f^{-1}(y)$
- Trapdoor one-way hash functions: hard to compute f^{-1} without the secret
- Good hash functions are collision-free: it is hard to generate two pre-images with the same hash value

Alternative names

- Contraction function
- Message digest
- Fingerprint
- Cryptographic checksum
- Message integrity check (MIC)
- Manipulation detection code (MDC)
- Message authentication code (MAC) $\equiv$ hash + key

Public-key Cryptography

The mathematical primitive is often similar to trapdoor one-way hash functions.

Canonical use:

1. Alice and Bob agree on a public-key cryptosystem.
2. Bob sends Alice his public key.
3. Alice encrypts her message using Bob's public key.
4. Alice sends the ciphertext to Bob.
5. Bob decrypts Alice's message using his private key.

Hybrid Cryptosystems

- Use public key cipher for key exchange (session key)
- Use symmetric cipher for data exchange
- Use hashing/MAC to verify data integrity

Signatures

Autenticic: The signer deliberately signed the document.

Unforgeable:
Nobody but the signer signed the document.

not reusable:
The signature cannot be moved to another document.

Unalterable:
The document cannot be changed after signing.

not repudiatable:
The signer cannot later claim not to have signed it.

Public-key Signatures

1. Alice encrypts the document with her private key.
2. Alice sends the signed document to Bob.
3. Bob decrypts the document with Alice's public key.

In practice, Alice does not encrypt the document, but a hash of the document. This is implied when we use the notations $S_K(M)$ and $V_K(M)$ for signing and verifying.

Attacks

- Repudiation by intentional key compromise
- Replay attacks
- Signing is decrypting!

Questions

?

Problem

Alice has an item x, and Bob has a set of five distinct items $y_1, \ldots, y_5$. Design a protocol through which Alice (but not Bob) finds out whether her x equals any of Bob's five items; Alice should not find out anything other than the answer ("Yes" or "No") to the above question, and Bob should not know that answer. Your solution must always be correct, not just with high probability.