EMC²
where information lives®

SCSI Communications Security

David L. Black

Senior Technologist, EMC

Communications Security Overview

- Two types of security goals
  1. Communications Security = protecting communications
  2. System Security = protecting systems
     - Including systems that engage in communications
- Pursue these security goals in order
  - First, secure the communications
  - Then provide secure access to systems (over secured channels)
- First SCSI communications security problem is here
  - SSC-3 transfer of encryption keys to tape devices
  - Need an alternative to transfer in the clear
- Two parts to this presentation
  1. Communications Security Background
  2. Communications Security Framework – Security Associations

EMC² where information lives

Communications Security Assumptions (06-388r1)

1. Assume end systems involved in communication are ok
   - Ok = Not under attacker's control
   - SCSI end systems = Application Clients and Device Servers
2. Assume communication channels are not ok
   - Not Ok = Attacker can read/write/change any communication
   - SCSI communication channels = Service delivery subsystem
     - Includes transport and intermediate SCSI-to-SCSI router/gateway/bridge
3. Assume attacker does not know secret keys
   - And cannot learn or discover them
   - Communication Security is based on secrets, especially keys

Communications Security Threats (06-388r1)

1. Passive Attacks = Attacker just listens
   - Observe data that should not have been sent in clear
   - Obtain raw material for off-line analysis (e.g., dictionary attack)
2. Active Attacks = Attacker does something
   - Address or identity spoofing
   - Replay of previous communication (e.g., install old key)
   - Insertion, deletion, and modification of communications
   - Man-in-the-middle – communication is *via* the attacker

- All of these attacks can be countered
  - But "No Free Lunch" principle applies
- See IETF RFC 3552 for a more extensive discussion

Communications Security Countermeasures

- Confidentiality – keep information secret
  - Implementation: Encryption
- Cryptographic Integrity – Prevent deliberate tampering
  - Implementation – Keyed secure hash
  - Anti-Replay is a specific form of Cryptographic Integrity
- Authentication – Know (prove) who is talking/listening
  - Implementation – Mechanism depends on form of proof
  - Form of proof strongly affects manageability
- Authorization – Limit what can be done
  - For communications security this means reachability (e.g., zoning)
- Multiple security services often appropriate/necessary
  - SSC-3 secure tape drive keying – confidentiality, cryptographic integrity (including anti-replay) and authentication

SCSI Security Considerations

- Communications Security already in some SCSI transports
  - Transport security not always used or usable
  - Multi-transport interactions (e.g., SCSI to SCSI router)
- SCSI environments often have limited connectivity
  - Physical and logical restrictions
  - LUN mapping and masking
  - Transport zoning
- SCSI security in light of transport security/connectivity
  - Reduces reliance on authentication for SCSI
    - Not authenticating at SCSI level can be acceptable
  - Ability to send command to device server can be implicit "proof"

Security Design Practices

- Reuse existing techniques and practices
  - Leverage security community expertise and interest
  - Ideally – make security design “someone else’s problem”
- Specify security services, allow usage to be optional
  - Security is available if needed, customer can decide
- Amortize expensive security operations
  - Support SCSI usage in embedded systems

EMC² where information lives®

SCSI Security Associations

SCSI should define Security Associations

- Communications security framework
  - First usage example - SSC-3 encryption key transfer to tape drives
    - There will be more
  - Multiple key consuming security services are needed
    - Encryption and Cryptographic Integrity
  - Multiple key generating security services are likely
    - Key Exchange and Authentication
    - Key Exchange = create shared secret key (authenticated or unauthenticated)
- Framework can lay groundwork for implementations
  - Coordination is essential to security and usability
    - Mix/match key generation and consumption
  - Common key derivation to support multiple security services
  - Common anti-replay design
  - Support for amortization of expensive crypto operations
    - E.g., public key (asymmetric cryptography) operations and key exchange

What's a Security Association (SA)? (06-369r3)

- Purpose: Security Service Coordination
  - Concept from IETF IP Security (IPsec) architecture
- Key identification (in order to use them) and associated security ops
  - SAI (Security Association Index)
- Prevent Replay Attacks – SQN sequence numbers
- Amortization of expensive operations – KEY(s) and random NONCEs
  - Key exchange/generation can be **very** expensive (computationally)
  - Generation of multiple symmetric keys is the major advantage
  - Nonces also enable faster rekey based on original key exchange
- Decouple key exchange/generation from key usage – KEY_SEED
  - Allow multiple mechanisms for key generation and usage
  - Use any key generation mechanism with any key usage mechanism
- Produce multiple keys (to apply multiple security services)
  - KDF (Key Derivation Function) and resulting KEY(s)

How does a Security Association Help?

- Reduces integration burden for new mechanisms
  - New key exchange mechanism only has to generate SA
  - New key usage mechanism only has to consume SA
  - Avoids m x n integration problems and inconsistencies
    - But more complex if m=n=1
  - Helps others design key exchange mechanisms for SCSI
- Standardizes derivation of multiple keys
  - From single result of key exchange
  - Easy to get wrong in subtle ways
- Standardizes amortization of expensive crypto
  - Generation of symmetric keys from key exchange
  - Rekey based on random nonces only

What if there are no Security Associations?

- Multiple key exchange mechanisms are possible/likely:
  - Unauthenticated Diffie-Hellman (Mod-p and/or Elliptic Curve)
  - Unauthenticated or self-authenticated public key (RSA)
  - PKI (certificate) authentication and key exchange
  - Authenticate via pre-existing key (shared secret)
  - Use TCG protocol(s)
- Key exchange mechanisms intersect at point of usage
  - All key exchanges can work without a Security Association
  - SSC-3: Each new key exchange needs a new key format
    - Even if the actual “bits on the wire” format (e.g., ESP) is the same
- Burden on future designs (beyond SSC-3 encryption keys)
  - Plug-in interface for multiple key exchange mechanisms
  - Redesign key derivation (subtle differences probable)
  - Redesign or lose ability to amortize expensive crypto

Generating a Security Association: How to Authenticate?

- Don't – unauthenticated security
  - Explicit/implicit authentication by other means – see slide 6
- Strong shared secret (key) – Don't send in clear!!
  - Requires pre-installation of secrets
  - Impersonation: Anyone who has X's secret can impersonate X
- Password (weak shared secret) – Don't send in clear!!
  - Requires pre-installation of password verifier (avoids impersonation)
  - Need serious security protocol to avoid dictionary attack or generate keys
- Public key (signature based on public/private key pair)
  - Naked public key for authentication is frowned upon
  - Public key authentication can lead to certificate/PKI rathole/adventure
- Kerberos (ticket)
  - Good for user authentication, not so good for machine authentication
- Other mechanisms exist (e.g., h/w token to create strong "password")