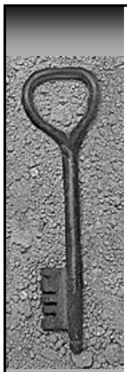
# Key Management Lifecycle



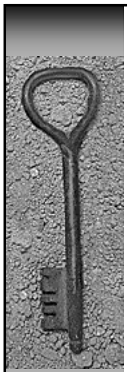
# Key Management Lifecycle

Cryptographic key management encompasses the entire lifecycle of cryptographic keys and other keying material. Basic key management guidance is provided in [SP800-21].

A single item of keying material (e.g., a key) has several states during its life, though some of these states may, in fact, be very short:

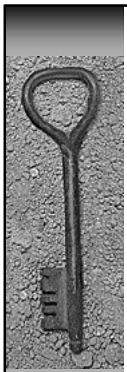
- **Pre-operational:** The keying material is not yet available for normal cryptographic operations.
- **Operational:** The keying material is available and in normal use.
- **Post-operational:** The keying material is no longer in normal use, but access to the material is possible.
- **Obsolete/destroyed:** The keying material is no longer available. All records of its existence may have been deleted.

The next viewgraph identifies the subsections that discuss various stages of key management for a given entity.



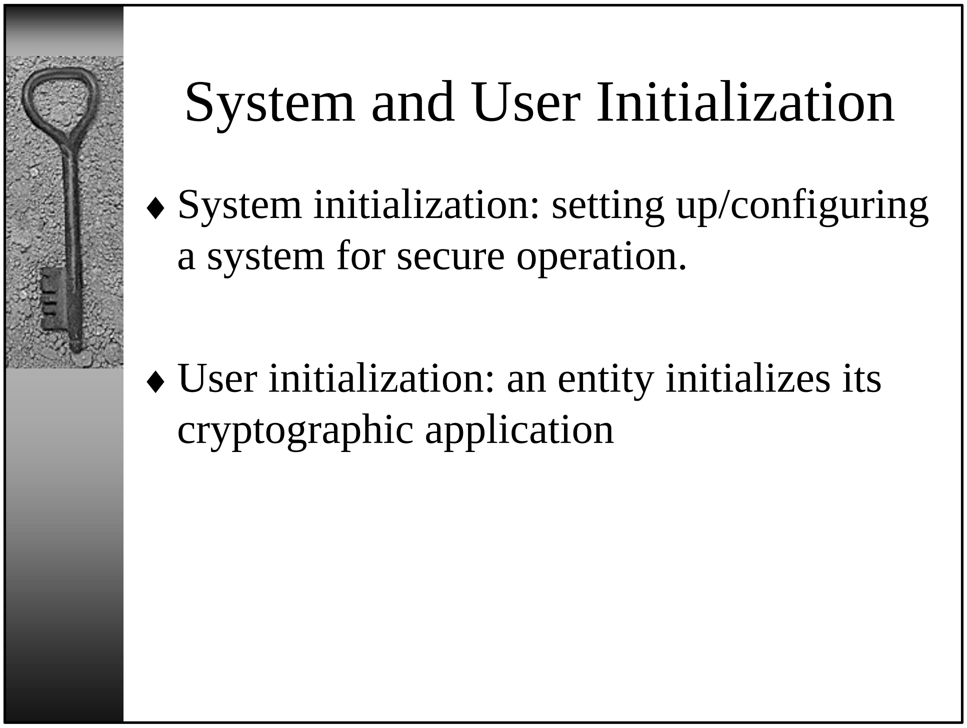
# Key Management Lifecycle

4.1 User Registration
4.2 System and User Initialization
4.3 Keying Material Installation
4.4 Key Establishment
4.5 Key Registration
4.6 Operational Use
4.7 Storage of Keying Material
4.8 Key Update
4.9 Key Recovery
4.10 Key De-registration and Destruction
4.11 Key Revocation



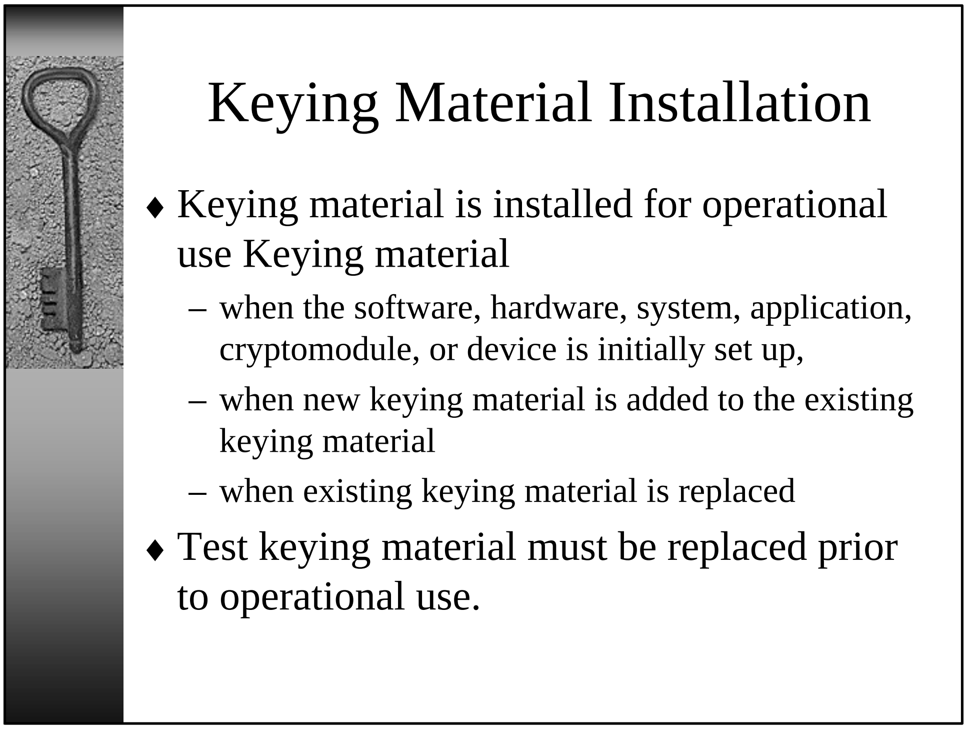
# User Registration

During registration, an entity becomes an authorized member of a security domain. This includes the acquisition or creation and exchange of initial keying material.



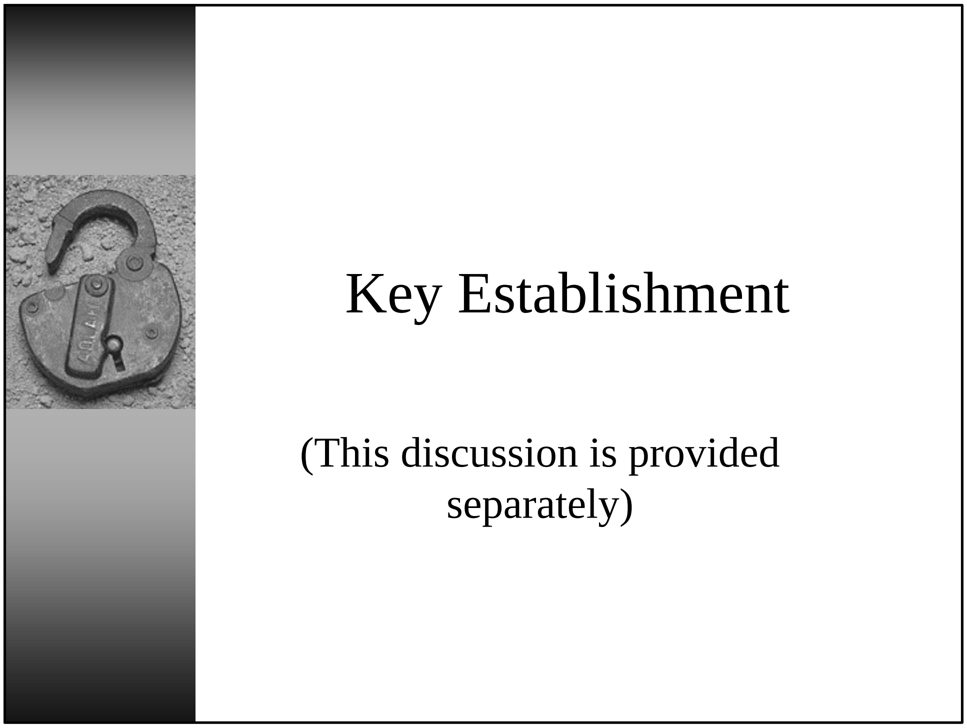
## System and User Initialization

- ♦ System initialization: setting up/configuring a system for secure operation.
- ♦ User initialization: an entity initializes its cryptographic application



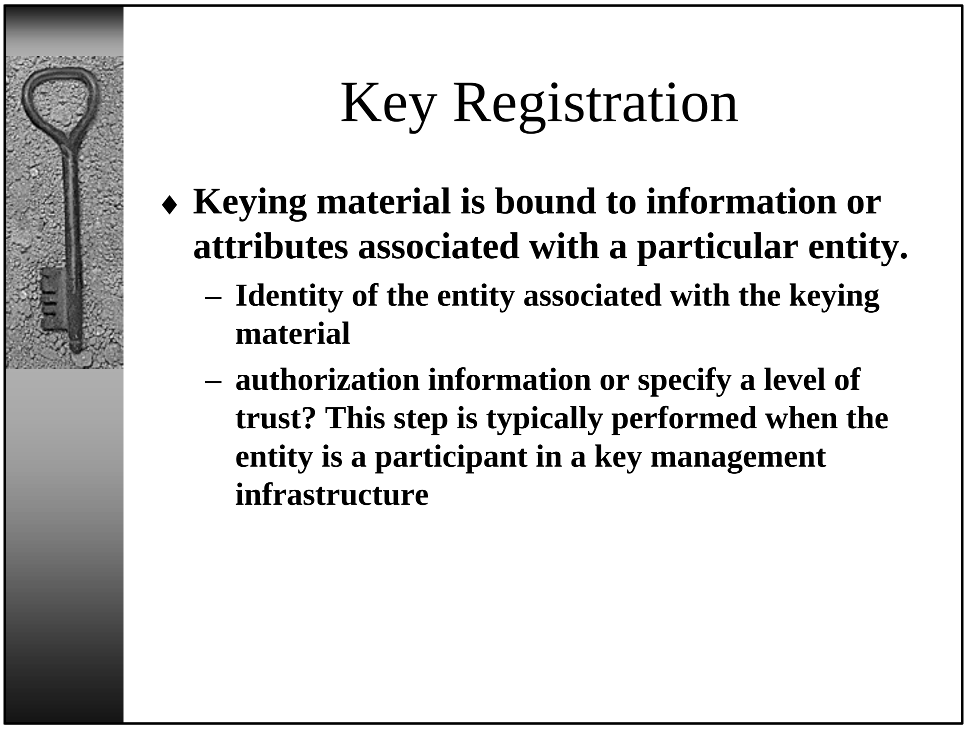
## Keying Material Installation

- ♦ Keying material is installed for operational use Keying material
  - – when the software, hardware, system, application, cryptomodule, or device is initially set up,
  - – when new keying material is added to the existing keying material
  - – when existing keying material is replaced
- ♦ Test keying material must be replaced prior to operational use.



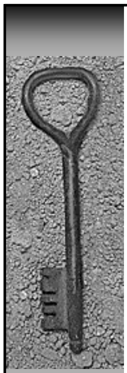
## Key Establishment

(This discussion is provided separately)



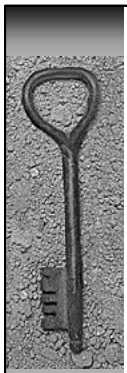
## Key Registration

- **Keying material is bound to information or attributes associated with a particular entity.**
  - **Identity of the entity associated with the keying material**
  - **authorization information or specify a level of trust? This step is typically performed when the entity is a participant in a key management infrastructure**



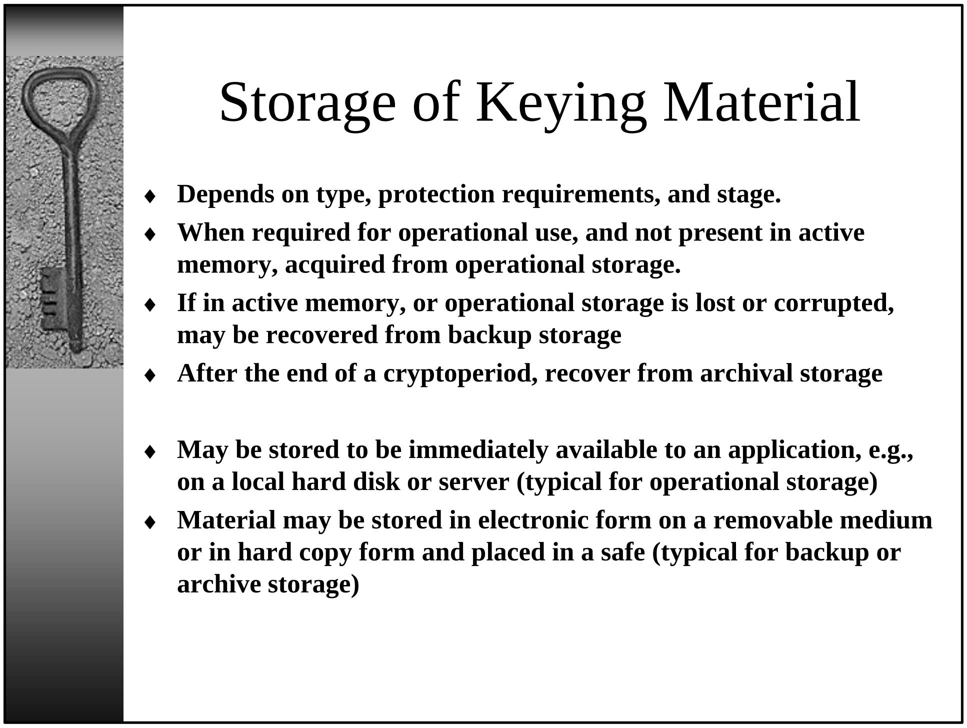
## Operational Use

- **The objective of the key management lifecycle is to facilitate the operational availability of keying material for standard cryptographic purposes.**
- **Under normal circumstances, a key remains operational until the end of the key's cryptoperiod.**



## Storage of Keying Material

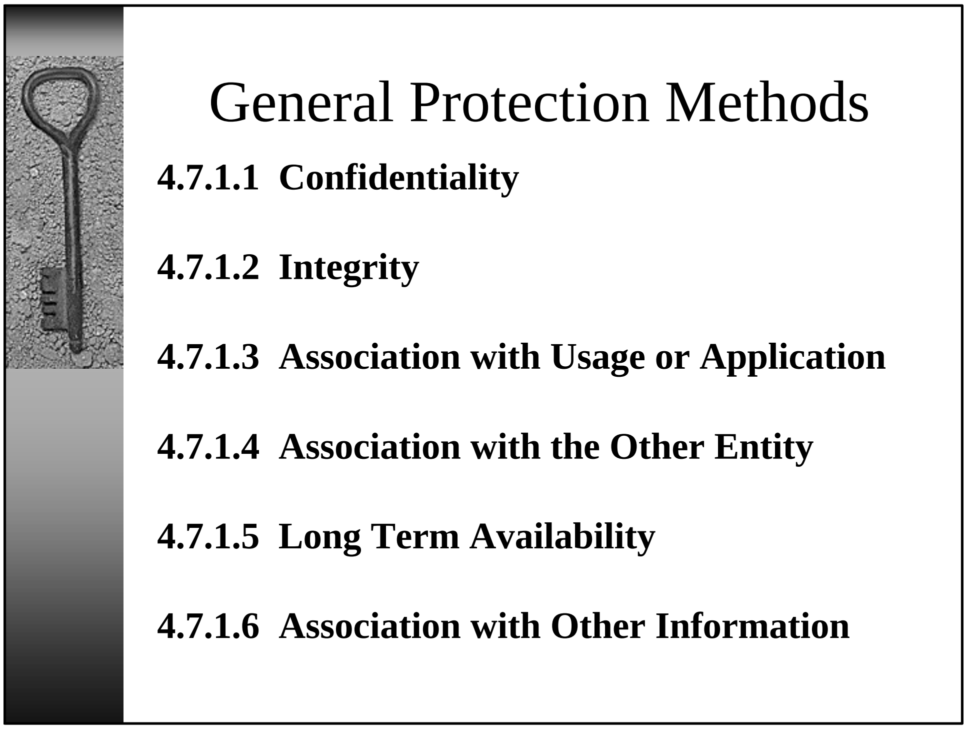
**4.7.1 General Protection Methods**
**Confidentiality**
**Integrity**
**Association With Usage or Application**
**Association With the Other Entity**
**Long Term Availability**
**Association With Other Information**
**4.7.2 Operational Storage**
**4.7.3 Backup Storage**
**4.7.4 Key Archive Storage**



# Storage of Keying Material

- **Depends on type, protection requirements, and stage.**
- **When required for operational use, and not present in active memory, acquired from operational storage.**
- **If in active memory, or operational storage is lost or corrupted, may be recovered from backup storage**
- **After the end of a cryptoperiod, recover from archival storage**

- **May be stored to be immediately available to an application, e.g., on a local hard disk or server (typical for operational storage)**
- **Material may be stored in electronic form on a removable medium or in hard copy form and placed in a safe (typical for backup or archive storage)**



# General Protection Methods

**4.7.1.1 Confidentiality**

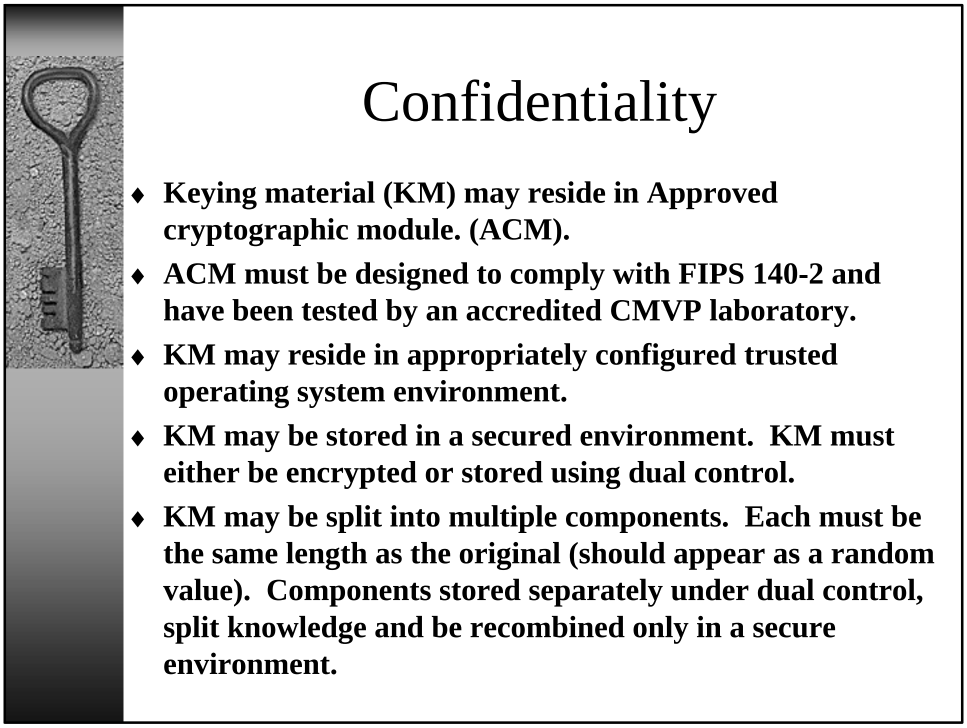
**4.7.1.2 Integrity**

**4.7.1.3 Association with Usage or Application**

**4.7.1.4 Association with the Other Entity**

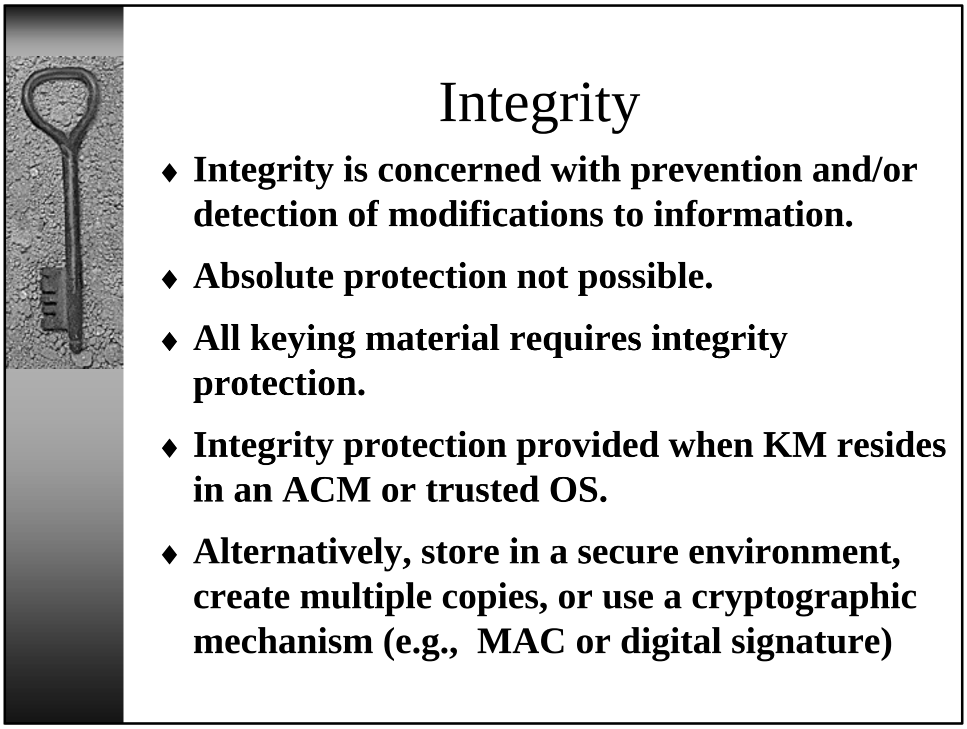
**4.7.1.5 Long Term Availability**

**4.7.1.6 Association with Other Information**



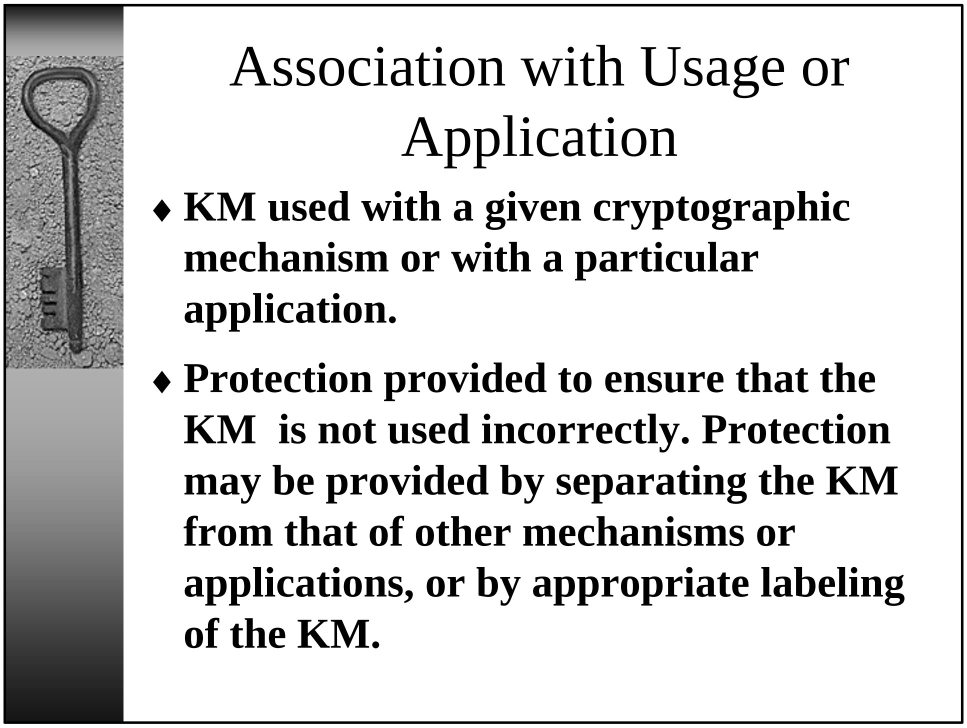
# Confidentiality

- ♦ Keying material (KM) may reside in Approved cryptographic module. (ACM).
- ♦ ACM must be designed to comply with FIPS 140-2 and have been tested by an accredited CMVP laboratory.
- ♦ KM may reside in appropriately configured trusted operating system environment.
- ♦ KM may be stored in a secured environment. KM must either be encrypted or stored using dual control.
- ♦ KM may be split into multiple components. Each must be the same length as the original (should appear as a random value). Components stored separately under dual control, split knowledge and be recombined only in a secure environment.



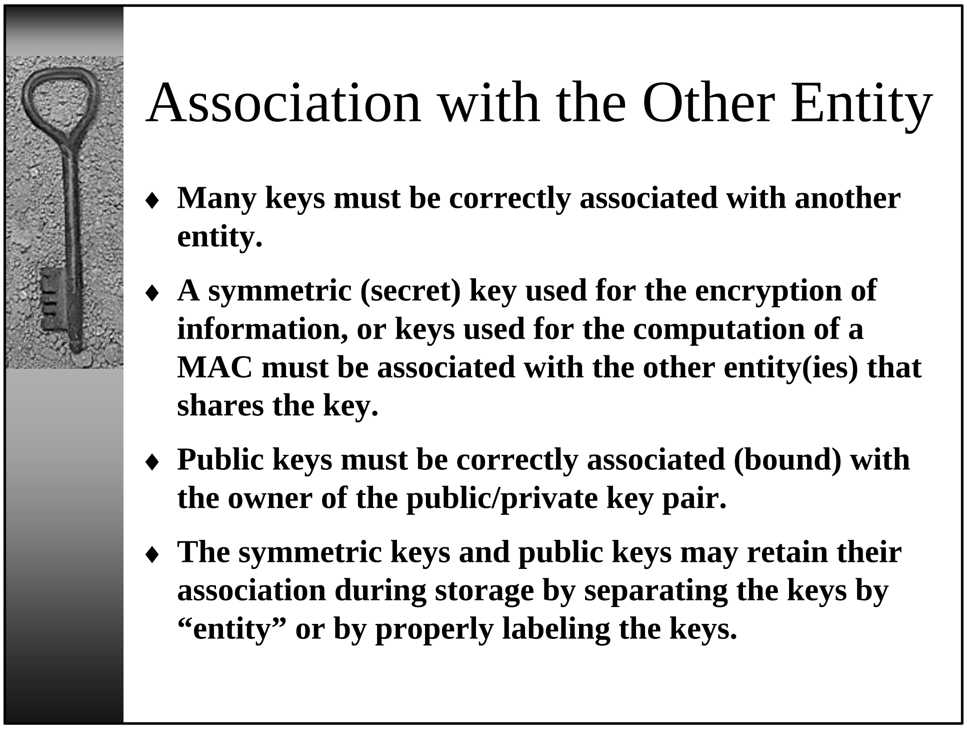
# Integrity

- ♦ Integrity is concerned with prevention and/or detection of modifications to information.
- ♦ Absolute protection not possible.
- ♦ All keying material requires integrity protection.
- ♦ Integrity protection provided when KM resides in an ACM or trusted OS.
- ♦ Alternatively, store in a secure environment, create multiple copies, or use a cryptographic mechanism (e.g., MAC or digital signature)



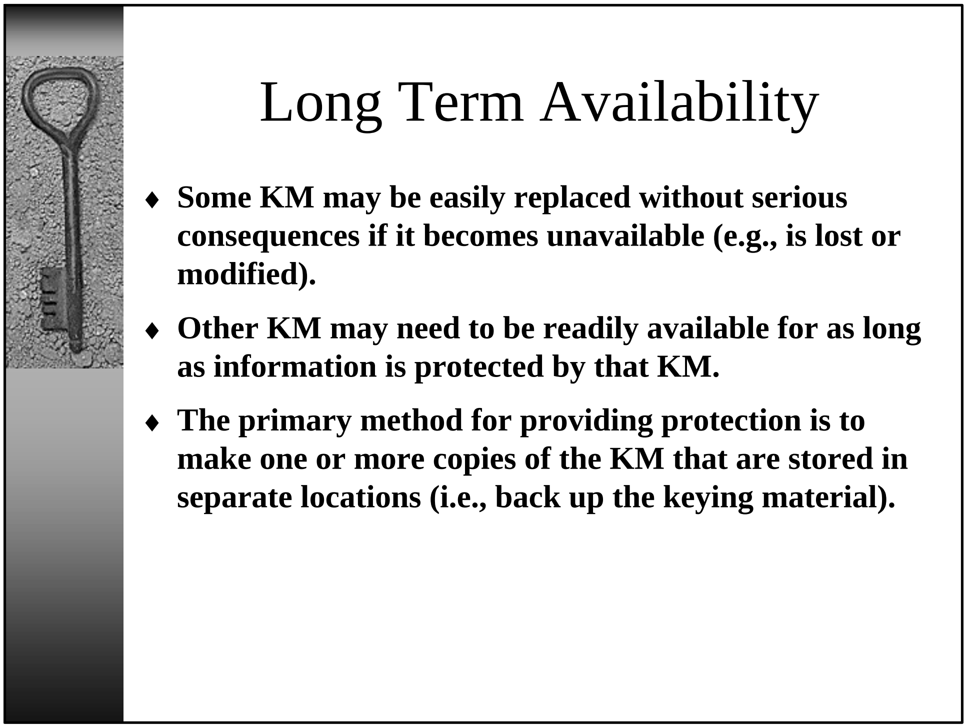
## Association with Usage or Application

- ♦ **KM used with a given cryptographic mechanism or with a particular application.**
- ♦ **Protection provided to ensure that the KM  is not used incorrectly. Protection may be provided by separating the KM from that of other mechanisms or applications, or by appropriate labeling of the KM.**



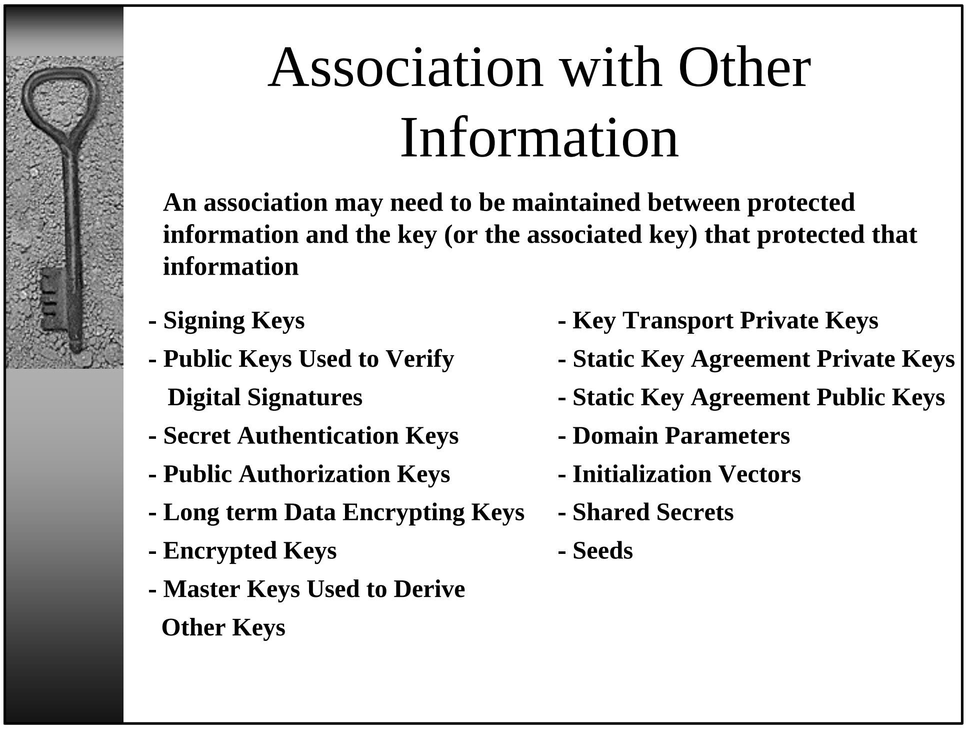
## Association with the Other Entity

- ♦ **Many keys must be correctly associated with another entity.**
- ♦ **A symmetric (secret) key used for the encryption of information, or keys used for the computation of a MAC must be associated with the other entity(ies) that shares the key.**
- ♦ **Public keys must be correctly associated (bound) with the owner of the public/private key pair.**
- ♦ **The symmetric keys and public keys may retain their association during storage by separating the keys by "entity" or by properly labeling the keys.**



## Long Term Availability

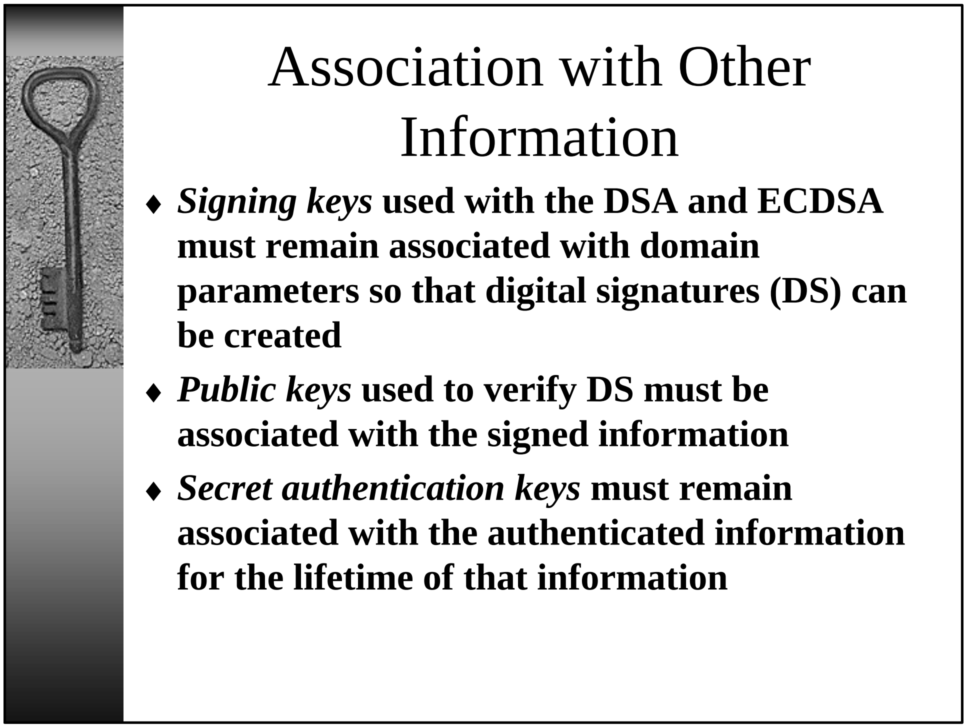
- ♦ Some KM may be easily replaced without serious consequences if it becomes unavailable (e.g., is lost or modified).
- ♦ Other KM may need to be readily available for as long as information is protected by that KM.
- ♦ The primary method for providing protection is to make one or more copies of the KM that are stored in separate locations (i.e., back up the keying material).



## Association with Other Information

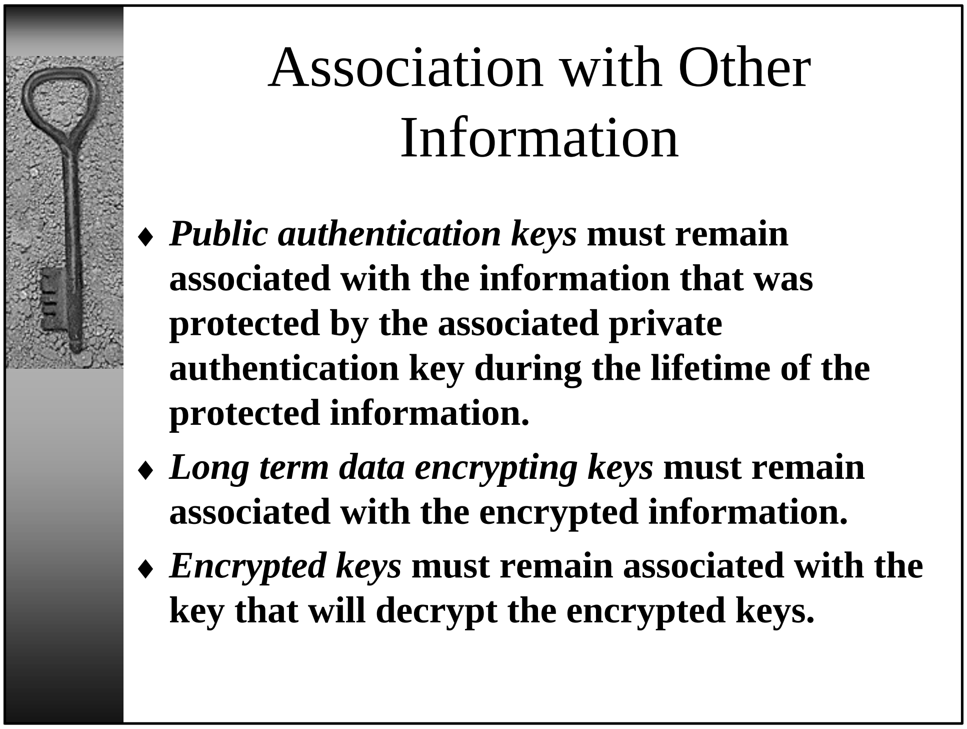
**An association may need to be maintained between protected information and the key (or the associated key) that protected that information**

**- Signing Keys**
**- Public Keys Used to Verify Digital Signatures**
**- Secret Authentication Keys**
**- Public Authorization Keys**
**- Long term Data Encrypting Keys**
**- Encrypted Keys**
**- Master Keys Used to Derive Other Keys**
**- Key Transport Private Keys**
**- Static Key Agreement Private Keys**
**- Static Key Agreement Public Keys**
**- Domain Parameters**
**- Initialization Vectors**
**- Shared Secrets**
**- Seeds**



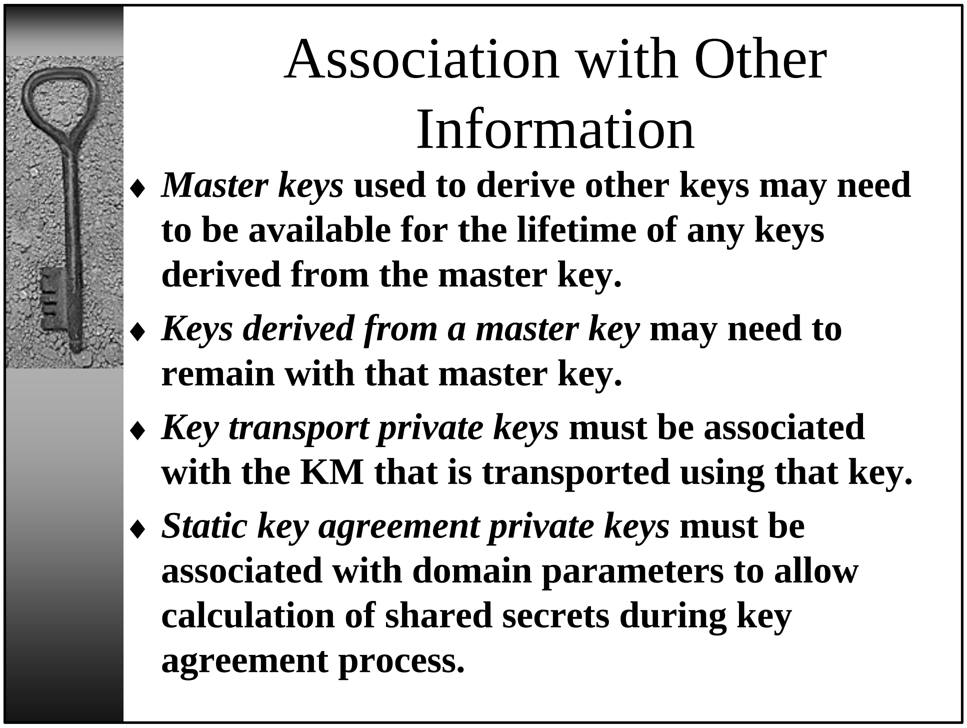
## Association with Other Information

- *Signing keys* **used with the DSA and ECDSA must remain associated with domain parameters so that digital signatures (DS) can be created**
- *Public keys* **used to verify DS must be associated with the signed information**
- *Secret authentication keys* **must remain associated with the authenticated information for the lifetime of that information**



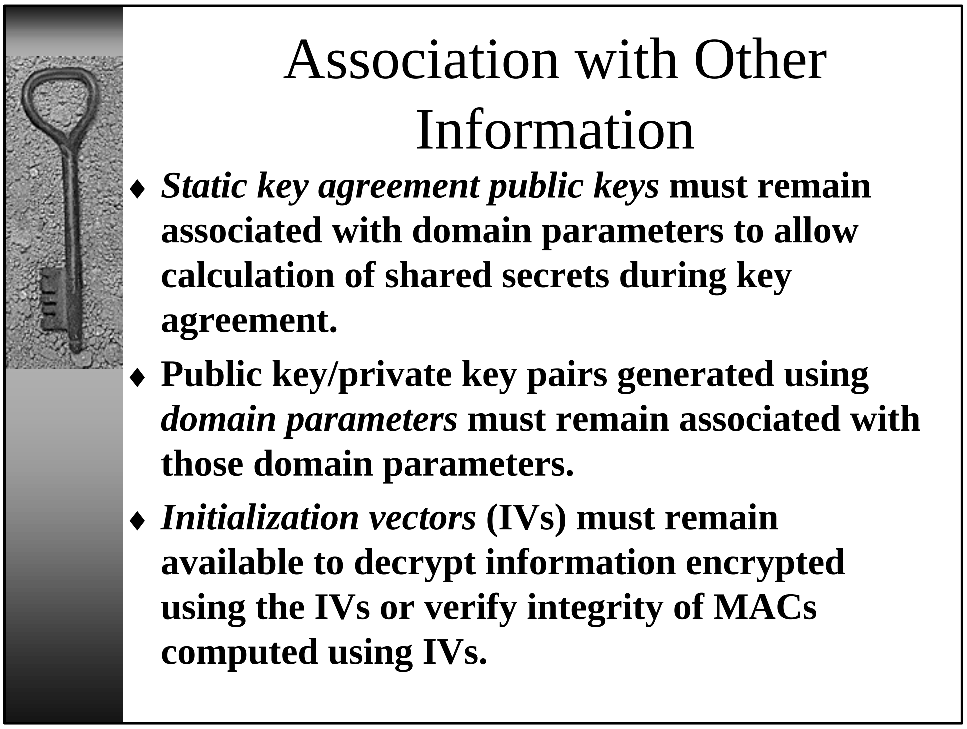
## Association with Other Information

- *Public authentication keys* **must remain associated with the information that was protected by the associated private authentication key during the lifetime of the protected information.**
- *Long term data encrypting keys* **must remain associated with the encrypted information.**
- *Encrypted keys* **must remain associated with the key that will decrypt the encrypted keys.**



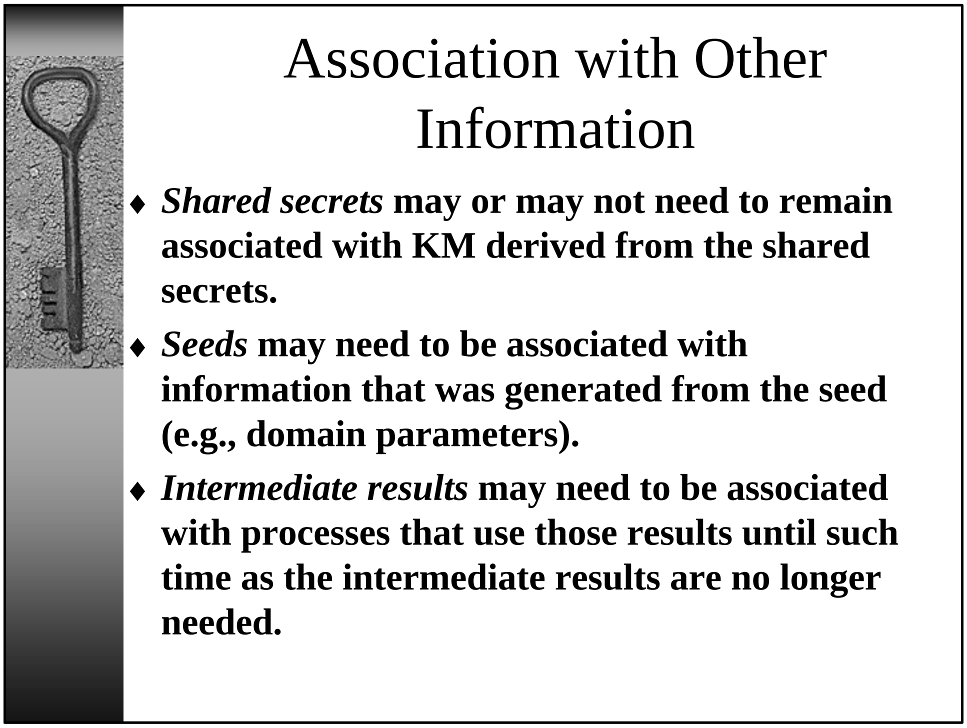
## Association with Other Information

- ♦ *Master keys* used to derive other keys may need to be available for the lifetime of any keys derived from the master key.
- ♦ *Keys derived from a master key* may need to remain with that master key.
- ♦ *Key transport private keys* must be associated with the KM that is transported using that key.
- ♦ *Static key agreement private keys* must be associated with domain parameters to allow calculation of shared secrets during key agreement process.



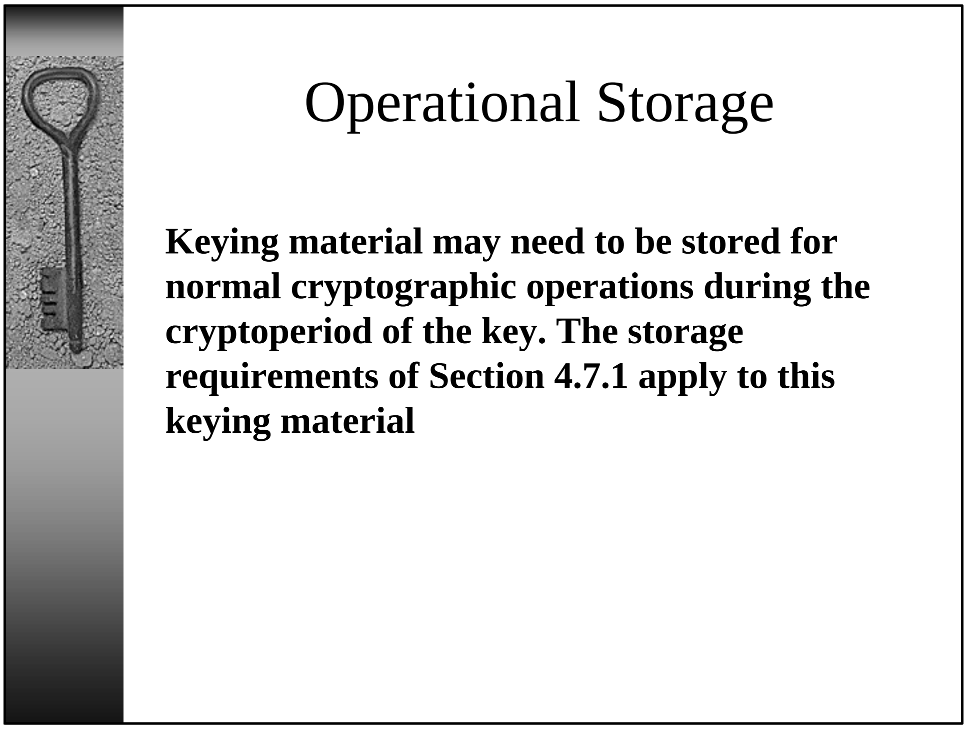
## Association with Other Information

- ♦ *Static key agreement public keys* must remain associated with domain parameters to allow calculation of shared secrets during key agreement.
- ♦ Public key/private key pairs generated using *domain parameters* must remain associated with those domain parameters.
- ♦ *Initialization vectors* (IVs) must remain available to decrypt information encrypted using the IVs or verify integrity of MACs computed using IVs.



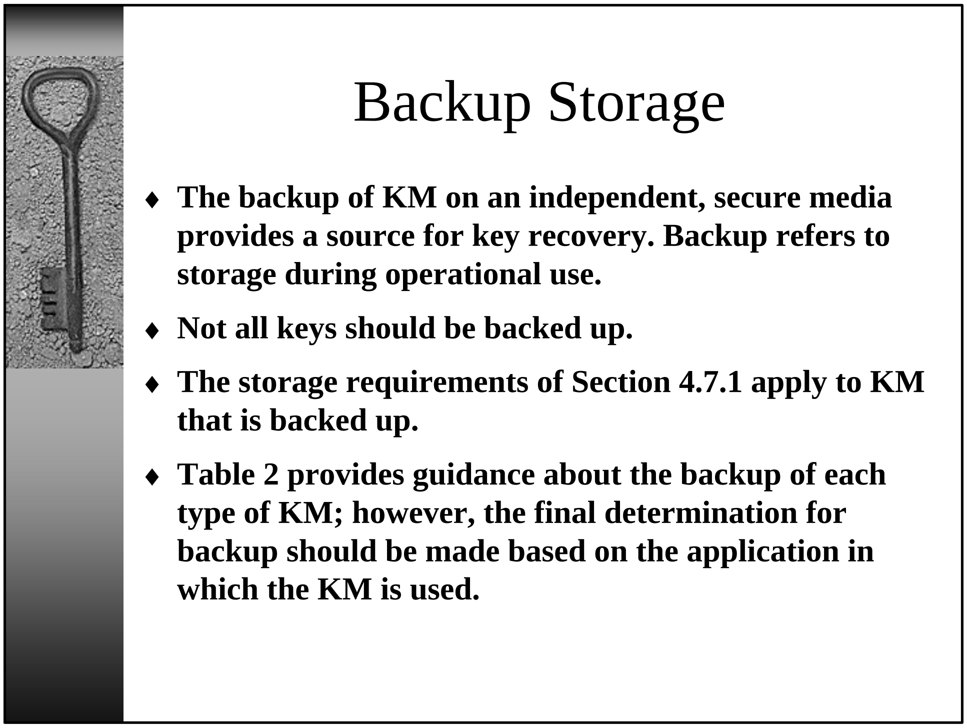
## Association with Other Information

- *Shared secrets* **may or may not need to remain associated with KM derived from the shared secrets.**
- *Seeds* **may need to be associated with information that was generated from the seed (e.g., domain parameters).**
- *Intermediate results* **may need to be associated with processes that use those results until such time as the intermediate results are no longer needed.**



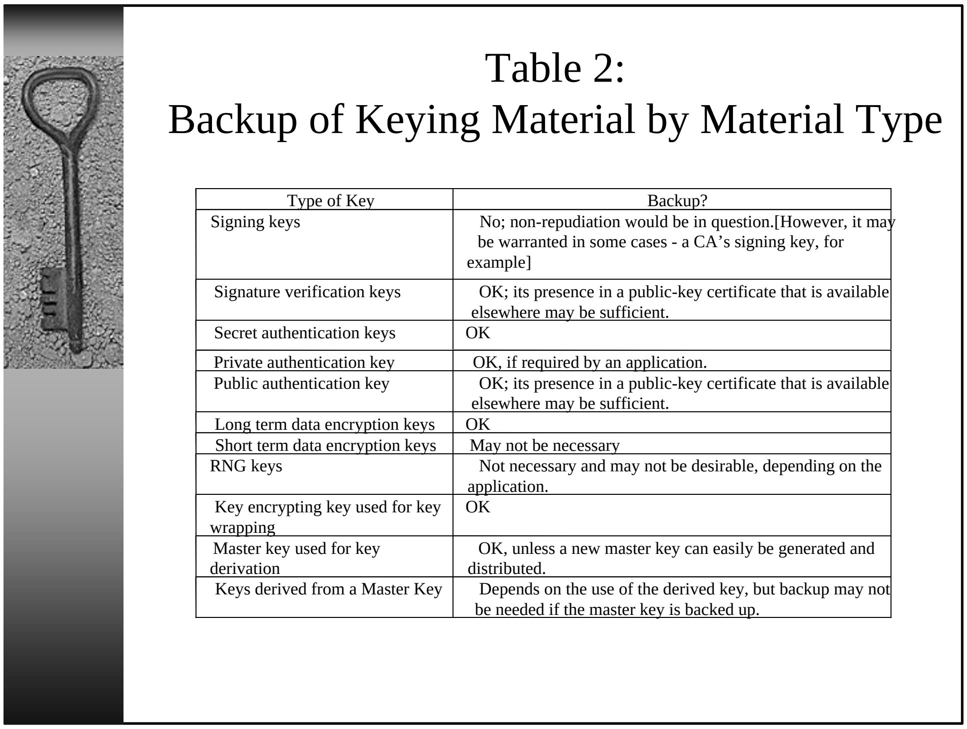
## Operational Storage

**Keying material may need to be stored for normal cryptographic operations during the cryptoperiod of the key. The storage requirements of Section 4.7.1 apply to this keying material**



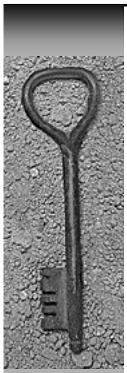
# Backup Storage

- The backup of KM on an independent, secure media provides a source for key recovery. Backup refers to storage during operational use.
- Not all keys should be backed up.
- The storage requirements of Section 4.7.1 apply to KM that is backed up.
- Table 2 provides guidance about the backup of each type of KM; however, the final determination for backup should be made based on the application in which the KM is used.



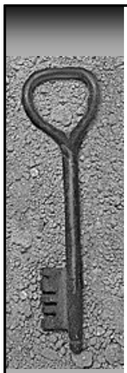
# Table 2:
# Backup of Keying Material by Material Type

| Type of Key | Backup? |
|---|---|
| Signing keys | No; non-repudiation would be in question.[However, it may be warranted in some cases - a CA’s signing key, for example] |
| Signature verification keys | OK; its presence in a public-key certificate that is available elsewhere may be sufficient. |
| Secret authentication keys | OK |
| Private authentication key | OK, if required by an application. |
| Public authentication key | OK; its presence in a public-key certificate that is available elsewhere may be sufficient. |
| Long term data encryption keys | OK |
| Short term data encryption keys | May not be necessary |
| RNG keys | Not necessary and may not be desirable, depending on the application. |
| Key encrypting key used for key wrapping | OK |
| Master key used for key derivation | OK, unless a new master key can easily be generated and distributed. |
| Keys derived from a Master Key | Depends on the use of the derived key, but backup may not be needed if the master key is backed up. |



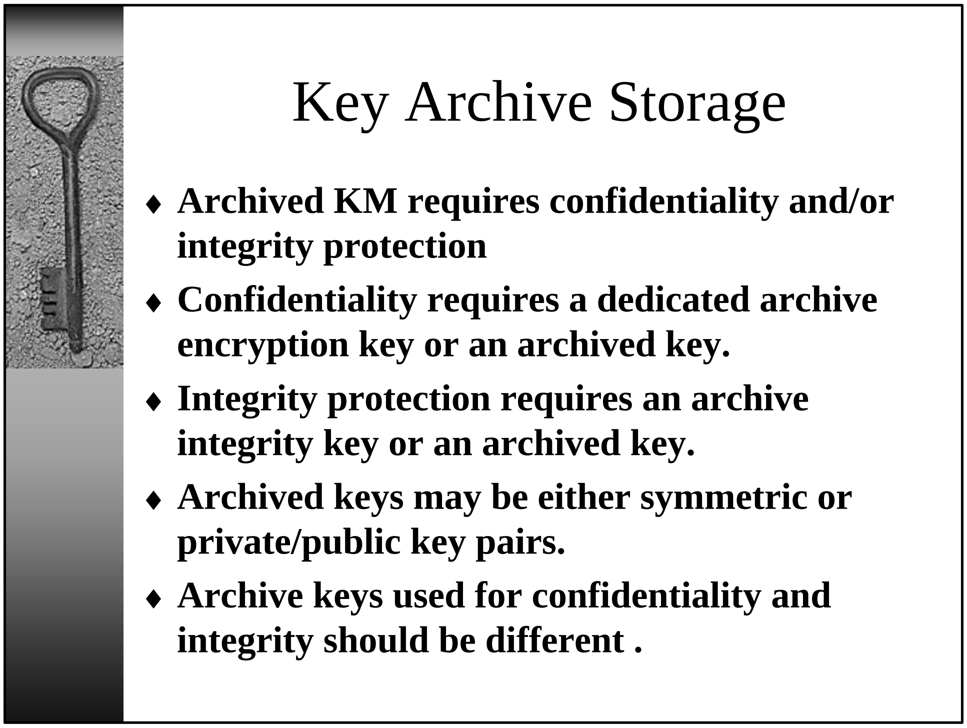
## Table 2:
## Backup of Keying Material by Material Type

| Type of Key | Backup? |
|---|---|
| Key transport private keys | OK |
| Key transport public keys | OK; presence in a public-key certificate available elsewhere may be sufficient. |
| Static key agreement private keys | No, unless needed for reconstruction during key recovery? |
| Static key agreement public keys | OK; its presence in a public-key certificate that is available elsewhere may be sufficient. |
| Ephemeral key agreement private keys | No |
| Ephemeral key agreement public keys | No, unless needed for reconstruction during key recovery? |
| Secret authorization key | OK |
| Private authorization key | OK |
| Public authorization key | OK; its presence in a public-key certificate that is available elsewhere may be sufficient. |
| Domain parameters | OK |
| Initialization vectors | OK, if necessary |
| Shared secrets | No, unless needed for reconstruction during key recovery? |
| Seeds | No, unless required for the validation of domain parameters |
| Intermediate results | No |



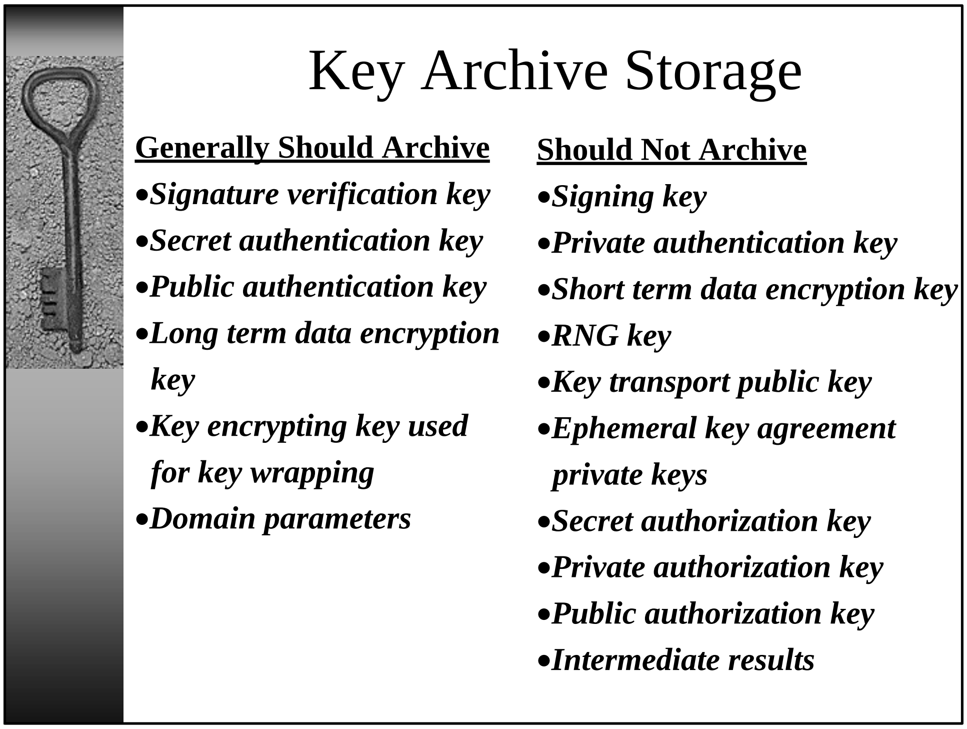
# Key Archive Storage

- **A KM archive is a repository containing KM of historical interest**
- **Not all KM needs to be archived**
- **Archived KM may be static or may need to be re-encrypted under a new key**
- **Archived data should be stored separately from active data**
- **Multiple copies should be available and stored separately from each other.**
- **The KM should be destroyed when no longer required.**



# Key Archive Storage

- ♦ **Archived KM requires confidentiality and/or integrity protection**
- ♦ **Confidentiality requires a dedicated archive encryption key or an archived key.**
- ♦ **Integrity protection requires an archive integrity key or an archived key.**
- ♦ **Archived keys may be either symmetric or private/public key pairs.**
- ♦ **Archive keys used for confidentiality and integrity should be different .**



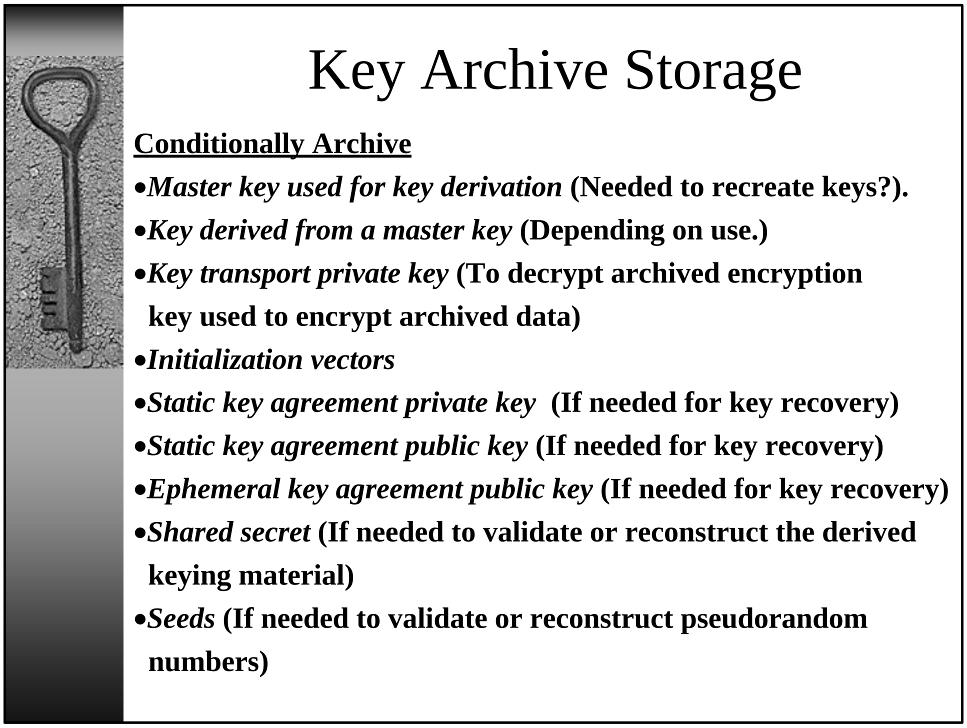
# Key Archive Storage

**<u>Generally Should Archive</u>**

- ***Signature verification key***
- ***Secret authentication key***
- ***Public authentication key***
- ***Long term data encryption key***
- ***Key encrypting key used for key wrapping***
- ***Domain parameters***

**<u>Should Not Archive</u>**

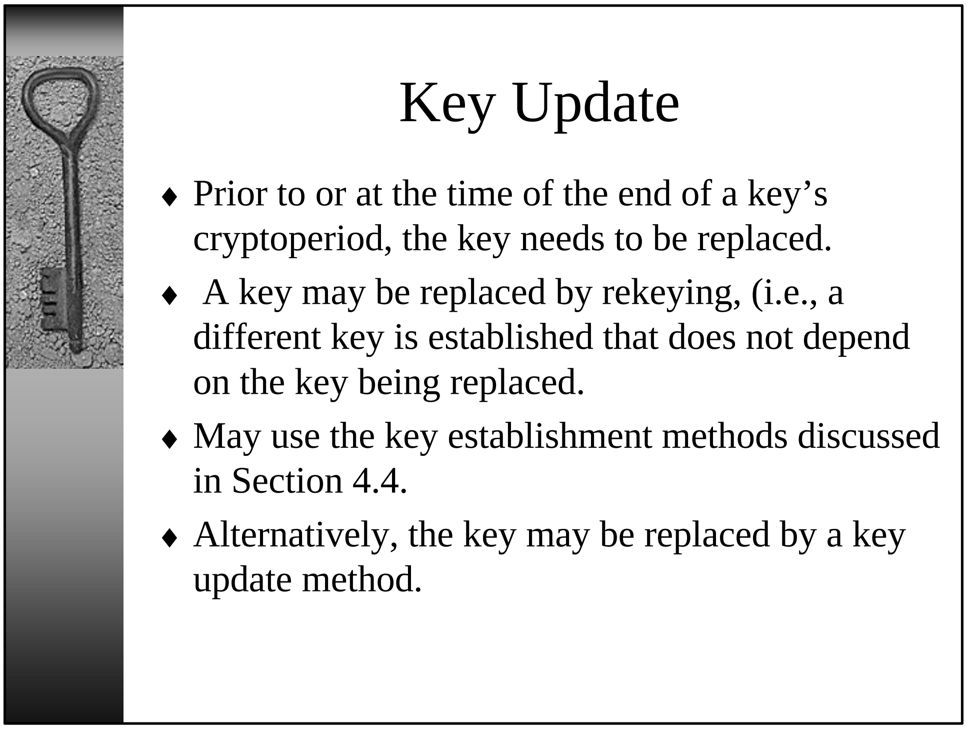
- ***Signing key***
- ***Private authentication key***
- ***Short term data encryption key***
- ***RNG key***
- ***Key transport public key***
- ***Ephemeral key agreement private keys***
- ***Secret authorization key***
- ***Private authorization key***
- ***Public authorization key***
- ***Intermediate results***



# Key Archive Storage

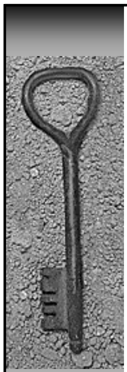
**<u>Conditionally Archive</u>**

- ***Master key used for key derivation*** **(Needed to recreate keys?).**
- ***Key derived from a master key*** **(Depending on use.)**
- ***Key transport private key*** **(To decrypt archived encryption key used to encrypt archived data)**
- ***Initialization vectors***
- ***Static key agreement private key*** **(If needed for key recovery)**
- ***Static key agreement public key*** **(If needed for key recovery)**
- ***Ephemeral key agreement public key*** **(If needed for key recovery)**
- ***Shared secret*** **(If needed to validate or reconstruct the derived keying material)**
- ***Seeds*** **(If needed to validate or reconstruct pseudorandom numbers)**



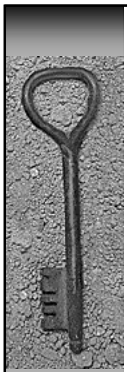
# Key Update

- ♦ Prior to or at the time of the end of a key’s cryptoperiod, the key needs to be replaced.
- ♦ A key may be replaced by rekeying, (i.e., a different key is established that does not depend on the key being replaced.
- ♦ May use the key establishment methods discussed in Section 4.4.
- ♦ Alternatively, the key may be replaced by a key update method.



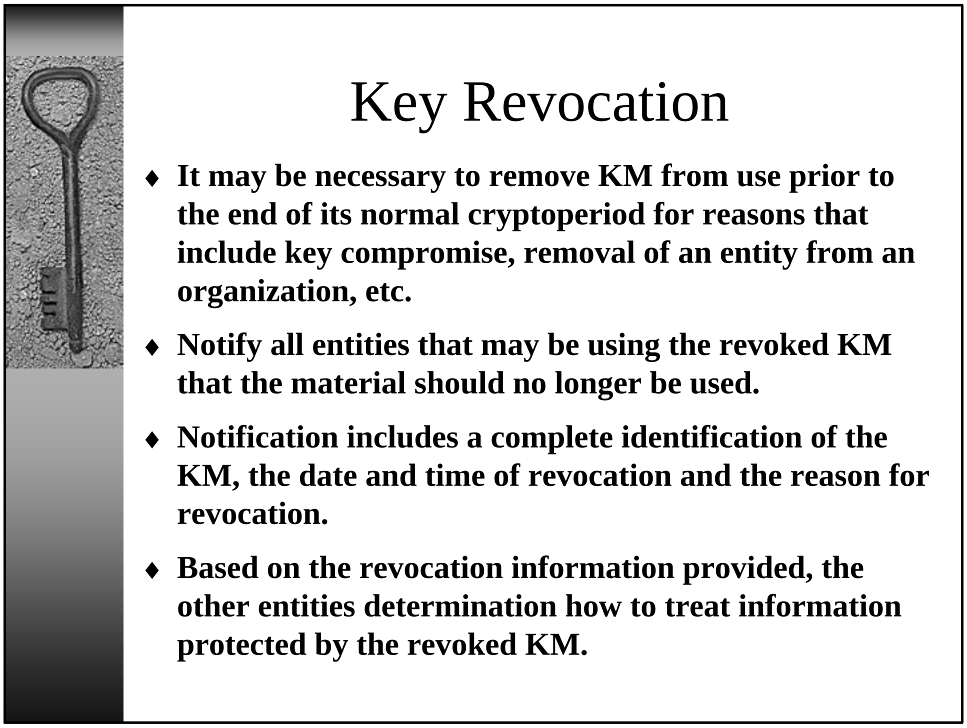
## Key Recovery (KR)

- **The process of retrieving the KM from backup or archive storage is called key recovery.**
- **There are several different KR techniques.**
- **The information required to recover that key may be different for each application or each KR technique.**
- **The term “Key Recovery Information” (KRI) refers to the aggregate of information needed to recover the key.**
- **The KRI includes the key to be recovered and other cryptographic data, the time when the key was created, the identity of the owner of the key and any conditions that must be met by a requestor to be able to recover the KM.**



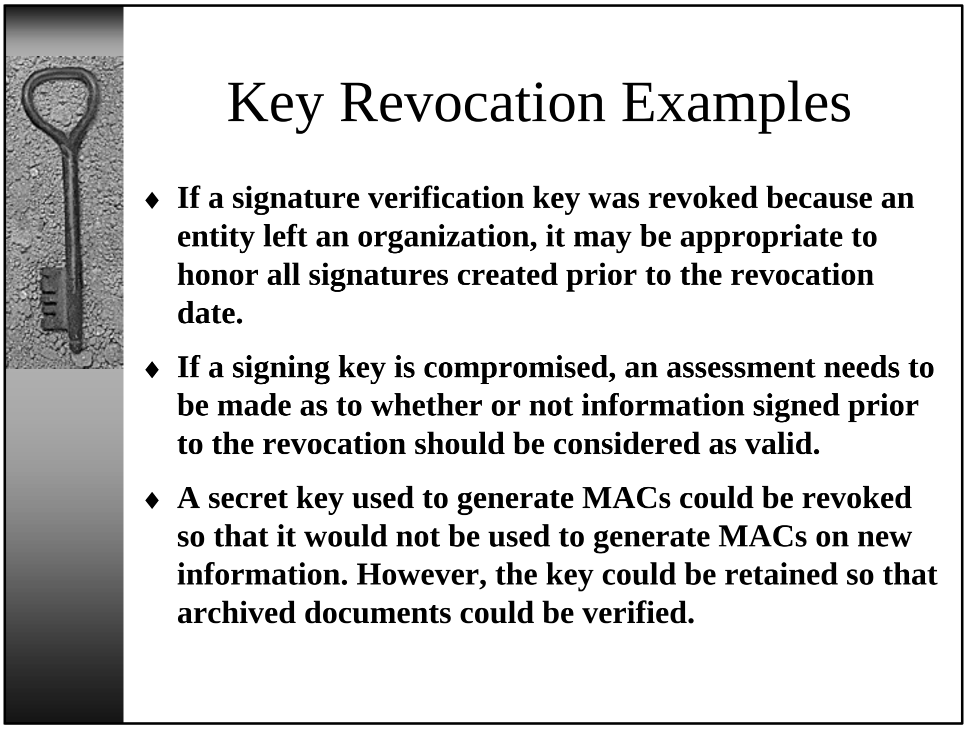
## Key De-Registration and Destruction

- **When there are no further requirements for retaining KM or its association with an entity, the key should be de-registered (i.e., all records of the KM and its associations should be destroyed), and all copies of the private or secret key should be destroyed.**
- **Any media on which the KM was stored should be erased in a manner that removes all traces of the KM so that it cannot be recovered by either physical or electronic means.**



# Key Revocation

- ♦ It may be necessary to remove KM from use prior to the end of its normal cryptoperiod for reasons that include key compromise, removal of an entity from an organization, etc.
- ♦ Notify all entities that may be using the revoked KM that the material should no longer be used.
- ♦ Notification includes a complete identification of the KM, the date and time of revocation and the reason for revocation.
- ♦ Based on the revocation information provided, the other entities determination how to treat information protected by the revoked KM.



# Key Revocation Examples

- ♦ If a signature verification key was revoked because an entity left an organization, it may be appropriate to honor all signatures created prior to the revocation date.
- ♦ If a signing key is compromised, an assessment needs to be made as to whether or not information signed prior to the revocation should be considered as valid.
- ♦ A secret key used to generate MACs could be revoked so that it would not be used to generate MACs on new information. However, the key could be retained so that archived documents could be verified.