EMC² where information lives®

Security Association Creation

David L. Black

November 2006

EMC² where information lives

The Big Picture: Three Proposals

- Framework: 06-369 specifies Security Associations (SAs)
  - Connect key generation to key usage (and identify keys)
  - Leverage one key exchange to generate multiple keys
  - Need to revise 06-369 (coming up later in these slides)
- SA Creation: 06-449 specifies SA creation
  - Algorithm selection, key exchange and authentication
  - 06-449r0 is not finished yet (see Editor's Notes in proposal)
    - Revised version will also specify SA destruction
- SA Usage: 06-225 successor will specify SSC-3 keying
  - Use SA to key ESP (IPsec) instead of AES key wrap
  - SSC-3 result: IPsec from start to finish (should be ok with NIST)
- 06-103 is dead (next slide ...)

Creating a Security Association: 06-449 replaces 06-103

- 06-103 is dead. Why?
  1. Need flexible selection of crypto algorithms
     - Too many choices for 06-103's "just use this" announcements
  2. Need Authentication (including defense vs. man-in-the-middle)
     - Adds significant complexity to 06-103's unauthenticated key exchange
  3. Desire FIPS 140-n eligibility
     - Strongly favors reuse of existing security technology/protocols
- 06-449 proposes IKEv2-SCSI, a cut-down version of IKEv2
  1. Device capabilities replace and simplify IKEv2 negotiation
  2. Authentication via shared secret and public key digital signature
  3. IKEv2 (IPsec) techniques should be acceptable to NIST
- 06-449 removes a lot of IKEv2 functionality
  - Examples: Child SAs, Traffic Selectors, NAT traversal, Compression

EMC² where information lives®

SA creation: 3 phases

1. Application client reads device security capabilities
   - Different SCSI Security Protocol value from. phases 2 and 3
2. Key exchange and security algorithm selection
   - Diffie-Hellman exchange with nonces (allows exponential reuse)
   - DH group/algorithm selectable, based on phase 1
     - IETF has specifications for elliptic curve (ECC) Diffie-Hellman
3. Authentication
   - Required: Shared Secret (used by iSCSI and Fibre Channel)
     - Same type of authentication secret as iSCSI and Fibre Channel
   - Optional: Public key and certificates (RSA or DSS)
     - IETF has an approved specification for ECDSA authentication
   - Proposal allows omission of authentication by mutual agreement
   - Additional authentication types can be added later
     - Example: Kerberos (makes sense if one thinks about iSCSI)

EMC² where information lives

06-449 Open Issues

- Command Sequencing – what if app. client gives up?
  - How long does device server have to remember in-progress state?
- Minimum parameter data size requirement?
  - Can be longer than typical IP datagrams. Think about certificates.
- Delete vs. timing out SAs? Probably need Delete
- Identity types (what's a name?)
  - Proposal includes certificate primary identities and opaque identities
  - Opaque identity - integration with directories and the like
  - SCSI-specific identities appear to be problematic
- Certificate formats – restrictions needed (cf. RFC 4718)
  - Should SCSI support HTTP certificate retrieval?
- Security review of progress indications and field pointers

EMC² where information lives®

06-449: Key Derivation Function for IKEv2

- Key Derivation Function (KDF)
  - Multiple keys derived from one key + additional inputs (e.g., nonces)
- IKEv2 authentication uses a KDF – may need six(!) keys
  - prf+() KDF family, based on selected pseudo-random function (prf)
- NIST KDF family specified in NIST publication 800-56A
- Problem: NIST KDF family requires identities as inputs
  - IKEv2 authentication KDF used before identities are known
  - Modifying the NIST KDF (e.g., fudging identity inputs) is a bad idea
  - Modifying IKEv2 authentication is an even worse idea.
- Result: Can't use NIST KDF for IKEv2 authentication
  - IKEv2 authentication needs to use IKEv2 prf+() KDF family

EMC² where information lives

06-369 Issue: Key Derivation Functions for SAs

- What about KDFs in Security Associations?
  - 06-369 currently uses NIST KDFs
  - Will need to add identities to continue using NIST KDFs
- Alternative: Use IKEv2 KDFs in Security Associations
  - Using the same KDF as IKEv2 authentication is simpler
  - For SSC-3, result is full IPsec, should be acceptable to NIST
- NIST and IETF: IETF will not use NIST KDF for TLS
  - NIST considering what to do instead of insisting on its KDFs
  - Issues appear to involve identity binding to derived keys


EMC²
where information lives®