18733: Applied Cryptography

Just Fast Keying Protocol

Anupam Datta
CMU
JFK

- Just Fast Keying (JFK) protocol
  - State-of-the-art key establishment protocol
    - [Aiello et al. 2002]
  - Informed design of IKEv2: the IPSec key exchange protocol

- Derivation of the JFK protocol
  - Combine known techniques for shared secret creation, authentication, identity and anti-DoS protection
    - [Datta et al. 2002]
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 → B: \( g^a \)

B → A: \( g^b \)

- Shared secret: \( g^{ab} \)
- Authentication
- Identity protection
- DoS protection
Ingredient 2: Challenge-Response

A → B: m, A
B → A: n, \text{sig}_B\{m, n, A\}
A → B: \text{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:

\[ \begin{align*}
A \rightarrow B & : \ g^a, A \\
B \rightarrow A & : \ g^b, \text{sig}_B\{g^a, g^b, A\} \\
A \rightarrow B & : \ \text{sig}_A\{g^a, g^b, B\}
\end{align*} \]

- Shared secret: \(g^{ab}\)
- Authentication
- Identity protection
- DoS protection
Ingredient 3: Encryption

Encrypt signatures to protect identities:

A → B: $g^a, A$

B → A: $g^b, E_K\{\text{sig}_B\{g^a, g^b, A\}\}$

A → B: $E_K\{\text{sig}_A\{g^a, g^b, B\}\}$

- 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
"Almost-JFK" protocol:

A → B: $g^a, A$
B → A: $g^b, \text{hash}_{K_b}\{g^b, g^a\}$
A → B: $g^a, g^b, \text{hash}_{K_b}\{g^b, g^a\}$
\[ E_K\{\text{sig}_A\{g^a, g^b, B\}\} \]
B → A: $g^b, E_K\{\text{sig}_B\{g^a, g^b, A\}\}$

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

Doesn’t quite work: B must remember his DH exponential b for every connection
Anti-DoS

- 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
Identity protection

- 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
JFKr Protocol

If initiator knows group g in advance

Same d_r for every connection

DH group

Derive a set of keys from shared secret and nonces

"hint" to responder which identity to use

Check integrity before decrypting

Real identity of the responder