A Cryptographically Secure Decentralised Communication Protocol

James Nicholson
University of Cambridge

Desirable Properties

- Preventing eavesdropping/modification:
  - Strong, end-to-end encryption.
  - Message authentication.
- Preventing network failure/censorship:
  - Decentralisation.
- Allowing remote trust/verification:
  - Public-key cryptography.

Cryptography

- Requirements:
  - Strong, fast, end-to-end encryption.
  - Public-key cryptography for trust.
- Problem:
  - Public key cryptography is very slow.
- Solution:
  - A hybrid cryptosystem.

Hybrid Cryptosystem

- Trust/identity verification; key exchange:
  - Needham-Schroeder-Lowe public-key protocol.
- Communication:
  - AES symmetric-key encryption.
- Message authentication/integrity:
  - Message Authentication Code (MAC).

Needham-Schroeder-Lowe

$A \rightarrow B : \{NA, A\}_{KB}$

$B \rightarrow A : \{NA, NB, B\}_{KA}$

$A \rightarrow B : \{NB\}_{KB}$

- Simple.
- Formally verified [Paulson 1998].

Keys

- Need keys for:
  - AES (communication encryption).
  - MAC (message authentication).
    - Specifically, HMAC: keyed-Hash MAC.
- Both keys are random, 128-bits in length.
- Need to be exchanged at start, i.e. during NSL.

NSL Nonces

- Needham-Schroeder-Lowe uses nonces to prevent replay attacks (NA and NB above).
- Nonce is random, 256-bits in length.
- Thus, the AES and MAC keys can be exchanged as the nonce.

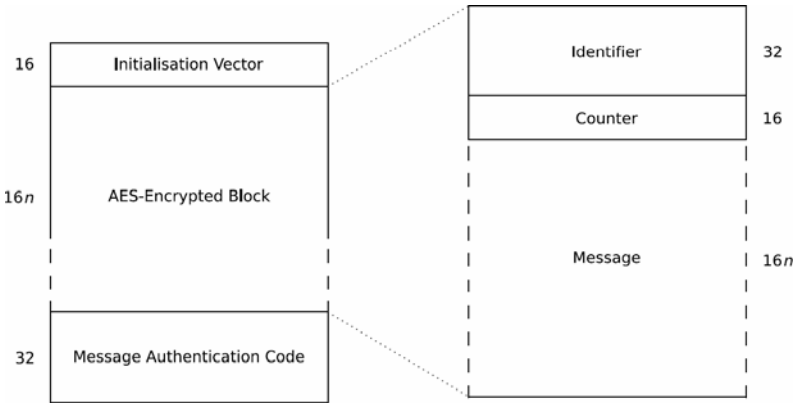
Attacks

- We now have
  - End-to-end message encryption.
  - Message authentication.
- However, replay attacks are still possible.

Preventing Replay Attacks

- Firstly, I add a counter to each message:
  - A message whose count not strictly greater than the previous is rejected.
  - A message whose counter is N greater than the previous means we have missed N-1 messages.
- Secondly, I add an identifier for the sender:
  - This prevents a message being replayed to the sender.

Overview: Packet Structure



Decentralisation

- Implemented using a Distributed Hash Table (DHT): Chord.
- The DHT maps a user's identifier (a fingerprint of their public key) to their public key and location (i.e. IP, port).

Decentralisation Issues

- The downside of this implementation is that everyone needs to be able to write to the DHT.
  - This can lead to denial of service.
  - Partial solution: key/value pairs can be widely replicated.
  - Still a problem.

Cryptographic Issues

- Using NSL for key exchange means that if *either* user's private key is compromised, *both* sides of the conversation can be decrypted.
  - This is a Bad Thing.
  - Solution: Diffie-Hellman for key exchange.
    - NSL can then be performed within this for trust, etc.