secureendpoints

Hardware Security Modules and Kerberos

Asanka Herath

Secure Endpoints Inc.

Advantages of integrating HSMs into the Kerberos infrastructure

WHY HSMS?

secureendpoints

Why HSMs?

- Compromise containment
- Reliable Auditing
- Performance
- Compliance (FIPS 140)

Changes to key management when integrating an HSM

KEY MANAGEMENT WITH AN HSM

secureendpoints

For example…

secureendpoints

nCipher nShield

secureendpoints

NCIPHER SECURITY WORLDS

Properties of a Security World

- Key management
  - Managed key usage
- Key encapsulation
  - Keys generated and live inside the security world and (ideally) never leave
- Secure code execution
  - Secure Execution Environment (SEE)

secureendpoints

A Security World Consists of …

- Hardware modules
- Administrator card set
- Operator card sets and/or softcards
- Encrypted key data or certificate data

secureendpoints

The Security World Key

- Contained in …
  - The administrator card set
- Protects …
  - Key recovery information

Module Key

- Contained in …
  - The hardware module
- Protects …
  - Application keys

Operator Keys

- Contained in …
  - Operator card sets
- Protects …
  - Application keys

secureendpoints

Application Keys

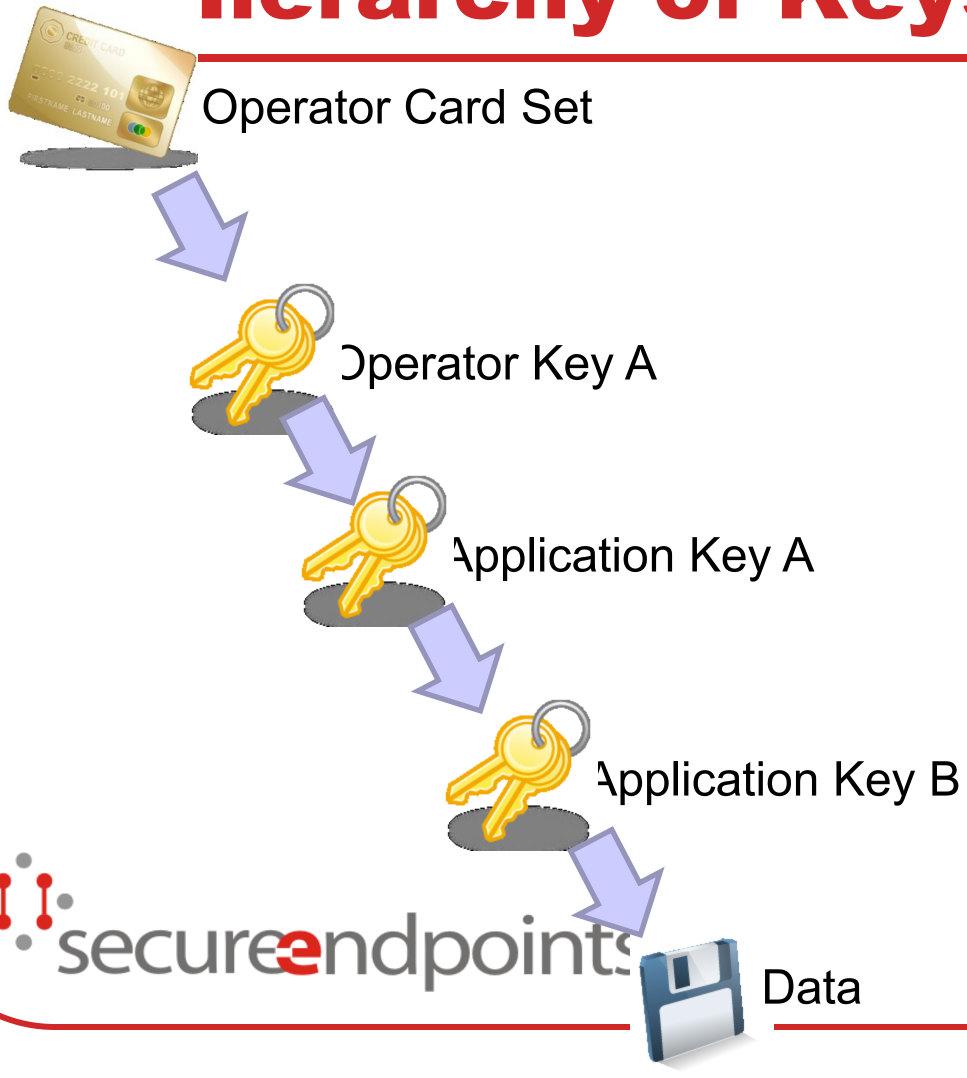
- Contained in …
  - Key blobs
- Protects …
  - Other applications keys
  - Data

secureendpoints

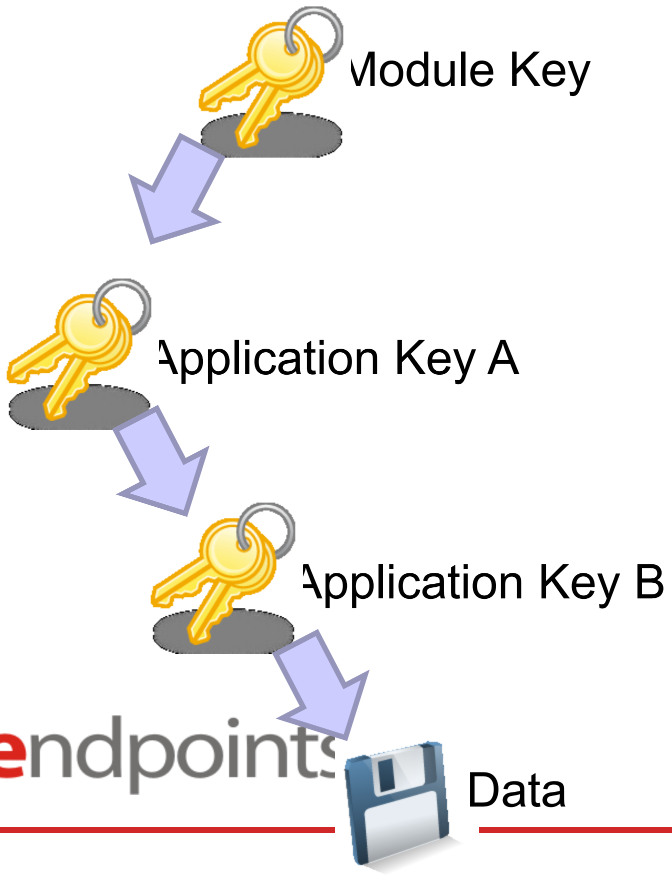
A Security World Consists of …

- Keys that protect other keys

A Hierarchy of Keys



A Hierarchy of Keys



Privilege Separation

- Ability to use a key gives the ability to use child keys
- But not vice-versa
- Facilitates privilege separation

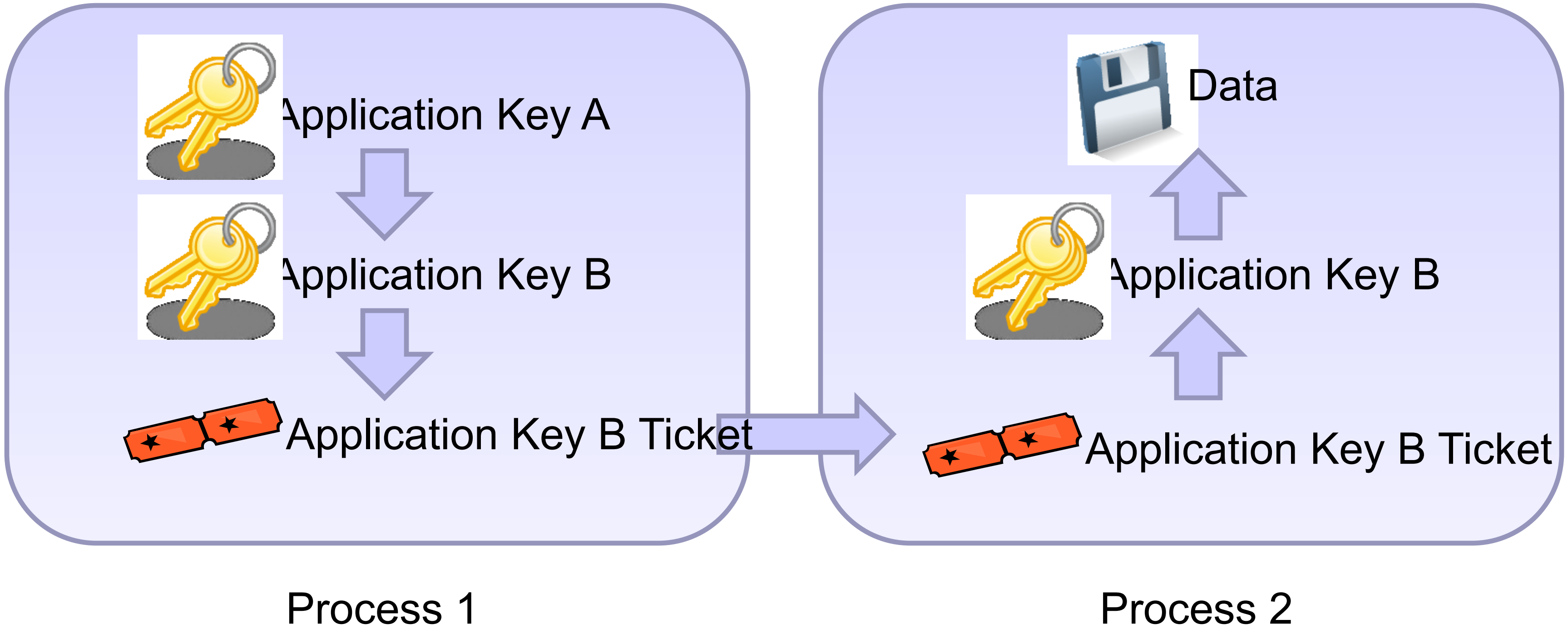
secureendpoints

Key Tickets

- Can be issued for any loaded key
- Can be redeemed for the use of a specific key
  - Necessary for delegating use of keys across application boundaries
- Transient
  - Per session

secureendpoints

Working With Key Tickets



Integrating HSMs with Kerberos infrastructure

HEIMDAL

Key Usage Model in Heimdal

- Application has access to plaintext key

secureendpoints

New Key Usage Model

- Separate key usage from key material
  - An application doesn’t need the plaintext key in order to use it
- Ability to specify a key without knowing the plaintext key
  - Key tickets
  - Key blobs
  - Key identifiers
- Ability to load and unload keys
  - Using already loaded keys

What's involved in integrating the use of HSMs into code

KERBEROS API AND LIBRARY

secureendpoints

Key Encoding

- Current …
  - Assumes knowledge of the physical key
    - Key length is fixed
    - No need to prepare/clean up keys. Stateless. (exceptions: derived keys, key schedules)
- Need …
  - Backwards compatibility
  - Support key references
    - Key tokens
    - Key blobs
    - Key identifiers

Key Blob

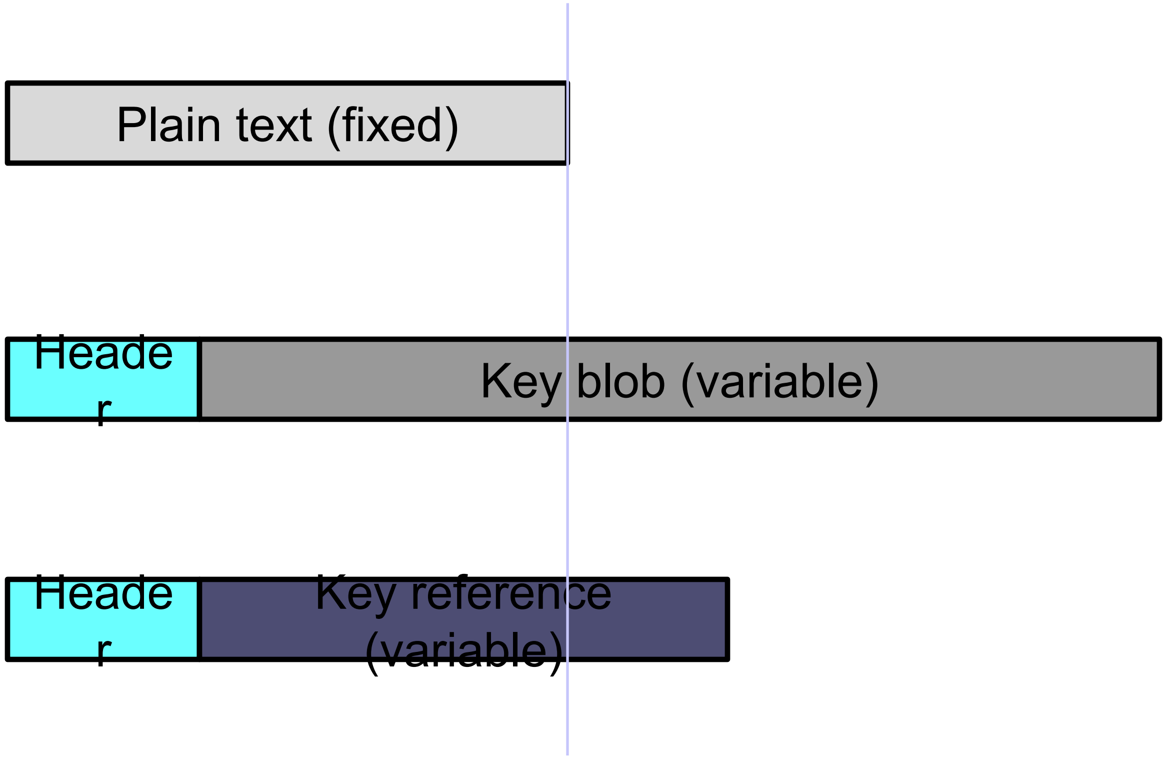
- Contains (encrypted with parent key)
  - Key
  - Access control list
  - Usage constraints
    - Time limits
    - Use limits
  - Issuance certificate
    - Security world and module information
  - Timestamps

Key Types and Encryption Types

- A new “Key Type”?
- A new “Encryption Type”?
- Must distinguish between a plaintext key and a key reference

secureendpoints

Something like ...



secureendpoints

Key “Lifetimes”

- Loading a key requires a parent/authorization key to be already loaded
- “Lifetime” refers to the time between loading and unloading a key

secureendpoints

Connection to Hardware Devices

- Connections must be established first
  - Can be somewhat expensive
  - Lazy connections
- Connections should be avoided if unnecessary
  - Specially for client libraries where we may or may not have access to the security world and access to security world is not necessary

secureendpoints

Privilege Separated Processes

- Break into privilege separates processes
- Move privileged code into SEE

asanka@secure-endpoints.com

Q&S