18739A: Foundations of Security and Privacy

Just Fast Keying (JFK) Protocol

Anupam Datta

CMU

Fall 2007-08

Outline

- "Rational derivation" of the JFK protocol
  - Combine known techniques for shared secret creation, authentication, identity and anti-DoS protection
    - [Datta, Mitchell, Pavlovic Tech report 2002]
- Just Fast Keying (JFK) protocol
  - State-of-the-art key establishment protocol
    - [Aiello, Bellovin, Blaze, Canetti, Ioannidis, Keromytis, Reingold CCS 2002]
- Modeling JFK in applied pi calculus
  - Specification of security properties as equivalences
    - [Abadi,Fournet POPL 2001]
    - [Abadi, Blanchet, Fournet ESOP 2004]

Later lecture

Design Objectives for Key Exchange

- Shared secret
  - Create and agree on a secret which is known only to protocol participants
- Authentication
  - Participants need to verify each other's identity
- Identity protection
  - Eavesdropper should not be able to infer participants' identities by observing protocol execution
- Protection against denial of service
  - Malicious participant should not be able to exploit the protocol to cause the other party to waste resources

Ingredient 1: Diffie-Hellman

$$A \rightarrow B: \quad g^a$$
$$B \rightarrow A: \quad g^b$$

- Shared secret: g^{ab}
  - Diffie-Hellman guarantees perfect forward secrecy
- Authentication
- Identity protection
- DoS protection

Ingredient 2: Challenge-Response

$$A \rightarrow B: \; m, A$$
$$B \rightarrow A: \; n, sig_B\{m, n, A\}$$
$$A \rightarrow B: \; sig_A\{m, n, B\}$$

- Shared secret
- Authentication
  - A receives his own number m signed by B's private key and deduces that B is on the other end; similar for B
- Identity protection
- DoS protection

DH + Challenge-Response

ISO 9798-3 protocol:

$A \rightarrow B: \; g^a, A$

$B \rightarrow A: \; g^b, sig_B\{g^a, g^b, A\}$

$A \rightarrow B: \; sig_A\{g^a, g^b, B\}$

$m := g^a$

$n := g^b$

- Shared secret: g^{ab}
- Authentication
- Identity protection
- DoS protection

Ingredient 3: Encryption

Encrypt signatures to protect identities:

$$A \rightarrow B: \quad g^a, A$$
$$B \rightarrow A: \quad g^b, E_K\{sig_B\{g^a, g^b, A\}\}$$
$$A \rightarrow B: \quad E_K\{sig_A\{g^a, g^b, B\}\}$$

- Shared secret: g^{ab}
- Authentication
- Identity protection (for responder only!)
- DoS protection

Refresher: Anti-DoS Cookie

- Typical protocol:
  - Client sends request (message #1) to server
  - Server sets up connection, responds with message #2
  - Client may complete session or not (potential DoS)
- Cookie version:
  - Client sends request to server
  - Server sends hashed connection data back
    - Send message #2 later, after client confirms
  - Client confirms by returning hashed data
  - Need extra step to send postponed message

Ingredient 4: Anti-DoS Cookie

"Almost-JFK" protocol:

$A \rightarrow B$: g^a, A

$B \rightarrow A$: g^b, $hash_{Kb}\{g^b, g^a\}$

$A \rightarrow B$: g^a, g^b, $hash_{Kb}\{g^b, g^a\}$
$E_K\{sig_A\{g^a, g^b, B\}\}$

$B \rightarrow A$: g^b, $E_K\{sig_B\{g^a, g^b, A\}\}$

Doesn't quite work: B must remember his DH exponential b for every connection

- Shared secret: g^{ab}
- Authentication
- Identity protection
- DoS protection?

Additional Features of JFK

- Keep g^a, g^b values medium-term, use (g^a,nonce)
  - Use same Diffie-Hellman value for every connection (helps against DoS), update every 10 minutes or so
  - Nonce guarantees freshness
  - More efficient, because computing g^a, g^b, g^{ab} is costly
- Two variants: JFKr and JFKi
  - JFKr protects identity of responder against active attacks and of initiator against passive attacks
  - JFKi protects only initiator's identity from active attack
- Responder may keep an authorization list
  - May reject connection after learning initiator's identity

JFKr Protocol [Aiello et al.]