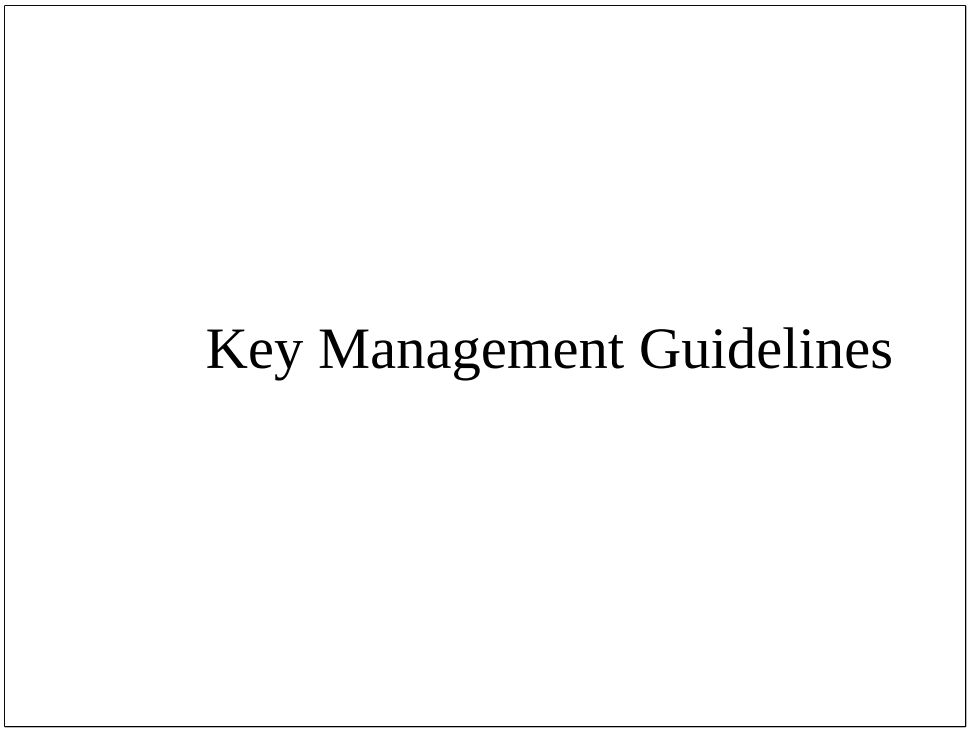


Key Management Guidelines

•This document has been developed specifically for the key management workshop.

•It is not intended as the draft of a finished document.

•The document should be reviewed with an eye to determining whether or not the concepts are correct and all necessary topics have been addressed, or it appears that they will be addressed in the future.

•This workshop document is being developed simultaneously with the key establishment schemes document [FIPS XXX].

Key Management Guideline

Table of Contents

•Your opinion is solicited on the basic approach taken for the final guideline, and the concepts and details addressed.

•Are they correct and complete?

•If not, you are requested to provide comments to assist in the development of the guideline.

Introduction

1.1 Goal/Purpose

1.2 Audience

1.3 Scope

1.4 Security Services

1.5 Content/Organization

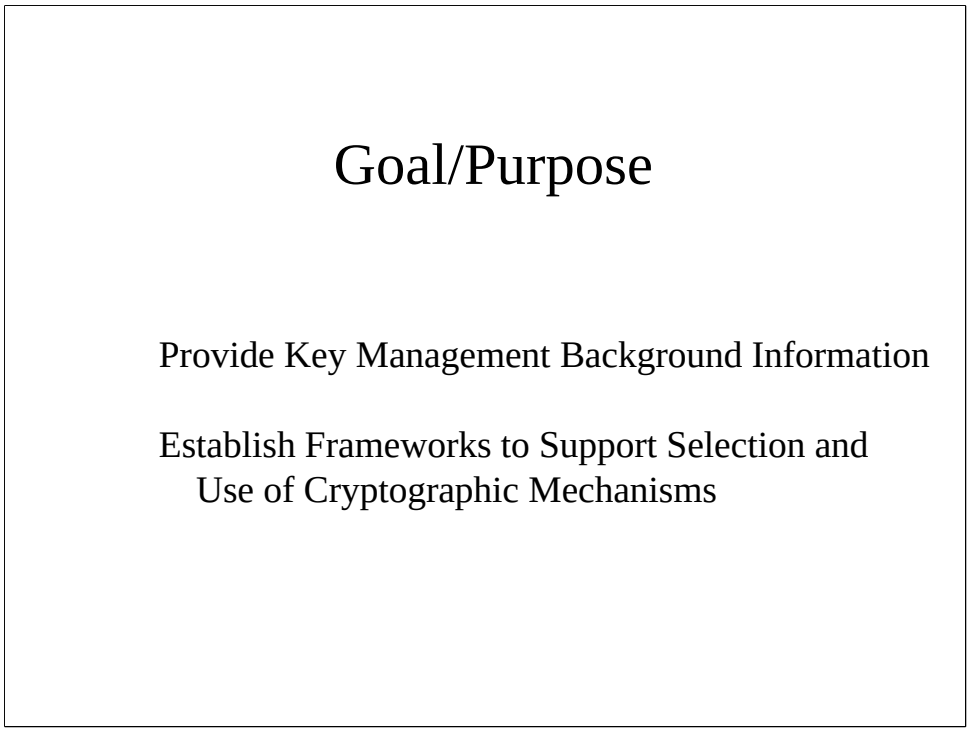
Section 1 - Introduction

•Since NIST published DES in 1977, a suite of approved algorithms for unclassified but sensitive applications has been growing.

•New classes of algorithms have been added, such as secure hash algorithms and asymmetric algorithms for digital signatures.

•The suite of algorithms now provides different levels of cryptographic strength through a variety of key sizes.

• In addition, the algorithms may be combined in many ways to support increasingly complex protocols and applications.

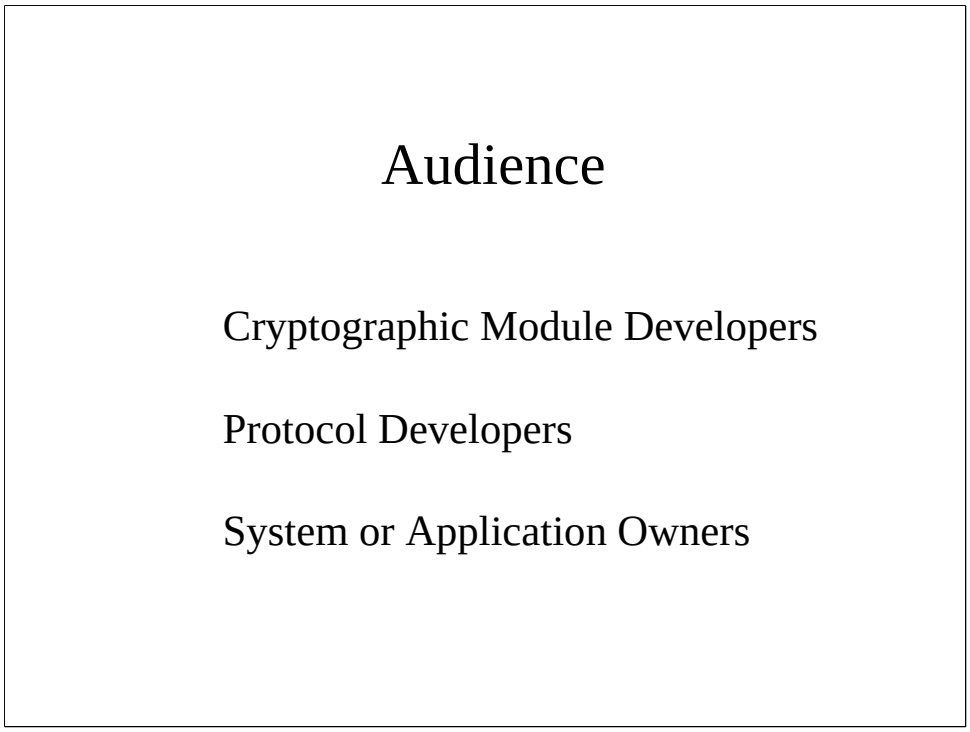


Goal/Purpose

•Users and developers are presented with many new choices in their use of cryptographic mechanisms.

• Inappropriate choices may result in an illusion of security, but little or no real security for the protocol or application.

•This workshop document provides background information and establishes frameworks to support appropriate decisions when selecting and using cryptographic mechanisms.



Audience

This document assumes that the reader has a basic understanding of cryptography.

There are three primary audiences for this document.

•Cryptographic module developers may benefit from this document through a greater understanding of features required to support the intended range of applications.

•Protocol developers may identify appropriate suites of algorithms and gain a greater understanding of the security services provided by those algorithms.

•System or application owners may use this document to determine which configuration settings are most appropriate for their information.

Suggestions concerning the intended audience will be solicited during the detailed discussion session.

Scope

Cryptographic Algorithms, Infrastructures, Protocols, and Applications

Management of Cryptographic Keys

- Generation
- Use
- Destruction

Scope

The scope of the guideline encompasses cryptographic algorithms, infrastructures, protocols, and applications.

•All cryptographic algorithms currently approved or under active consideration by NIST for the protection of unclassified but sensitive information are in scope.

•Common key distribution infrastructures for approved algorithms are considered, but protocols and applications are limited to those widely used by Federal agencies, such as SSL.

•This document focuses on management of generation, use, and eventual destruction of keys.

•Related topics, such as algorithm selection and appropriate key size, cryptographic policy, and cryptographic module selection, are also included.

•The document does not address implementation details for cryptographic modules. These details are addressed in other sources.

Note that, in many cases, cryptographic management requirements will be determined by key type or usage. The proposed Guidelines document often lists issues and suggests guidance by key type.

Security Services Covered

Cryptography may be used to perform several basic security services:

•confidentiality - prevention of disclosure to unauthorized entities,

•data integrity - verification that data has not been altered,

•authentication - verification of the creator of data,

•non-repudiation - proof of the integrity of the origin of data.

These services may also be required to protect cryptographic keys.

Note that suggestions concerning additions and/or shifts in emphasis with respect to security services will be solicited during the detailed discussion session.

Content/Organization

Glossary of Terms and Acronyms
Cryptographic Algorithms, Keys and Other Keying Material
Key Management Life Cycle
General Key Management Guidance
Guidance for Selected Infrastructures
Guidance -for Selected Applications
Appendices As Required

Content/Organization

•The document is organized so that each section builds upon earlier material.

• Frameworks and classifications established in the background material are essential to discussions of specific protocols and applications later in the document.

Suggestions regarding content, organization, and requirements for specific appendices will be welcomed during the detailed discussion session.

Glossary

2.1 Glossary of Terms

2.2 Acronyms

Glossary

The detailed discussion of the Glossary should be restricted to major issues. Minutia can be dealt with in post-workshop correspondence.

Cryptographic Algorithms, Keys, and Other Keying Material

3.1 Classes of Cryptographic Algorithms

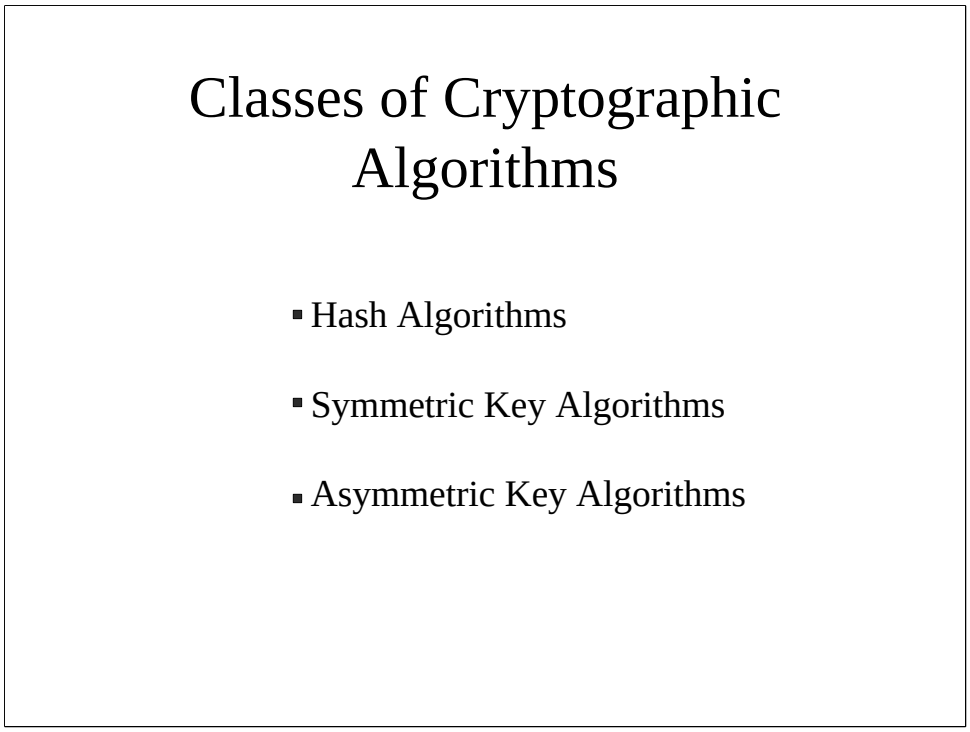
3.2 Cryptographic Algorithm Functionality

3.3 Cryptographic Keys and Other Keying Material

Cryptographic Algorithms, Keys, and Other Keying Material

•This section describes Approved cryptographic algorithms that make use of cryptographic keys to provide security services such as confidentiality, data integrity, authentication, and non-repudiation.

•A general discussion on keys and other keying material follows the overview of cryptographic algorithms.



Classes of Cryptographic Algorithms

There are three basic classes of Approved cryptographic algorithms. The classes are defined by the number of cryptographic keys that must be used in conjunction with the algorithm.

Hash algorithms generate a small message digest from a large message and require no keys. They are used in Message Authentication Codes (MACs), digital signatures, key establishment, and random number generation.

Symmetric key algorithms (sometimes known as a secret key algorithms) use a single key to transform data. (Uses: data confidentiality, authentication and integrity services, as part of the key establishment process, and pseudorandom number generation.

Asymmetric key algorithms, or public key algorithms, use two related keys: a public key and a private key. Access to the public key does not reveal the private key. Public key algorithms are commonly used to compute digital signatures and to establish cryptographic keying material

Suggestions regarding the cryptographic algorithm taxonomy employed in the Guidelines document will be solicited during the detailed discussion session.

Cryptographic Algorithm Functionality

3.2.1 Hash Function

3.2.2 Encryption/Decryption Algorithms

3.2.3 Message Authentication Codes

3.2.4 Digital Signature Algorithms

3.2.5 Key Establishment Algorithms

3.2.6 Random Number Generation

Cryptographic Algorithm Functionality

Algorithms, techniques, and applicable standards and guidelines are provided for each function. In many cases, the same algorithm may be used to provide multiple services.

Hash function - used in conjunction with a digital signature algorithm to compute a digital signature, as part of a random number generator, or with a key to provide a Keyed-Hash MAC.

The approved algorithms for encryption/decryption are Advanced Encryption Standard and Triple DES symmetric key algorithms.

.A MAC is a cryptographic checksum on the data used to provide assurance that data has not changed and that the MAC was computed by the expected entity. MAC computation requires a secret key {shared by the party that generates the MAC and the intended recipient(s) of the MAC) and the data on which the MAC is calculated. Either encryption or hash algorithms may be used.

Cryptographic Algorithm Functionality

3.2.1 Hash Function

3.2.2 Encryption/Decryption Algorithms

3.2.3 Message Authentication Codes

3.2.4 Digital Signature Algorithms

3.2.5 Key Establishment Algorithms

3.2.6 Random Number Generation

Cryptographic Algorithm Functionality

Digital signatures are used in conjunction with hash algorithmsto provide authentication, integrity and non-repudiation. FIPS 186-2 defines the Digital Signature Algorithm and adopts RSA and Rabin-Williams and the Elliptic Curve Digital Signature Algorithm..

Key establishment and key agreement algorithms are used to set up keys to be used between communicating entities. Key transport is the distribution of a key (and other keying material) from one entity to another entity. The keying material is usually encrypted by the sending entity. Key agreement uses public key techniques for creation of shared keying material.

Deterministic and non-deterministic random number generators are used for the generation of keying material. Deterministic RNGs use cryptographic algorithms and associated keying material to generate random numbers, and are often called pseudorandom number generators; non-deterministic RNGs produce output that is dependent on some unpredictable physical source that is outside human control.

Discussion is anticipated regarding the list of cryptographic functions, algorithms and techniques included, and emphasis afforded each function.

Cryptographic Keys and Other Keying Material

3.3.1 Classes of Keys and Protection Requirements

3.3.2 Other Keying Material and Its Protection

Cryptographic Keys and Other Keying Material

Several different classes of keys are used by the Approved algorithms. Many of these keys are associated with other keying material. Each requires various degrees of protection.

Classes of Keys & Protection Requirements:

- Several different classes of keys, grouped according to function and according to their useful life span (life cycle), are defined.
- The life cycle of each type of key is further discussed in Section 4.
- Table 1 provides a summary of the protection requirements for these keys during distribution and storage.
- Methods for providing the necessary protection are discussed in Section 4.

Key Types

Signing Keys
Signature Verification Keys
Secret Authentication Keys
Private Authentication Keys
Public Authentication Keys
Long Term Data Encryption Keys
Short Term Data Encryption Keys
Random Number Generation Keys
Key Encrypting Keys Used for Key Wrapping
Master Keys used for Key Derivation
Keys Derived From a Master Key
Key Transport Private Keys
Key Transport Public Keys
Static Key Agreement Private Keys
Static Key Agreement Public Keys
Ephemeral Key Agreement Private Keys
Ephemeral Key Agreement Public Keys
Secret Authorization Keys
Private Authorization Keys
Public Authorization Keys

Key Types

- Note that we've tried to include a comprehensive list of key types.
- We solicit discussion concerning the appropriateness and/or actual utility of key types during the detailed discussion session.
- Also, are there key types other than those listed?
- We also solicit discussion of whether or not we should include use of keys for authorization purposes.
- No such use is identified in a FIPS, though a FIPS algorithm might be used for authorization.

Cryptographic Keys and Other Keying Material

3.3.1 Classes of Keys and Protection Requirements

3.3.2 Other Keying Material and Its Protection

Other Keying Material and Its Protection

•Other information used in conjunction with cryptographic algorithms also requires protection.

• The following viewgraph provides a summary of protection requirements during distribution and storage.

•Methods for providing protection are covered in Section 4.

Table 1:
Protection Requirements for Key Classes

	Confiden-tiality	Integrity	Long Term Availability	Associated with usage or application	Association with owner/other entity	Associated with other info.	Validation	
Signing keys	X	X		X		Domain parameters; signature verification key		
Signature verification keys		X	X	X	X	Domain parameters; signing key	For association with private key	
Secret authentication keys	X	X	X	X	X	Authenticated data		
Private authentication key	X	X		X		Public authentication key		
Public authentication key		X	X	X	X	Authenticated data; private authentication key	For association with private key	
Long term data encryption keys	X	X	X	X	X	Encrypted data		
Short term data encryption keys	X	X						
RNG keys	X	X		X				
Key encrypting key used for key wrapping	X	X	X	X	X	Encrypted keys		
Master key used for key derivation	X	X	X	X	X	Derived keys		
Keys derived from a Master Key	X?	X	X?	X?	X?	Master key and protected data		
Key transport private keys	X	X		X?		Encrypted keys; key transport public key		

This will probably be multiple screens.

Table 1 (Contd.): Protection Requirements for Key Classes

	Confiden-tiality	Integrity	Long Term Availability	Associated with usage or application	Association with owner/other entity	Associated with other info.	Validation	
Key transport public keys		X	X		X	Key transport private key	X	
Static key agreement private keys	X	X	X	X?		Domain parameters; static key agreement public key		
Static key agreement public keys		X	X	X?	X	Domain parameters; static key agreement private key	X	
Ephemeral key agreement private keys	X	X						
Ephemeral key agreement public keys		X					X	
Secret authorization key	X	X		X	X			
Private authorization key	X	X		X		Public authorization key		
Public authorization key		X		X	X	Private authorization key		
Domain parameters		X	X?	X		Private and public keys	X	
Initialization vectors	?	X	X			Protected data		
Shared secrets	X	X	?	X	X	?		
Seeds	X?			X?		Generated data?		
Intermediate results	X			X		Process data		

This will probably be multiple screens.

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

Key Management Lifecycle

Cryptographic key management encompasses the entire life cycle of cryptographic keys and other keying material. Basic key management guidance is provided in NIST SP800-21.

A single 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 have been deleted.

The viewgraph lists the document subsections that discuss the various stages of key management.

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

Key Management Lifecycle

The list shown on the viewgraph was based on [HAC].

During the detailed discussion period, we'll want to discuss whether we need all of the stages listed.

•During User Registration, an entity becomes an authorized user of a security domain. We might provide guidance concerning user authentication and presentation of identity credentials.

•System initialization involves setting up/configuring a system for secure operation. User initialization consists of an entity initializing its cryptographic application . Here, we might include user IDs, passwords, and the validation of system users. We could also include installation of keys at a CA, trust parameters, policy documentation, trusted parties, and algorithm preferences.

•The keying material installation section will contain guidance on: 1) initial installation of keying material, 2) issues related to the installation of additional keying material, 3) issues related to replacing existing keying material not covered in the key update section, and 4) issues related to keying material installation during key recovery not covered in the General Key Management Guidance section.

Key Establishment

4.4.1 Generation and Distribution of Public/Private Key Pairs
- Static Public Keys
- Ephimeral Public Keys
- Centrally Generated Private Keys

4.4.2 Generation and Distribution of Symmetric Keys
- Key Generation
- Key Distribution
- Key Agreement

4.4.3 Generation and Distribution of Other Keying Material
- Domain Parameters
- Initialization Vectors
- Shared Secrets
- Seeds
- Intermediate Results

Key Establishment

Key establishment includes the generation and sharing of cryptographic keys and other keying material between entities.

Generation and Distribution of Public/Private Key Pairs:

Public/private key pairs are used with digital signature and key establishment algorithms. They should be generated in accordance with the mathematical specifications of the appropriate Approved standard.

Private signing, authentication and authorization keys should not be distributed to other entities.

The distribution of public keys and of private keys other than signing, authentication and authorization keys is dependent on the type of key and whether it is static or ephemeral.

With respect to distribution of static public keys, the discussion assumes that the true owner of the key (the owner of the public key/private key pair) is known. This doesn't treat the use of anonymous public keys. The section on distribution of centrally generated private keys, includes distribution of key transport private keys and static key agreement private keys. Additions may need to be made to the assurances included in the key distribution section.

Key Establishment

4.4.1 Generation and Distribution of Public/Private Key Pairs
- Static Public Keys
- Ephimeral Public Keys
- Centrally Generated Private Keys

4.4.2 Generation and Distribution of Symmetric Keys
- Key Generation
- Key Distribution
- Key Agreement

4.4.3 Generation and Distribution of Other Keying Material
- Domain Parameters
- Initialization Vectors
- Shared Secrets
- Seeds
- Intermediate Results

Key Establishment

Symmetric keys may be:

•generated and subsequently distributed either manually, using a public key transport mechanism, or using a previously distributed or agreed upon key encrypting key, or

•determined using a key agreement scheme (i.e., the generation and distribution are accomplished with one process).

The symmetric keys used for the encryption/decryption of data or other keys and for the computation of MACs must be determined by an Approved method. The keys should be randomly generated and (optionally) distributed (transported) to another party, or may be determined by a key agreement mechanism.

The key agreement section assumes that each entity in the key establishment process knows the correct identity of any other entities. This ignores anonymous entities.

Keys are usually generated in conjunction with or are used with other keying material. This includes domain parameters, IVs, shared secrets, seeds, and intermediate results.

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

Key Registration: Keying material is bound to information or attributes associated with a particular entity. This includes the identity of the entity associated with the key, but may also include authorization information or specify the level of trust. The entity is typically a participant in a key management infrastructure, such as PKI or Kerberos. The binding is performed by a trusted third party, such as a PKI certification authority or a Kerberos realm server.

We'll want to discuss coming up with additional guidance regarding what information or attributes need to be bound to keying material, how the binding is performed by what entity(ies), and at what points in the lifecycle binding is required.

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 (i.e., the expiration date).

We need to discuss adding guidance for the various types of keys and specification that keys should be used for only their specifically intended/authorized purpose.

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

Keying material should be stored depending on its type, protection requirements and lifecycle stage. When the keying material is required for operational use, the keying material is acquired from operational storage when not present in active memory. If the keying material in active memory or operational storage is lost or corrupted, the keying material may be recovered from backup storage, providing that the keying material has been backed up. After the end of a key's cryptoperiod, keying material may be recovered from archival storage, providing that the keying material has been archived. Secret and Private keys require confidentiality protection, and all keys require integrity protection, protection against misuse. .A symmetric 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 share(s) the key. Public keys must be correctly associated (bound) with the owner of the public/private key pair. Long term availability of material must often be maintained. Finally, an association may need to be maintained between protected information and the key (or the associated key) that protected that information.

We may need to discuss whether guidance is required regarding labels that associate keys with usage or application.

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

Operational Storage: Keying material may need to be stored for normal cryptographic operations during the cryptoperiod of the key.

Back-up Storage:: The backup of keying material 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. Table 2 provides guidance about the backup of each type of keying material; however, the final determination for backup should be made based on the application in which the keying material is used.

Table 2:
Back-up 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.

This will probably be multiple screens.

Table 2 (Contd.): Back-up 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

This will probably be multiple screens.

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

Key Archive Storage

A key management archive is a repository containing keying material of historical interest. Not all keying material needs to be archived. Archived keying material may be either static (i.e., never changing) or may need to be re-encrypted under a new archive encryption key. Archived data should be stored separately from active data. Multiple copies of the archived data should be available and stored separately from each other. When no longer required, the keying material should be destroyed in accordance with Section 4.10.

Archived keying material requires confidentiality and/or integrity protection.

With respect to key archive storage, we may also need to discuss random access protection vs database protection. (Do we need to read the entire database to retrieve a specific key?) This is a key privacy issue.

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

Key Update: Prior to or at the end of a key's cryptoperiod, it needs to be replaced by a new key. A key may be replaced by rekeying, whereby a different key is established that does not depend (mathematically) on the key being replaced. Rekeying may use the key establishment methods discussed in Section 4.4. Alternatively, the key may be replaced by a key update method, (I.e., the new key is derived from the old key or the master key). [Note: Need more guidance, including discussions about limiting the number of updates before rekeying, and the evaluation of update procedures.]

Key Recovery (The process of retrieving the keying material from backup or archive storage): Key recovery is a broad term that may apply to several different key recovery techniques. The information required to recover a key may be different for each application or each key recovery technique. The term "Key Recovery Information" (KRI) is used to refer to the aggregate of information needed to recover the key. The KRI includes the key to be recovered (perhaps encrypted or divided into multiple components) and other cryptographic data (e.g., IVs), the time when the key was created, the identity of the owner of the key (the individual, application or organization who created the key or who own the data protected by that key) and any conditions that must be met by a requestor to be able to recover the keying material.

A list of general requirements for a key recovery system should be reviewed during the detailed discussion session.

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

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

While it may be desirable to destroy all copies of a public key in many cases, it is not generally possible to guarantee that this is actually done. Retention of the public key is not currently assumed to introduce a security problem.

Key Revocation:: It may be necessary to remove keying material from use prior to the end of its normal cryptoperiod for reasons that include key compromise and removal of an entity form an organization. This is accomplished by notifying all entities that may be using the revoked keying material that the material should no longer be used. The notification should include a complete identification of the keying material, the date and time of revocation and the reason for revocation (e.g., key compromised). Based on the revocation information provided, the other entities can make a determination of how they would treat information protected by the revoked keying material.

General Key Management Guidance

5.1 Key Management Policy

5.2 Guidance for Cryptographic Algorithm and Key Size Selection

5.3 Key Establishment Schemes

General Key Management Guidance

General Key Management Guidance includes:

•Key Management Policy,

•Guidance for Cryptographic Algorithm and Key Size Selection, and

•Key Establishment Schemes

Key Management Policy

5.1.1 Key Management Practices Statement
5.1.2 Key Usage
5.1.3 Cryptoperiods
5.1.4 Domain Parameter Validation and Public Key Validation
5.1.5 Compromise of Keys and Other Keying Material
5.1.6 Accountability
5.1.7 Audit
5.1.8 Key Recovery
Considerations
Policy

Key Management Policy

U. S. Government agencies that use cryptography are responsible for defining the Key Management Policy (KMP) that governs the lifecycle for the cryptographic keys as specified in Section 4. A Key Management Practices Statement (KMPS) is then developed based on the KMP and the actual applications supported. The KMPS should specify how key management procedures, and techniques are used to enforce the KMP. For example, a key management policy statement might be that secret and private keys must be protected from unauthorized disclosure. The corresponding key management practices statement might state that secret and private keys must be either encrypted or physically protected.

It is not intended that the Key Management Policy or the Key Management Practices Statement should create significant additional documentation development requirements for an agency. The intent is to provide guidance regarding what should be included in documents that are currently required.

A cryptographic key should be used for only one purpose. E.g., a given symmetric key may be used for data encryption OR the key encryption OR the creation of a MAC OR the generation of random numbers, but should not be used for more than one of these functions.

We'll need to discuss provision of guidance on specific cryptoperiod lengths

Considerations for key recovery might also include examples advising whether or not key recovery is appropriate in particular scenarios and who may request recovery of a key.

Key Management Policy

5.1.1 Key Management Practices Statement
5.1.2 Key Usage
5.1.3 Cryptoperiods
5.1.4 Domain Parameter Validation and Public Key Validation
5.1.5 Compromise of Keys and Other Keying Material
5.1.6 Accountability
5.1.7 Audit
5.1.8 Key Recovery
Considerations
Policy

Key Management Policy

Cryptoperiod: A cryptoperiod is the time span during which a specific key is authorized for use by legitimate entities, or the keys for a given system may remain in effect. A suitably defined cryptoperiod 1) limits the amount of information protected by a given key that is available for cryptanalysis, 2) limits the amount of exposure if a single key is compromised, 3) limits the use of a particular algorithm to its estimated effective lifetime, and 4) may limit the amount of time available for cryptanalytic attacks to be useful. Trade-offs associated with the determination of cryptoperiods involve the risk and consequences of exposure. We'll need to provide guidance on specific period lengths

Domain Parameter and Public Key Validation: The domain parameters used for DSA, ECDSA and key agreement algorithms should be generated by a trusted party, OR validated by a trusted party or the participating entities themselves. Public signature verification keys must be validated to ensure that they are associated with a private signing key known by the owner of the public key. The public keys used for DSA, ECDSA and the key agreement algorithms must also be validated. [Note: Need to address the validation of public key transport keys.]

Key Management Policy

5.1.1 Key Management Practices Statement
5.1.2 Key Usage
5.1.3 Cryptoperiods
5.1.4 Domain Parameter Validation and Public Key Validation
5.1.5 Compromise of Keys and Other Keying Material
5.1.6 Accountability
5.1.7 Audit
5.1.8 Key Recovery
Considerations
Policy

Key Management Policy

Key compromise occurs when the protective mechanisms for the key fail (e.g., the confidentiality, integrity or association of the key to its owner fail), and the key can no longer be trusted to provide the required security. When a key is compromised, all use of the key should cease, the compromised key should be revoked, and all entities using or relying on that key should be notified. Limiting the cryptoperiod of the key, using different keys for different purposes, and limiting the amount of information protected by a single key limit the amount of material exposed if a key is compromised.

Accountability involves the identification of those who have control of, and access to, cryptographic keys throughout their lifecycles. Certain principles are useful in enforcing key accountability. These principles may not apply to all systems or types of keys. Some apply to longer-term keys that are controlled by humans. The principles include 1) uniquely identifying keys, 2) identifying key users, 3) identifying dates and times of key use along and the data that is protected, 4) identifying other keys that are protected by a secret or private key, 5) limiting the amount of time a secret or private key is in plaintext form, 6) restricting plaintext secret and private keys to approved key,7) preventing humans from viewing plaintext secret and private keys, and 8) destroying keys as soon as they are no longer needed.

Key Management Policy

5.1.1 Key Management Practices Statement
5.1.2 Key Usage
5.1.3 Cryptoperiods
5.1.4 Domain Parameter Validation and Public Key Validation
5.1.5 Compromise of Keys and Other Keying Material
5.1.6 Accountability
5.1.7 Audit
5.1.8 Key Recovery
Considerations
Policy

Key Management Policy

Audit: Key management systems should be periodically audited to ensure that practices continue to support the Key Management Policy. The protective mechanisms should be reassessed as to the level of security that they provide and are expected to provide in the future. New technical developments and attacks should be taken into consideration. Most importantly, the actions of the humans that use, operate and maintain the system should be reviewed to verify that they continue to follow established security procedures.

Key Recovery: The process of retrieving the keying material from backup or archive storage is called key recovery and is motivated by a need to recover or ascertain the validity of cryptographically protected information. The decision as to whether key recovery is required should be made on a case by case basis. Considerations for key recovery might also include examples advising whether or not key recovery is appropriate in particular scenarios and who may request recovery of a key. An organization using cryptographic protection should develop a policy that addresses the continued accessibility of protected information. When it is determined that recovery of keying material is required, a Key Recovery System needs to be defined. For each key type, we may need to discuss risk assessment regarding whether key replacement is preferable to key recovery.

Guidance for Cryptographic Algorithm and Key Size Selection

5.2.1 Equivalent Algorithm Strength

5.2.2 Defining Appropriate Algorithm Strengths

5.2.3 Transitioning to New Algorithms and Key Sizes

Guidance for Crypto Algorithm and Key Size Selection

Cryptographic algorithms provide different “strengths” of security, depending on the algorithm and the key size used. In this document, two algorithms are considered to be of equivalent strength for the given key sizes if the amount of time needed to “break the algorithms” or determine the keys (with the given key sizes) is the same. The strength of an algorithm for a given key size is traditionally described in terms of the amount of time it takes to try all keys for a symmetric algorithm that has no short cut attacks (i.e., exhaust the key space). In this document, this equivalence is phased as providing “X bits of security”. [Note: Other metrics for algorithm equivalence exist.]

The user should be aware that the recommended key size equivalencies are based on assessments made as of the publication of this document. Periodic reviews will be performed to determine whether the stated equivalencies need revision.

Table 3 provides equivalence guidelines for the Approved algorithms.

Table 3:
Equivalent Algorithm Strengths

Bits of security	Symmetric key algs.	Hash algs.	Discrete Logs (DSA, DH, MQV)	RSA	Elliptic Curves
80		SHA-1	$L = 1024$ $N = 160$	$k = 1024$	$f = 160$
112	TDES		$L = 2048$ $N = 224$	$k = 2048$	$f = 224$
128	AES-128	SHA-256	$L = 3072$ $N = 256$	$k = 3072$	$f = 256$
192	AES-192	SHA-384	$L = 7680$ $N = 384$	$k = 7680$	$f = 384$
256	AES-256	SHA-512	$L = 15360$ $N = 512$	$k = 15360$	$f = 512$

Guidance for Cryptographic Algorithm and Key Size Selection

5.2.1 Equivalent Algorithm Strength

5.2.2 Defining Appropriate Algorithm Strengths

5.2.3 Transitioning to New Algorithms and Key Sizes

Defining Appropriate Algorithm Suites

Many applications require the use of several different algorithms. Some algorithms are inherently more efficient because of their design (e.g., AES has been designed to be more efficient than Triple DES).

In many cases, a variety of key sizes may be available. For some of the algorithms, the use of larger key sizes than are required may impact operations, (e.g., larger keys may take longer to generate or longer to process the data). However, the use of key sizes that are too small may provide inadequate security.

Table 4 provides recommendations that may be used to select an appropriate suite of algorithms and key sizes for Federal Government unclassified applications. A minimum of eighty bits of security are considered adequate for most applications until 2015. Thereafter, it is recommended that at least 112 bits of security be used.

We may wish to discuss inclusion of a list of algorithm combinations and description of the security implications of various combinations.

Examples could also be given regarding appropriate algorithm and key size selections for representative security requirements.

Table 4: Recommended Algorithms and Minimum Key Sizes

Years	**Symmetric key algs. (Encryption & MAC)**	**Hash Alg.**	**HMAC**	**DSA**	**RSA**	**Elliptic Curves**
Present - 2015	TDES AES-128 AES-192 AES-256	SHA-1 SHA-256 SHA-384 SHA-512	SHA-1 (≥80 bit key) SHA-256 (≥128 bit key) SHA-384 (≥192 bit key) SHA-512 (≥256 bit key)	Min.: L = 1024; N =160	Min.: k=1024	Min.: f=160
2016 and beyond	TDES AES-128 AES-192 AES-256	SHA-256 SHA-384 SHA-512	SHA-256 (≥128 bit key) SHA-384 (≥192 bit key) SHA-512 (≥256 bit key)	Min.: L = 2048 N = 224	Min.: k=2048	Min.: f=224

Guidance for Cryptographic Algorithm and Key Size Selection

5.2.1 Equivalent Algorithm Strength

5.2.2 Defining Appropriate Algorithm Strengths

5.2.3 Transitioning to New Algorithms and Key Sizes

Transitioning to New Algorithms/Key Sizes

Many legacy applications currently use algorithms and key sizes not specified in Table 4. Information protected by these algorithms that must be retained should be updated or suitably archived to ensure the continued protection of that information. Information protected by an algorithm or key size that is considered to be adequate until 2015, but not beyond, should be updated or archived if protection is to be afforded after 2015.

It may not bepossible to protect encrypted information once the security life of the encryption algorithm, or its key size, has expired (e.g., information that was encrypted using legacy applications). One assumes that encrypted data could have been collected and retained by unauthorized parties. In the future, when the cryptographic system becomes vulnerable, the unauthorized parties may attempt to decrypt the information. Even though the system has been replaced (e.g., by a different algorithm or key size), the previously encrypted information is vulnerable. Thus, when using Table 4 to select the appropriate key size for an encryption algorithm, it is very important to take the expected security life of the data into consideration.

Need to provide guidance about how to transition, including how to extend the protection on data that has been protected using “no longer secure” methods.

General Key Management Guidance

5.1 Key Management Policy

5.2 Guidance for Cryptographic Algorithm and Key Size Selection

5.3 Key Establishment Schemes

Key Establishment Schemes

Schemes for the establishment of keying material are provided in a future FIPS. Methods have been provided for both key agreement and key transport. Key agreement schemes use asymmetric (public key) techniques. Key transport schemes use either symmetric (secret key) or asymmetric (public key) techniques. Key transport schemes using symmetric techniques are commonly known as key wrapping algorithms.

Discussions will be included on using the schemes in the future FIPS .

Key Management Guidance
Selected Infrastructures

6.1 Public Key Infrastructure

6.2 Kerberos

Key Management Guidance
Selected Infrastructures

We're looking at providing key management guidance for PKI and Kerberos. Suggestions are solicited regarding both infrastructure selection and guidance content.

Key Management Guidance
Selected Protocols

7.1 S/MIME

7.2 TLS/SSL

7.2.1 Version
7.2.2 Cipher Suite Selection
7.2.3 Public Key Certificates for TLS

Key Management Guidance
Selected Protocols

We’re looking at providing key management guidance for S/MIME and TLS/SSL Suggestions are solicited regarding both protocol selection and guidance content.

Key Management Guidance
Selected Applications

8.1 Encrypted File Storage

8.2 ???

Key Management Guidance
Selected Applications

We're looking at providing key management guidance for encrypted file storage. We're hoping for suggestions regarding other applications.

Appendix A

Cryptoperiods for
Signing Key Pairs

Appendices

So far, the only appendices we've identified a need for are detail regarding cryptoperiods for signing key pairs and a list of references. Any other suggestions?

Appendix X

References