NIST
National Institute of
Standards and Technology
Technology Administration
U.S. Department of Commerce

Special Publication 800-111
(Draft)

Guide to Storage Encryption Technologies for End User Devices (Draft)

Recommendations of the National Institute of Standards and Technology

Karen Scarfone
Murugiah Souppaya
Matt Sexton

NIST Special Publication 800-111 (Draft)

Guide to Storage Encryption Technologies for End User Devices (Draft)

Recommendations of the National Institute of Standards and Technology

Karen Scarfone
Murugiah Souppaya
Matt Sexton

COMPUTER SECURITY

Computer Security Division
Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8930

August 2007

U.S. Department of Commerce

Carlos M. Gutierrez, Secretary

Technology Administration

Robert C. Cresanti, Under Secretary of Commerce for Technology

National Institute of Standards and Technology

William Jeffrey, Director

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analysis to advance the development and productive use of information technology. ITL's responsibilities include the development of technical, physical, administrative, and management standards and guidelines for the cost-effective security and privacy of sensitive unclassified information in Federal computer systems. This Special Publication 800-series reports on ITL's research, guidance, and outreach efforts in computer security and its collaborative activities with industry, government, and academic organizations.

National Institute of Standards and Technology Special Publication 800-111 (Draft)
Natl. Inst. Stand. Technol. Spec. Publ. 800-111, 37 pages (Aug. 2007)

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

Acknowledgements

The authors, Karen Scarfone and Murugiah Souppaya of the National Institute of Standards and Technology (NIST), and Matt Sexton of Booz Allen Hamilton, wish to thank their colleagues who reviewed drafts of this document and contributed to its technical content. The authors would like to acknowledge Tim Grance and Bill Burr of NIST, and Angela Orebaugh, Manuel Villar, Mike Zeberlein, and Mike Zirkle of Booz Allen Hamilton, for their keen and insightful assistance throughout the development of the document. Additional acknowledgements will be added to the final version of the publication.

Note to Reviewers

The initial scope of this publication is intentionally tightly-focused so that it could be available more quickly as a resource for agencies planning and implementing solutions for protecting personally identifiable information on mobile devices. NIST is considering expanding this publication to address storage encryption for devices other than end user devices, including servers, storage area networks, and backup tapes. The expanded publication might also cover storage encryption-related policy, storage encryption infrastructure needs, and strategies for identifying where sensitive information is stored and determining what types of protections it needs. The authors welcome suggestions on which aspects of storage encryption would be helpful to include in the expanded publication. NIST thanks the reviewers in advance for sharing their expertise and valuable time to perform this public service.

Table of Contents

List of Tables

Executive Summary

In today's computing environment, there are many threats to the confidentiality of information stored on end user devices, such as personal computers, consumer devices (e.g., personal digital assistant, smart phone), and removable storage media (e.g., USB flash drive, memory card, external hard drive, writeable CD or DVD). Some threats are unintentional, such as human error, while others are intentional. These threats are posed by people with many different motivations, including causing mischief and disruption and committing identity theft and other fraud. A common threat against end user devices is loss or theft. Someone with physical access to a device has many options for attempting to view or copy the information stored on the device. Another concern is insider attacks, such as an employee attempting to access sensitive information stored on another employee's computer. Malware is another common threat; it can give attackers unauthorized access to a device, transfer information from the device to an attacker's system, and perform other actions that jeopardize the confidentiality of the information on a device.

Many threats against end user devices could cause information stored on the devices to be accessed by unauthorized parties. To prevent such disclosures of information, particularly of personally identifiable information (PII) and other sensitive data, the information needs to be secured. Securing other components of end user devices, such as operating systems, is also necessary, but in many cases additional measures are needed to secure the stored information. This publication explains the basics of storage security, which is the process of allowing only authorized parties to access and use stored information. The primary security controls for restricting access to sensitive information stored on end user devices are encryption and authentication. Encryption can be applied granularly, such as to an individual file containing sensitive information, or broadly, such as encrypting all storage. The appropriate encryption solution for a particular situation depends primarily upon the type of storage, the amount of information that needs to be protected, and the threats that need to be mitigated. This publication describes three types of solutions—full disk encryption, volume and virtual disk encryption, and file/folder encryption—and makes recommendations for implementing and using each type. This publication also includes several use case examples, which illustrate that there are multiple ways to meet most storage encryption needs. When selecting a solution type, organizations should consider the range of solutions, and not just the solution type that is most commonly used.

Implementing the following recommendations should facilitate more efficient and effective storage encryption solution design, implementation, and management for Federal departments and agencies.

When selecting a storage encryption technology, organizations should consider solutions that use existing system features and infrastructure.

There are many factors for organizations to consider when selecting storage encryption solutions, such as the platforms they support, the data they protect, and the threats they mitigate. Some solutions involve deploying various servers and installing software on the devices to be protected, while other solutions can use existing servers, as well as software built in to the devices to be protected. Generally, the more extensive the changes are to the infrastructure and devices, the more likely it is that the storage encryption solution will cause a loss of functionality or other problems with the devices. Solutions that require extensive changes to the infrastructure and end user devices should generally be used only when other solutions cannot meet the organization's needs.

Organizations should use centralized management for all deployments of storage encryption except for standalone deployments and very small-scale deployments.

Centralized management is recommended for most storage encryption deployments because of its effectiveness and efficiency for policy verification and enforcement, key management, authenticator

management, data recovery, and other management tasks. Centralized management can also automate deployment and configuration of storage encryption software to end user devices, distribution and installation of updates, collection and review of logs, and recovery of information from local failures.

Organizations should ensure that all cryptographic keys used in a storage encryption solution are secured and managed properly to support the security of the solution.

Storage encryption technologies use one or more cryptographic keys to encrypt and decrypt the data that they protect. If a key is lost or damaged, it may not be possible to recover the encrypted data from the computer. Therefore, organizations should perform extensive planning of key management processes, procedures, and technologies before implementing storage encryption technologies. This planning should include all aspects of key management, including key generation, use, storage, and destruction. Organizations should carefully consider how key management practices can support the recovery of encrypted data if a key is inadvertently destroyed or otherwise becomes unavailable. Organizations planning on encrypting removable media also need to consider how changing keys will affect access to encrypted storage on removable media and develop feasible solutions, such as retaining the previous keys in case they are needed.

Organizations should select appropriate user authenticators for storage encryption solutions.

Storage encryption solutions require users to authenticate successfully before accessing the information that has been encrypted. Common authentication mechanisms are passwords, personal identification numbers, cryptographic tokens, biometrics, and smart cards. Organizations often want to leverage existing enterprise authentication solutions (e.g., Active Directory, public key infrastructure [PKI]) instead of adding another authenticator for users. Generally, this is acceptable if two-factor authentication is being used because of its strength. Using the same single-factor authenticator for multiple purposes, such as operating system (OS) authentication and storage encryption authentication, significantly weakens the protection that authentication provides. For example, an attacker who learns a single password could gain full access to the device's information. Organizations should carefully consider the security implications of using the same single-factor authenticator for multiple purposes. In particular, organizations should not use email passwords and other passwords sometimes transmitted in plaintext as single-factor authenticators for storage encryption.

Organizations should implement measures that support and complement storage encryption implementations for end user devices.

Storage encryption by itself cannot provide adequate security for stored information; additional security controls are needed. Organizations should select and deploy the necessary controls based on FIPS 199's categories for the potential impact of a security breach involving a particular system and NIST Special Publication 800-53's recommendations for minimum management, operational, and technical security controls. Examples of supporting controls are as follows:

- Revising organizational policies as needed to incorporate appropriate usage of the storage encryption solution.

- Securing and maintaining end user devices properly, which should reduce the risk of compromise or misuse. This includes securing device operating systems, applications, and communications, and physically securing devices.

- Making users aware of their responsibilities for storage encryption, such as encrypting sensitive files or physically protecting mobile devices and removable media.

1. Introduction

1.1 Authority

The National Institute of Standards and Technology (NIST) developed this document in furtherance of its statutory responsibilities under the Federal Information Security Management Act (FISMA) of 2002, Public Law 107-347.

NIST is responsible for developing standards and guidelines, including minimum requirements, for providing adequate information security for all agency operations and assets; but such standards and guidelines shall not apply to national security systems. This guideline is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130, Section 8b(3), "Securing Agency Information Systems," as analyzed in A-130, Appendix IV: Analysis of Key Sections. Supplemental information is provided in A-130, Appendix III.

This guideline has been prepared for use by Federal agencies. It may be used by nongovernmental organizations on a voluntary basis and is not subject to copyright, though attribution is desired.

Nothing in this document should be taken to contradict standards and guidelines made mandatory and binding on Federal agencies by the Secretary of Commerce under statutory authority, nor should these guidelines be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the OMB, or any other Federal official.

1.2 Purpose and Scope

The purpose of this document is to assist organizations in understanding storage encryption technologies for end user devices and in planning, implementing, and maintaining storage encryption solutions. This document provides practical, real-world guidance for three classes of storage encryption techniques: full disk encryption, volume and virtual disk encryption, and file/folder encryption. It also discusses important security elements of a storage encryption deployment, including cryptographic key management and authentication. It only discusses the encryption of data at rest (storage), and does not address the encryption of data in motion (transmission).

1.3 Audience

This document has been created for information security program managers and staff, system administrators, and others who are responsible for selecting, deploying, managing, and maintaining storage encryption technologies for end user devices. This document does not assume that the reader has previous experience with any storage encryption technologies, but it does assume that the reader has experience with information security.

1.4 Document Structure

The remainder of this document is organized into three major sections.

- Section 2 provides an overview of the basic concepts of storage encryption for end user devices.

- Section 3 describes the most commonly used categories of storage encryption technologies for end user devices, and explains the types of protection they provide.

- Section 4 discusses the process of planning and implementing storage encryption technologies for end user devices. It includes a detailed discussion of the importance of cryptography and authentication to a storage encryption solution.

The document also contains several appendices with supporting material.

- Appendix A describes alternatives to encrypting storage on end user devices.
- Appendices B and C contain a glossary and acronym list, respectively.
- Appendix D lists online tools and resources that may be useful references for gaining a better understanding of storage encryption for end user devices.

2. Storage Security Overview

An *end user device* is a personal computer (desktop or laptop), consumer device (e.g., personal digital assistant [PDA], smart phone), or removable storage media (e.g., USB flash drive, memory card, external hard drive, writeable CD or DVD) that can store information.[1] *Storage security* is the process of allowing only authorized parties to access and use stored information. This section introduces the basic concepts of storage security for end user devices.[2]

2.1 File Storage Basics

A *file* is a collection of information logically grouped into a single entity and referenced by a unique name, such as a *filename*. On end user devices, there are typically two types of files: data files, such as text documents, spreadsheets, images, and videos, and system files, such as operating system and application binaries and libraries. A *filesystem* defines the way that files are named, stored, organized, and accessed. *Directories*, also known as *folders*, are organizational structures used by filesystems to group files. Another feature of a filesystem is *metadata*, which is information regarding the files and folders themselves, such as file and folder names, creation dates and times, and sizes.

Filesystems are designed to store folders, system and data files, and metadata on storage media. However, filesystems may also hold *residual data*, which is data from deleted files or earlier versions of existing files. The following items describe common forms of residual data:

- **Deleted Files**. When a file is deleted, it is typically not erased from the media; instead, the information in the directory's data structure that points to the location of the file is marked as deleted. This means that the file is still stored on the media but is no longer enumerated by the operating system (OS). The OS considers this to be free space and can overwrite any portion of or the entire deleted file at any time.
- **Slack Space**. Filesystems store files in chunks known as file allocation units. Even if a file requires less space than the file allocation unit size, an entire file allocation unit is still reserved for the file. For example, if the file allocation unit size is 32 kilobytes (KB) and a file is only 7 KB, the entire 32 KB is still allocated to the file, but only 7 KB is used, resulting in 25 KB of unused space. This unused space is referred to as *file slack space,* and it may hold residual data such as portions of deleted files.
- **Free Space**. *Free space* is the area on media that is not currently allocated to a partition. This often includes space on the media where files may have resided at one point but have since been deleted. The free space may still contain pieces of data.

Before media can be used to store files, the media must usually be partitioned and formatted into logical volumes. *Partitioning* is the act of logically dividing a media into portions that function as physically separate units. A *logical volume* is a partition or a collection of partitions acting as a single entity that has been formatted with a filesystem. Some media types can contain at most one partition (and consequently, one logical volume).

[1] This publication only addresses technologies for encrypting files stored on end user devices. Information on storage security for servers, storage area networks, enterprise backup tapes, and other devices is outside the scope of this publication.

[2] Storage security is only one component of data security, which includes network, host, and application security, and also addresses how data may be used after it is accessed. All elements of data security other than storage security, such as encrypting data in motion (e.g., network communications), are outside the scope of this publication.

2.2 The Need for Storage Security

In today's computing environment, there are many threats to the confidentiality of information stored on end user devices. Some threats are unintentional, such as human error, while others are intentional. These threats are posed by people with many different motivations, including causing mischief and disruption and committing identity theft and other fraud. One of the most common threats is *malware*, also known as *malicious code*, which refers to a program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or OS. Types of malware threats include viruses, worms, malicious mobile code, Trojan horses, rootkits, and spyware. Malware can give attackers unauthorized access to a device, transfer information from the device to an attacker's system, and perform other actions that jeopardize the confidentiality of the information on a device. Another common threat against end user devices is loss or theft. Someone with physical access to a device has many options for attempting to view the information stored on the device. This is also a concern for insider attacks, such as an employee attempting to access sensitive information stored on another employee's computer. Another form of insider attack is a user attempting to access another user's files on a device that the two users share.

Many threats against end user devices could cause information stored on the devices to be accessed by unauthorized parties. To prevent such disclosures of information, particularly of personally identifiable information (PII)[3] and other sensitive data, the information needs to be secured. Securing other components of end user devices, such as OSs, is also necessary, but in many cases additional measures are needed to secure the stored information. For example, without these additional measures, an attacker that steals a device could use forensic tools and techniques to recover information directly from the storage media, circumventing the protections applied by the device's OS.

A number of laws and regulations compel organizations to ensure that sensitive information is protected appropriately. The following is a list of key regulations, standards, and guidelines that help define organizations' needs for storage security:[4]

- **Federal Information Security Management Act of 2002 (FISMA).** FISMA emphasizes the need for each Federal agency to develop, document, and implement an organization-wide program to provide information security for the information systems that support its operations and assets. NIST Special Publication (SP) 800-53, *Recommended Security Controls for Federal Information Systems*, was developed in support of FISMA.[5] NIST SP 800-53 is the primary source of recommended security controls for Federal agencies. It describes several controls related to storage security, such as controlling access through encryption of stored information, restricting access to mobile computing devices and information system media, and storing media in physically secure locations.

- **OMB Memorandum M-06-16.** OMB has issued a memorandum directly related to storage security. OMB M-06-16 addresses the protection of agency information that is either "accessed remotely or physically transported outside of the agency's secured, physical perimeter". It specifically requires

[3] OMB Memorandum 06-19, "Reporting Incidents Involving Personally Identifiable Information and Incorporating the Cost for Security in Agency Information Technology Investments", defines PII as "any information about an individual maintained by an agency, including, but not limited to, education, financial transactions, medical history, and criminal or employment history and information which can be used to distinguish or trace an individual's identity, such as their name, social security number, date and place of birth, mother's maiden name, biometric records, etc., including any other personal information which is linked or linkable to an individual." The full text of the memorandum is available at http://www.whitehouse.gov/omb/memoranda/fy2006/m-06-19.pdf.

[4] It is outside this publication's scope to explain which types of information organizations need to protect and how each type should be protected. NIST SP 800-60, *Guide for Mapping Types of Information and Information Systems to Security Categories*, discusses the identification of common types of information. It is at http://csrc.nist.gov/publications/nistpubs/.

[5] Copies of FISMA and NIST SP 800-53 are available at http://csrc.nist.gov/sec-cert/ca-library.html.

that agencies encrypt all data stored on mobile computing devices, such as laptops and PDAs, unless the data has been determined by the designated agency official to be non-sensitive.[6] Similar requirements are also included in OMB Memorandum M-07-16.[7]

- **Privacy Act of 1974.** The Privacy Act regulates the collection, use, maintenance, and dissemination of personal information about U.S. citizens or aliens lawfully admitted for permanent residence. It applies to records maintained by agencies in the executive branch of the government.

- **Gramm-Leach-Bliley Act (GLBA).** GLBA requires financial institutions to protect their customers' information against security threats. This includes ensuring "the security and confidentiality of customer records and information" and protecting "against unauthorized access to or use of such records or information".[8]

- **Health Insurance Portability and Accountability Act of 1996 (HIPAA).** HIPAA includes security standards for certain health information. NIST SP 800-66, *An Introductory Resource Guide for Implementing the Health Insurance Portability and Accountability Act (HIPAA) Security Rule*, lists HIPAA-related storage security needs.[9] For example, Section 4.14 of NIST SP 800-66 describes the need to encrypt and decrypt electronic protected health information (EPHI).

2.3 Security Controls for Storage

The primary security controls for restricting access to sensitive information stored on end user devices are encryption and authentication. Encryption can be applied granularly, such as to an individual file containing sensitive information, or broadly, such as encrypting all storage. The appropriate encryption solution for a particular situation depends primarily upon the type of storage, the amount of information that needs to be protected, and the threats that need to be mitigated. Section 3 discusses the most commonly used options in detail; Appendix A briefly discusses some additional options. Storage encryption solutions require users to authenticate successfully before accessing the information that has been encrypted. Common authentication mechanisms are passwords, personal identification numbers (PIN), cryptographic tokens, biometrics, and smart cards. The combination of encryption and authentication controls access to the stored information.

Organizations also need to consider the security of backups of stored information. Some organizations permit users to back up their local files to a centralized system, while other organizations recommend that their users perform local backups (e.g., burning CDs). In the latter case, organizations should ensure that the backups will be secured at least as well as the original source. This could be done with similar controls, such as encrypting the backups, or with different types of controls, such as storing backup tapes in a physically secured room within the organization's facilities.

Organizations should also implement other measures that support and complement storage encryption implementations. These measures help to ensure that storage encryption is implemented in an environment with the management, operational, and technical controls necessary to provide adequate security for the storage encryption implementation. Examples of supporting measures are as follows:

[6] The memorandum is available at http://www.whitehouse.gov/OMB/memoranda/fy2006/m06-16.pdf.

[7] M-07-16 is available at http://www.whitehouse.gov/omb/memoranda/fy2007/m07-16.pdf.

[8] More information on GLBA is available at http://www.ftc.gov/privacy/privacyinitiatives/glbact.html. A copy of GLBA can be downloaded from http://www.ftc.gov/privacy/privacyinitiatives/financial_rule_lr.html.

[9] HIPAA is available for download from http://www.hhs.gov/ocr/hipaa/. NIST SP 800-66 is located at http://csrc.nist.gov/publications/nistpubs/.

- Revise organizational policies as needed to incorporate appropriate usage of the storage encryption solution. Policies should provide the foundation for the planning and implementation of storage encryption.

- Ensure that end user devices are secured and maintained properly, which should reduce the risk of compromise or misuse. This includes securing device OSs, applications, and communications (e.g., encrypting wired and wireless network traffic) and physically securing devices, such as requiring that laptops be secured using cable locks when in hotels, conferences, and other locations where third parties could easily gain physical access to the devices. Physical security for devices is also an important consideration in home environments, such as preventing others within the house from using an organization-issued device by keeping the device in a locked room.

- Make users aware of their responsibilities for storage encryption, such as encrypting sensitive files or physically protecting mobile devices and removable media.

Organizations should select and deploy the necessary security controls based on existing guidelines. Federal Information Processing Standards (FIPS) 199 establishes three security categories—low, moderate, and high—based on the potential impact of a security breach involving a particular system.[10] NIST SP 800-53 provides recommendations for minimum management, operational, and technical security controls for information systems based on the FIPS 199 impact categories.[11] The recommendations in NIST SP 800-53 should be helpful to organizations in identifying controls that are needed to protect end user devices, which should be used in addition to the specific recommendations for storage encryption listed in this document.[12]

[10] FIPS 199, *Standards for Security Categorization of Federal Information and Information Systems*, is available at http://csrc.nist.gov/publications/fips/.

[11] NIST SP 800-53 Revision 1, *Recommended Security Controls for Federal Information Systems*, is available at http://csrc.nist.gov/publications/nistpubs/.

[12] In addition to securing the wireless networks, the wireless devices using the networks also need to be secured; however, an explanation of securing laptops, PDAs, and other wireless devices is outside the scope of this guide.

3. Storage Encryption Technologies

There are many technologies available for encrypting data stored on end user devices. This section describes the most commonly used technologies, discusses the protections provided by each type, and explains the ways in which the technologies are typically managed.

3.1 Common Types of Storage Encryption Technologies

This section provides a high-level overview of the most commonly used options for encrypting stored information: full disk encryption, volume and virtual disk encryption, and file/folder encryption.[13] It briefly defines each option and explains at a high level how it works.

3.1.1 Full Disk Encryption

Full disk encryption (FDE), also known as whole disk encryption, is the process of encrypting all the data on the hard drive used to boot a computer, including the computer's OS, and permitting access to the data only after successful authentication to the FDE product. Most FDE products are software-based, so this section focuses on explaining the capabilities and characteristics of software-based FDE solutions. Hardware-based solutions are discussed briefly at the end of this section.

FDE software works by redirecting a computer's *master boot record (MBR)*, which is a reserved sector on bootable media that determines which software (e.g., OS, utility) will be run when the computer boots from the media. Before FDE software is installed onto a computer, the MBR is usually pointing to the computer's primary OS. When FDE software is being used, the computer's MBR is redirected to a special pre-boot environment (PBE) that controls access to the computer.[14] This redirection is depicted in Figure 3-1. The PBE prompts the user to authenticate successfully, such as entering a user ID and password, before decrypting and booting the OS. This is known as *pre-boot authentication (PBA).*[15] Most FDE products support the use of both network-based authentication (e.g., Active Directory, PKI) and local authentication sources (e.g., locally stored, locally cached from network source) for PBA.

Once successful PBA occurs, the FDE software decrypts the boot sector for the OS, as depicted by the second arrow in Figure 3-1, and the boot loader in the boot sector starts to load the OS. As it loads, the FDE software decrypts the OS files (which are stored in the system volume) as needed, indicated in Figure 3-1 by the third arrow. Once the OS has finished booting, the user provides OS authentication and uses the computer normally. When the user needs to open encrypted files, save new files, or perform other operations involving the hard drive, the FDE software transparently decrypts and encrypts the necessary sectors[16] of the hard drive as needed.[17] This may marginally increase the time needed to open or save files, but the delay generally should only be noticeable for particularly large files. On an FDE-protected computer, users will typically notice a delay of at least a few seconds when booting the

[13] There are many ways in which information stored on end user devices can be encrypted. For example, an application that accesses sensitive information could be responsible for encrypting that information. Applications such as backup programs might also offer encryption options. Another method for protecting files is digital rights management (DRM) software.

[14] Some FDE implementations verify the integrity of the boot components, including the MBR, before proceeding. Figure 3-1 depicts the PBE as being stored in the space between the MBR and the boot sector; although many software-based FDE products store the PBE there, this is not required, and some products store the PBE elsewhere.

[15] Most software-based FDE products use PBA. Other products require the user to authenticate after the OS has booted, which provides weaker protection than PBA. This publication assumes that FDE implementations are configured to require PBA. PBA can be performed with stronger authenticators than user ID and password; see Section 4.2 for additional information.

[16] A *sector* is the smallest logical component of a hard drive that can be read or written. Many hard drives have 512-byte sectors. Even if a single byte of data needs to be accessed, the hard drive will read the entire sector containing that data.

[17] Figure 3-1 does not show data volumes and other volumes that might be on a hard drive; such volumes would also be encrypted by an FDE solution.

computer or shutting it down. Delays may also occur when using hibernation[18] features, because the FDE software has to encrypt and decrypt the large hibernation file (which includes a copy of the computer's memory) that is stored on the hard drive. The length of delays is dependent on the size of memory, the hard drive's size and speed, and other factors.

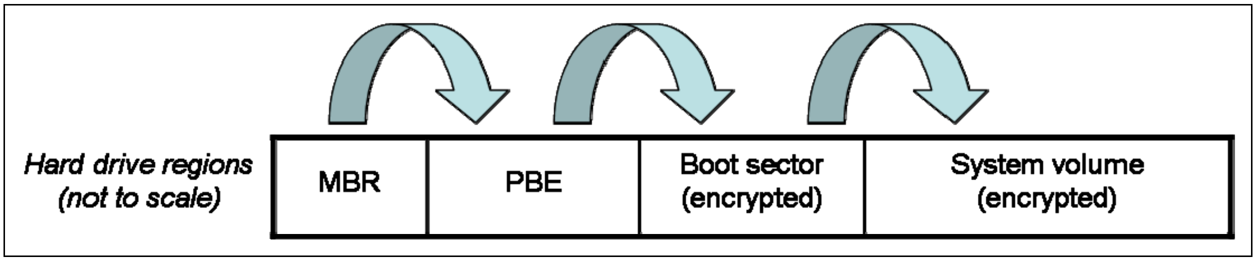

Figure 3-1. Boot Sequence for Full Disk Encryption Software

Because FDE alters how a computer boots, it can cause operational problems. For example, modifying the MBR can prevent computers with dual-boot configurations from functioning properly,[19] and storing the PBE in the space between the MBR and the boot sector can cause conflicts with other software, such as disk-level software tools, that also store code in that space. FDE-protected devices may also have problems with asset management tools and the use of wake-on-LAN.[20]

FDE software is most commonly used on desktop and laptop computers. The requirement for pre-boot authentication means that users have to be able to authenticate using the most fundamental components of a device, such as a standard keyboard—because the OS is not loaded, OS-level drivers are unavailable. For example, a PDA or smart phone could not display a keyboard on the screen for entering a password because that would be an OS-level capability.

As mentioned at the beginning of the section, FDE can also be built into a hard drive disk controller.[21] Hardware and software-based FDE offer similar capabilities through different mechanisms. When a user tries to boot a device protected with hardware-based FDE, the hard drive prompts the user to authenticate before it allows an OS to load. The FDE capability is built into the hardware in such a way that it cannot be disabled or removed from the drive. The encryption code and authenticators, such as passwords and cryptographic keys, are stored securely on the hard drive. Because the decryption and encryption is performed by the hard drive itself, with no OS participation, typically there is very little performance impact.

[18] Hibernation refers to saving the state of the computer, including the contents of memory, and placing the computer in a low power-usage mode that uses just enough power to maintain its state. Depending on the OS, hibernation mode may also be called sleep, standby, or suspend mode; however, some of these terms do not have universally accepted definitions, and some OSs have features with these names that do not actually write memory out to a file. Modes that do not write memory to a file should not be used with FDE software because the FDE software will not protect the data in these modes.

[19] Dual-boot computers usually modify the MBR to enable users to select which OS will be booted. The details of this depend on the OSs and any utilities used to manage the disk partitions. Using FDE with a dual-boot computer increases the complexity of the configuration and the likelihood for errors that cause loss of data or loss of availability.

[20] The problem with wake-on-LAN technologies is the requirement to perform PBA before booting the system. Some FDE products can be configured to skip PBA when wake-on-LAN is used, either every time or for a certain number of reboots. Access to the OS logon can be suppressed to somewhat compensate for the lack of PBA. However, skipping PBA during wake-on-LAN still introduces additional risk, because an attacker that takes a device that is configured this way could easily set up wake-on-LAN services, circumvent PBA, and use forensic tools to gain access to the device's information. Organizations should carefully consider these risks before using wake-on-LAN for FDE-protected devices.

[21] As of mid-2007, few products are available.

A major difference between software and hardware-based FDE is that software-based FDE can be centrally managed, but hardware-based FDE can usually only be managed locally. This makes key management and recovery actions considerably more resource-intensive and cumbersome for hardware-based FDE than software-based. Another major difference is that because hardware-based FDE does all cryptographic processing within the hard drive's hardware, it does not need to place its cryptographic keys into the computer's memory, which could potentially expose the keys to malware and other threats.

3.1.2 Virtual Disk Encryption and Volume Encryption

Virtual disk encryption is the process of encrypting a file called a *container*, which can hold many files and folders, and permitting access to the data within the container only after proper authentication is provided, at which point the container is typically mounted as a virtual disk. Virtual disk encryption is used on all types of end user device storage. The container is a single file that resides within a volume. Examples of volumes are boot, system, and data volumes on a personal computer, and a USB flash drive formatted with a single filesystem. *Volume encryption* is the process of encrypting an entire volume and permitting access to the data on the volume only after proper authentication is provided. Volume encryption is most often performed on hard drive data volumes and volume-based removable media, such as USB flash drives and external hard drives. Volume encryption of boot and system volumes is essentially a special form of FDE, and it is not discussed in this section; see the FDE material in Section 3.1.1 for additional information.

At a high level, volume and virtual disk encryption are performed similarly. Software running on the OS used to access the volume or container handles all attempts to read to or write from the protected volume or container.[22] Once the OS has been loaded, if the user needs to use the encrypted volume or container, it will be mounted after the user has provided the required authentication. The software will then automatically decrypt and encrypt the appropriate sectors as needed. This increases the time needed to open or save files, but the delay generally should be noticeable for only particularly large files. There may also be slight delays associated with mounting and unmounting an encrypted volume or container.

The key difference between volume and virtual disk encryption is that containers are portable and volumes are not—a container can be copied from one medium to another, with encryption intact. This allows containers to be burned to CDs and DVDs and to be used on other media that are not volume-based. Virtual disk encryption also makes it trivial to back up sensitive data; the container is simply copied to the backup server or media. Another advantage of virtual disk encryption over volume encryption is that virtual disk encryption can be used in situations where volume-based removable media needs to have both protected and unprotected storage; the volume can be left unprotected and a container placed onto the volume for the sensitive information.

Some virtual disk encryption products further support mobility by offering features that can place executables on the medium holding a container. The medium can then be moved to another computer and the executables run, through methods such as installing drivers onto the computer or running an authentication and decryption utility.[23] The protected contents of the medium can then be accessed by a user after providing the requested authentication.

The responsibilities of the users of virtual disk and volume encryption solutions vary, primarily depending on the devices' access control. For example, if a laptop's OS is configured so that a user can only write files to an encrypted container or volume, then the user does not need to take steps to ensure that files are saved to the appropriate location. However, if the OS is not configured this way, permitting

[22] Some products install kernel mode drivers to perform volume and virtual disk encryption. Other products, especially those specifically designed for removable media, either contain their own resident OSs or provide software applications.

[23] This only speaks to the portability of the logical entity on the media—the media itself might be physically portable.

users to save files to various locations, or if the encrypted device is removable media that is not protected through OS access control features, then users will be responsible for ensuring that they save files in the appropriate location. In this case, if users fail to follow the necessary procedures, then some files that should be protected may not be.

3.1.3 File/Folder Encryption

File encryption is the process of encrypting individual files on a storage medium and permitting access to the encrypted data only after proper authentication is provided. *Folder encryption* is very similar to file encryption, only it addresses individual folders instead of files. Some OSs offer built-in file and/or folder encryption capabilities,[24] and many third-party programs are also available. Although folder encryption and virtual disk encryption sound similar—both a folder and a container are intended to contain and protect multiple files—there is a difference. A container is a single opaque file, meaning that no one can see what files or folders are inside the container until the container is decrypted. File/folder encryption is transparent, meaning that anyone with access to the filesystem can view the names and possibly other metadata for the encrypted files and folders, including files and folders within encrypted folders, if they are not protected through OS access control features. File/folder encryption is used on all types of storage for end user devices.

File/folder encryption can be implemented in many ways, including through drivers, services, and applications. When a user attempts to open an encrypted file (either encrypted by itself or located in an encrypted folder), the software requires the user to first authenticate successfully. Once that has been done, the software will automatically decrypt the chosen file. Because it decrypts a single file at a time, the performance impact of file/folder encryption should be minimal. File/folder encryption is most commonly used on user data files, such as word processing documents and spreadsheets. File/folder encryption solutions can sometimes encrypt swap files, but typically not OS executables and hibernation files.

Many file/folder encryption products offer several options for selecting which files and folders should be encrypted and defining the user's role in using the solution—manually enabling encryption for each new file or folder that needs protected, remembering to store files and folders in the proper locations, or doing nothing differently because the files and folders are encrypted automatically. Common options include:

- Relying on the user to specifically designate the files and folders
- Automatically encrypting the contents of administrator-designated folders
- Automatically encrypting certain types of files, such as those with a particular file extension
- Automatically encrypting all files written to by particular applications
- Automatically encrypting all data files for particular users.

There are also various applications, such as file compression utilities and office productivity suites, that offer limited file/folder encryption capabilities. Such applications are usually completely dependent on the user to ensure that the necessary files are encrypted, and these applications are often not centrally managed, which can complicate key management and other aspects of managing the use of the applications' file/folder encryption features. Appendix A presents additional examples of applications that can encrypt the information that they store.

[24] For example, NTFS supports file and folder encryption using the Encrypting File System (EFS).

3.2 Protection Provided by Storage Encryption Technologies

The following explains the types of protection each storage encryption technology can and cannot provide.

- **Full Disk Encryption.** For a computer that is not booted, all the information encrypted by FDE is protected, assuming that pre-boot authentication is required. When the device is booted, then FDE provides no protection; once the OS is loaded, the OS becomes fully responsible for protecting the unencrypted information. The exception to this is when the device is in a hibernation mode; most FDE products can encrypt the hibernation file.

- **Virtual Disk and Volume Encryption.** When virtual disk encryption is employed, the contents of containers are protected until the user is authenticated for the containers. If single sign-on is being used for authentication to the solution, this usually means that the containers are protected until the user logs onto the device. If single sign-on is not being used, then protection is typically provided until the user explicitly authenticates to a container. Virtual disk encryption does not provide any protection for data outside the container, including swap and hibernation files that could contain the contents of unencrypted files that were being held in memory. Volume encryption provides the same protection as virtual disk encryption, but for a volume instead of a container.

- **File/Folder Encryption.** File/folder encryption protects the contents of encrypted files (including files in encrypted folders) until the user is authenticated for the files or folders. If single sign-on is being used, this usually means that the files are only protected until the user logs onto the device. If single sign-on is not being used, then protection is typically provided until the user explicitly authenticates to a file or folder. File/folder encryption does not provide any protection for data outside the protected files or folders, including swap and hibernation files that could contain the contents of unencrypted files that were being held in memory. File/folder encryption software also cannot protect the confidentiality of filenames and other file metadata, which itself could provide valuable information to attackers (for examples, files that are named by Social Security number).

In many cases, especially for FDE and volume encryption, these products do not provide any protection for files copied or moved from the encrypted storage to another location (either local or on the network), because they automatically decrypt the files as part of the copy or move process.[25] The target location is responsible for protecting the files, and no protection is provided in transit from the source to the target. However, some storage encryption technologies allow protection to be retained if desired. Most virtual disk encryption products allow an entire container to be transferred, including the container's protection, but individual files or folders copied or moved from a container will not be protected. Some file/folder encryption products allow files or folders to retain their protection when they are copied or moved, in some cases only within a single filesystem, and in other cases also to other filesystems.

The main threat that all these types of technology mitigate is unauthorized access to information on a lost or stolen device. Virtual disk/volume encryption and file/folder encryption technologies can also mitigate some OS and application layer threats to protected information involving malware, remote access to the protected information, and other methods that depend on the OS being booted, until the user successfully authenticates to the encryption solution. Once this authentication occurs, then any process being run on the device (such as malware) with access to the user's files can get the decrypted information. Because the files are only protected until successful authentication occurs, it may be beneficial to use a solution that is configured to encrypt only the necessary files (e.g., using file/folder encryption to encrypt 10

[25] Some products display a warning message or prompt the user to confirm the action before decrypting and copying or moving the files.

sensitive files instead of using volume encryption to encrypt 10 sensitive files and 1000 non-sensitive files). The more files that are protected, the sooner the user is likely to authenticate to the storage encryption solution, which increases the window of exposure for the decrypted files.

Some products also permit storage to be encrypted either for a single user or for multiple users of a device. If encrypted for a single user, the confidentiality of that user's encrypted storage is protected from other users of the device, including (in most cases) the device's administrators. Encrypting for multiple users allows sensitive data to be shared by those users, while still protecting it from other users of the device. This provides protection against insider threats.

In some cases, multiple types of technology can be used concurrently to protect against different classes of threats; for example, FDE could be used to protect all data on a device from device loss or theft, and volume, virtual disk, or file/folder encryption could be used to provide additional protection for a subset of data that is more sensitive than the rest of the data.[26]

When thinking about threats, organizations should be aware that after storage encryption technology has been implemented, there may be residual data on the device that remains unprotected. For example, when a file is encrypted using file/folder encryption and the original file is deleted, the remnants of the original plaintext file might still be present on the storage media. Another example is FDE and volume encryption products that encrypt only the disk sectors that contain current files, not disk sectors that only contain deleted files or other remnants of data. These remnants may be recoverable using forensic tools by an attacker who gets physical access to the computer, without having to provide any authentication. Organizations should take into account threats against both the files and remnants of the files.

Organizations should be aware that if an end user device is compromised at any time, any storage encryption technologies on it may become partially or wholly ineffective. For example, when the device is in use and the user has been authenticated to the storage encryption solution, malware could access decrypted files and transfer copies of them to external hosts or extract sensitive information from them. Other examples are an attacker disabling or reconfiguring storage encryption, malware installing a keylogger that captures passwords used for storage encryption authentication, or malware acquiring a copy of a storage encryption key from the device's memory (for software-based storage encryption solutions).

3.3 Comparison of Storage Encryption Technologies

Table 3-1 lists several characteristics of storage encryption technologies as a means for comparing the types of technologies described in this publication.

[26] When multiple encryption methods are used simultaneously, the cryptographic keys used by the methods are usually different.

Table 3-1. Characteristics of Storage Encryption Technologies

Characteristic	Full Disk Encryption	Volume Encryption	Virtual Disk Encryption	File/Folder Encryption
Typical platforms supported	Desktop and laptop computers	Desktop and laptop computers, volume-based removable media (e.g., USB flash drives)	All types of end user devices	All types of end user devices
Data protected by encryption	All data on the media (data files, system files, residual data, and metadata)	All data in the volume (data files, system files, residual data, and metadata)	All data in the container (data files, residual data and metadata, but not system files)	Individual files/folders (data files only)
Mitigates loss or theft of device?	Yes	Yes	Yes	Yes
Mitigates OS and application layer threats (such as malware and insider threats)?	No	If the data volume is being protected, it sometimes mitigates such threats.* If the data volume is not being protected, then there is no mitigation of these threats.	It sometimes mitigates such threats*	It sometimes mitigates such threats*
Potential impact to devices in case of solution failure	Loss of all data and device functionality	Loss of all data in volume; can cause loss of device functionality, depending on which volume is being protected	Loss of all data in container	Loss of all protected files/folders
Portability of encrypted information	Not portable	Not portable	Portable	Often portable

** These storage encryption technologies can only protect the files against some OS and application layer threats if the user has not been authenticated in this session to access the files. If a single sign-on solution is used, then generally the user is authenticated to the storage encryption technology during OS login, so the files are not protected against these threats once OS login occurs. If a separate authentication solution is used, the files are protected until that separate authentication is performed.*

When selecting storage encryption technologies, an organization should take into consideration the extent to which each technology will require the infrastructure and end user devices to be changed. For example, using some technologies requires deploying additional servers and installing software on the devices to be protected, while other technologies can use existing servers, as well as software built in to the devices to be protected. Generally, the more extensive the changes are to the infrastructure and devices, the more likely it is that the storage encryption technology will cause a loss of functionality or other problems with the devices. Technologies that require extensive changes to the infrastructure and end user devices should generally be used only when other technologies cannot meet the organization's needs.

The following are use cases that highlight the types of storage encryption technologies that may be suitable for certain situations. Each use case presents a brief scenario, including the threats that need to be mitigated, and then proposes possible solutions, both storage encryption technologies and alternate solutions (if applicable). Each use case only lists high-level solutions that may be feasible, and is not intended to imply that other solutions are not possible or that these solutions are preferable to others. Each use case also omits security controls that are universal to the solutions, such as user awareness and general endpoint security (e.g., patching, antivirus software, access control).

3.3.1 Use Case 1: Sharing a Laptop

Three users share a laptop. One of the users uses the laptop to access data that the other two users are not authorized to access. For this data, the major threats that the organization needs to mitigate are an insider threat from the other two users, and the loss or theft of the laptop. Possible solutions include the following:

- Implement volume, virtual disk, or file/folder encryption on the laptop. Protect the first user's data using the storage encryption software, with the authentication and cryptographic keys implemented so that only the first user, and not the other two users, can access the protected data. If there is concern about the first user always remembering where to save files, configure the laptop's access control so that the first user's data is all saved to a particular location, and protect that location with the storage encryption software.

- Store the data on external media, such as a flash drive or external hard drive, and use volume, virtual disk, or file/folder encryption to protect the media. The user needs to protect physical access to the media and to remember to save new or modified data to the media.

- Store the data on a remote system and give the first user access to the data through secured means (e.g., VPN). Provide the data in such a way that it is not saved to the laptop (e.g., the user views and modifies the remote data through a Web interface).

3.3.2 Use Case 2: Transferring Files Between Computers

A user edits documents using both a desktop PC at the organization's office and a personally owned computer at home. The user transfers documents between the computers on a daily basis using a USB flash drive. The two computers run different types of OSs. For the documents, the major threat that the organization needs to mitigate is loss or theft of the user's flash drive. Possible solutions include the following:

- Acquire and use a USB flash drive with self-contained storage encryption capabilities, such as encryption software and secure key storage.

- Acquire a volume, virtual disk, or file/folder encryption solution that will work on both PCs, and deploy it. Encrypt the documents using the solution and store the encrypted data on a flash drive.

3.3.3 Use Case 3: Sharing Data with Contractor

A user wants to provide a contractor with copies of large data sets on a daily basis because the contractor has no direct access to the system containing the data. The user will copy the data onto removable media for the contractor.[27] For this data, the major threat that the organization needs to mitigate is loss or theft of the removable media. Possible solutions include the following:

- Deploy virtual disk or file/folder encryption software to the user and contractor's computers. Encrypt the data using the software and burn the encrypted data onto CDs or DVDs.

- Acquire USB flash drives or external hard drives that have built-in storage encryption capabilities. Store the copies of the data on the encrypted drives.

[27] Another use case, with similar possible solutions, is a user that needs to protect backups of a PC from loss or theft.

- Acquire USB flash drives or external hard drives. Deploy virtual disk, volume, or file/folder encryption software to the user and contractor's computers. Encrypt the data using the software and store it on the drives.

3.3.4 Use Case 4: Traveling with a Laptop

A user occasionally travels on behalf of the organization and carries a laptop that contains sensitive data. For this data, the major threat that the organization needs to mitigate is the loss or theft of the laptop. Possible solutions include the following:

- Use the laptop's OS access control features to strictly limit where the user can save files. Implement volume, virtual disk, or file/folder encryption on the laptop to protect the user's files.

- Implement FDE on the laptop, and require pre-boot authentication.

- Provide the user with a loaner laptop when needed for travel. Protect the user's sensitive data on the laptop using either of the methods described above. When the user returns from travel, wipe and rebuild the loaner laptop to remove any traces of sensitive data from it. Using a loaner laptop in this way is particularly helpful if the laptop is being used in hostile environments, where the laptop is at greater risk of being compromised.

3.3.5 Use Case 5: Traveling with a Dual-Boot Laptop

A user frequently travels on behalf of the organization and carries a laptop that contains sensitive data. The laptop is dual-boot, using two OSs. For the user's data, the major threat that the organization needs to mitigate is the loss or theft of the laptop. Possible solutions include the following:

- Use the OS access control features of each OS to strictly limit where the user can save files. Implement volume, virtual disk, or file/folder encryption on both of the laptop's OSs to protect the user's files.

- Implement an FDE solution that supports dual-boot configurations.

- Convert the laptop to be single-boot, and access the second (removed) OS through a virtual machine run by the primary OS. See use case 4 for further information on protecting the laptop.

3.4 Storage Encryption Technology Management

Most storage encryption deployments are managed centrally. Centralized management is most often performed through special management utilities provided by the storage encryption vendor. If the storage encryption solution is built into the devices' OSs, then it could be managed through the mechanisms already in place to manage OS configurations. The capabilities of centralized management utilities for storage encryption technologies vary considerably. Examples of commonly implemented capabilities are as follows:

- Deploying storage encryption software to additional devices

- Updating storage encryption software (e.g., patching, upgrading)

- Configuring storage encryption software, such as specifying encryption algorithms and setting authentication policies (in some cases, the policies are specific for types of devices, groups of users, and/or individual users)
- Managing storage encryption authenticators and cryptographic keys[28]
- Collecting and reviewing storage encryption-related logs
- Recovering stored information from device failures
- Performing routine system maintenance
- Enabling the encryption of data and managing encrypted storage
  - For FDE, this involves encrypting the computer's hard drive. Some products allow the initial encryption to be done while the computer is in use, but it can cause a substantial performance impact if not configured properly, and additional hard drive space may be needed. Other products require that the computer not be in use while the drive is initially encrypted.
  - For volume encryption, this could involve either encrypting an existing volume, or creating a new volume, encrypting it, and then having the user add files to the volume as needed.
  - For virtual disk encryption, this simply involves creating a container. Files can then be added to the container by the user as needed.
  - For file/folder encryption, this could involve encrypting existing files, setting up an encrypted folder for future files, or establishing policies to automatically encrypt certain types of files.

Some storage encryption products, particularly ones intended for standalone deployment, can be managed locally. Local management is typically performed by a system administrator (for managed devices) or a user (for unmanaged devices) who has physical access to the device running the storage encryption technology. A common local management task is recovering data; many products allow users to recover their own data by running a recovery utility. For example, a device using FDE might experience a failure that prevents it from booting the OS; a user could run a recovery utility from the pre-boot environment or a CD to extract the data from the device.

Organizations may choose to deploy storage encryption without a centralized management capability and perform all management locally. This is generally acceptable for standalone deployments and very small-scale deployments, particularly ones that need to be done quickly, without waiting for a centralized management infrastructure to be implemented. However, for all other deployments, centralized management is recommended because it is more effective and efficient for most management tasks, including policy verification and enforcement, key management, authenticator management, and data recovery.

[28] Section 4 contains additional information on cryptography, key management, and authentication.

4. Storage Encryption Technology Planning and Implementation

This section discusses considerations for planning and implementing storage encryption technologies for end user devices. As with any new technology deployment, storage encryption technology planning and implementation should be addressed in a phased approach. A successful deployment can be achieved by following a clear, step-by-step planning and implementation process. The use of a phased approach for deployment can minimize unforeseen issues and identify potential pitfalls early in the process. This model also allows for incorporating advances in new technology and adapting the technology to the ever-changing enterprise. The following is an example of planning and implementation phases:

1. **Identify Needs.** The first phase involves identifying the needs to encrypt storage on end user devices, determining which devices and data need protected, and identifying related requirements (e.g., minimum performance). This phase also involves determining how that need can best be met (e.g., FDE, virtual disk encryption) and deciding where and how the security should be implemented.

2. **Design the Solution.** The second phase involves all facets of designing the solution. Examples include architectural considerations, authentication methods, cryptography policy, and supporting security controls.

3. **Implement and Test a Prototype.** The next phase involves implementing and testing a prototype of the designed solution in a lab or test environment. The primary goals of the testing are to evaluate the functionality, performance, scalability, and security of the solution, and to identify any issues with the components, such as interoperability issues.

4. **Deploy the Solution.** Once the testing is completed and all issues are resolved, the next phase includes the gradual deployment of the storage encryption technology throughout the enterprise.

5. **Manage the Solution.** After the solution has been deployed, it is managed throughout its lifecycle. Management includes maintenance of the storage encryption components and support for operational issues. The lifecycle process is repeated when enhancements or significant changes need to be incorporated into the solution.

This document does not describe the planning and implementation process in depth because the same basic steps are performed for any security technology. For example, the document assumes that the organization has already determined what information needs to be protected and assigned an impact level from FIPS 199 to the information. This section only highlights those considerations that are particular to storage encryption for end user devices.

4.1 Identify Needs

The purpose of this phase is to identify the needs to protect information stored on end user devices and determine how those needs can best be met. Requirements specific to storage encryption that should be considered include the following:

- **External Requirements.** The organization may be subject to oversight or review by another organization that requires storage encryption. An example is a legal requirement to protect stored PII.

- **System and Network Environments.** It is important to understand the characteristics of the organization's system and network environments so that storage encryption solutions can be selected that will be compatible with them and able to provide the necessary protection. Aspects to consider include the following:

- The characteristics of the devices that need protection, especially the OSs, applications, and filesystems they use, and their hardware capabilities and characteristics
- The technical attributes of the interfaces of other systems with which the storage encryption solution might be integrated, such as authentication services, centralized logging servers and security information and event management (SIEM) software, and patch management software

The outcome of the organization's analysis should be a determination of which files or types of files need to be encrypted and which types of threats the storage encryption software should protect against, stating the concerns as specifically as possible. For example, the organization may decide to encrypt sensitive information on all devices used outside the organization's facilities and to encrypt certain types of sensitive information on devices used from any location.

The analysis should also lead to the identification of the type or types of storage encryption technologies that can meet the organization's security needs. Another outcome of the analysis is the documentation of the requirements for the storage encryption technologies themselves, including security capabilities (e.g., authentication, cryptography, key management), performance requirements, management requirements (including reliability, interoperability, scalability), the security of the technology itself, usability (by both administrators and users), and maintenance requirements (such as applying updates).

In most cases, a single storage encryption product cannot meet all of the organization's identified needs. For example, the organization may need to protect information on devices running several different OSs, yet no appropriate product can work on all those platforms. Some devices might also not meet the minimum hardware requirements for storage encryption products. Organizations can solve this problem in several ways, such as acquiring multiple products, using multiple types of storage encryption technologies, replacing older devices, or identifying compensating controls to be used instead of storage encryption that provide the same level of protection. Appendix A discusses some potential alternatives to storage encryption. Organizations should ensure that effective solutions are identified for all the types of end user devices that need their stored information protected, if possible, and that a waiver process is created for unusual cases that cannot be addressed by the identified solutions.

4.2 Design the Solution

Once the needs have been identified and the appropriate type(s) of storage encryption technology have been chosen, the next phase is to design a solution that meets the needs. If these design decisions are incorrect, then the storage encryption implementation will be more susceptible to compromise. Major aspects of solution design that are particularly important for storage encryption are as follows:

- **Cryptography.** Encryption and integrity protection algorithms must be selected, as well as the key strength for algorithms that support multiple key lengths. Key management and protection is another important component of solution design. Section 4.2.1 contains additional cryptography information.
- **Authentication.** Authentication methods must be chosen for users and administrators. Decisions also need to be made regarding the protection of the authenticators themselves. Section 4.2.2 contains a more detailed discussion of authentication.
- **Solution architecture.** The architecture of the storage encryption implementation refers to the selection of devices and software to provide storage encryption services and the placement of centralized elements within the existing network infrastructure, such as authentication credential servers, Web servers for self-service recovery, and management servers. Each end user device must

have hardware and/or software that provides protection for the stored information. Designing the architecture includes component placement, redundancy, reliability, and interoperability.

- **Other security controls.** These support and complement the storage encryption implementation. For example, organizations should have policies regarding acceptable usage of storage encryption technologies. Organizations may also set minimum security standards for end user devices, such as mandatory host hardening measures and patch levels, and specify security controls that must be employed, such as host-based personal firewalls, antivirus software, and antispyware software.

- **Minimum requirements for end user devices.** The minimum requirements for the hardware, OS, and supporting software should be defined. They should be based on the requirements supplied by the product vendor and the organization's performance requirements.

Another aspect of solution design is planning the logistics of the solution's deployment. For example, the organization may need to replace devices that do not meet minimum requirements or run on a platform that the organization will not support for storage encryption. This could cause out-of-cycle upgrades or replacements of hardware, OSs, and supporting software. Another logistical consideration is how the solution will be deployed to end user devices. If devices need to be updated locally, such as upgrading the OS, replacing the hard drive, backing up user files, or installing storage encryption software, then organizations need to plan who will perform these actions and when and where the work will be done. Some organizations may need to set up staging areas and get additional personnel to perform this work.

4.2.1 Cryptography

Storage encryption technologies use one or more cryptographic keys to encrypt and decrypt the data that they protect. The number of keys and the types of keys used are product and implementation-dependent. For example, public key cryptography uses a pair of keys, and symmetric cryptography uses a single key. Some products support the use of a recovery key that can be used to recover the encrypted data if the regular key is lost. Also, some technologies permit encrypted storage to be shared by multiple users, which could be enabled by having a different key for each user. Often, users' keys are not directly used to decrypt their stored data; instead, those keys are used to decrypt another key, which in turn is used to decrypt the stored data.

If a key is lost or damaged, it may not be possible to recover the encrypted data. Therefore, organizations need to ensure that all keys used in a storage encryption solution are secured and managed properly to support the security of the solution. Organizations should perform extensive planning of key management processes, procedures, and technologies before implementing storage encryption technologies. This planning should include all aspects of key management, including key generation, use, storage, and destruction.[29] Organizations should carefully consider how key management practices can support the recovery of encrypted data if a key is inadvertently destroyed or otherwise becomes unavailable (such as a user unexpectedly leaving an organization or losing a cryptographic token containing a key). An example of recovery preparation is storing duplicates of keys in a centralized, secured key repository or on physically secured removable media. Organizations planning on encrypting removable media also need to consider how changing keys will affect access to encrypted storage on the media and develop feasible solutions, such as retaining the previous keys in case they are needed.

Another decision that may need to be made is where the local keys should be stored. For some encryption technologies, such as FDE and many file/folder encryption products, there are often several options for

[29] NIST SP 800-57, *Recommendation for Key Management*, provides detailed information on key management planning, algorithm selection and appropriate key sizes, cryptographic policy, and cryptographic module selection.

key location, including the local hard drive, a USB flash drive, a cryptographic token, or a Trusted Platform Module (TPM) chip.[30] Some products also permit keys to be stored on a centralized server and retrieved automatically after the user authenticates successfully. For volume and virtual disk encryption, the main encryption key is often stored encrypted within the volume or container itself. Some storage encryption products do not store a key; instead, they perform a cryptographic hash function on the password entered by the user and use that hash as the key.

Organizations need to ensure that access to keys is properly restricted. Storage encryption solutions should require the use of one or more authentication mechanisms, such as passwords, smart cards, and cryptographic tokens, to decrypt or otherwise gain access to a storage encryption key. The keys themselves should be logically secured (e.g., encrypted) or physically secured (e.g., stored in a tamper-resistant cryptographic token). The authenticators used to retrieve keys should also be secured properly.

In addition to key management, there are several other aspects of cryptography that need to be considered when planning a storage encryption solution. Setting the cryptography policy involves choosing encryption and integrity protection algorithms and key lengths.[31] Federal agencies must use FIPS-approved algorithms contained in validated cryptographic modules.[32] Whenever possible, AES[33] should be used for the encryption algorithm because of its strength and speed. Several FIPS-approved algorithms are available for integrity checking, including HMAC-SHA, Cipher-Based Message Authentication Code (CMAC), and Counter with Cipher Block Chaining-Message Authentication Code (CCM).[34] Organizations should consider how easily the solution can be updated when stronger algorithms and key sizes become available in the future.

4.2.2 Authentication

There are two types of authentication important to storage encryption. Administrators authenticate so that they can perform storage encryption management functions, including reconfiguring and updating encryption software, managing user accounts, and recovering encrypted data.[35] Users authenticate so that they can access encrypted information. If a single authenticator is used (often a user ID and password, sometimes a token), that authenticator typically grants the storage encryption software access to the key used to encrypt and decrypt the stored information. If two-factor authentication is used, typically one of the factors grants access to information secured in another factor, which is then used to gain access to the storage encryption key. For example, a PIN or password could be used to retrieve a key from a smart card or cryptographic token; that key could then be used to decrypt the storage encryption key.[36]

[30] A *TPM chip* is a tamper-resistant integrated circuit built into some motherboards that can perform cryptographic operations (including key generation) and protect small amounts of sensitive information, such as passwords and cryptographic keys. As of this writing, no FIPS-approved TPM chips are yet available.

[31] SP 800-21, Second Edition, *Guideline for Implementing Cryptography in the Federal Government*, presents guidelines for selecting, specifying, employing, and evaluating cryptographic protection mechanisms in Federal information systems. It defines a process for selecting cryptographic products and discusses implementation issues, including solution management, key management, and authentication.

[32] The Cryptographic Module Validation Program (CMVP) at NIST coordinates FIPS 140-2 testing; the CMVP Web site is located at http://csrc.nist.gov/cryptval/. See http://csrc.nist.gov/cryptval/des.htm for information on FIPS-approved symmetric key algorithms, and http://csrc.nist.gov/cryptval/dss.htm for information on digital signature algorithms. FIPS 140-2, *Security Requirements for Cryptographic Modules*, is available at http://csrc.nist.gov/publications/fips/.

[33] For more information, read FIPS 197, *Advanced Encryption Standard (AES)*, at http://csrc.nist.gov/publications/fips/.

[34] Additional information on these algorithms is available at http://csrc.nist.gov/CryptoToolkit/modes/.

[35] The term “administrators” is used generically to refer to the individuals responsible for managing the storage encryption solution. Some organizations use more specific terms for these individuals, such as “cryptographic officers”.

[36] The following authentication methods for storage encryption are listed from weakest to strongest based on their general effectiveness: single sign-on, unique password or PIN, token, token with unique password or PIN. See NIST SP 800-63, *Electronic Authentication Guideline*, for additional information (http://csrc.nist.gov/publications/nistpubs/).

Some storage encryption products allow the use of multiple user IDs on a single device. If the IDs are tied to a single storage encryption key, then each user can access the same protected information. If each ID is linked to a separate key, then the access is dependent on how the keys are used—for example, a container could be encrypted using a single key so that only one user can access it, or encrypted using several keys so that users can share the contents of the container.

For storage encryption authentication, organizations often want to leverage existing enterprise authentication solutions (e.g., Active Directory, RADIUS, PKI) instead of adding another authenticator for users. Generally, this is acceptable if two-factor authentication is being used because of its strength. Using the same single-factor authenticator for multiple purposes significantly weakens the protection that authentication provides.[37] For example, reusing a user's OS password for pre-boot authentication in an FDE deployment would allow an attacker to learn only a single password to gain full access to the device's information. The password could potentially be acquired through technical methods, such as infecting the device with malware, or through physical means, such as watching a user type in a password in a public location. Another example is having a volume, virtual disk, or file/folder encryption product use the OS's authentication for single sign-on (SSO). Once a user authenticates to the OS at login, the user can access the encrypted files without further authentication, so the security of the solution is heavily dependent on the strength of the OS authenticator. Organizations should carefully consider the security implications of using the same single-factor authenticator for multiple purposes. In particular, organizations should not use email passwords and other passwords sometimes transmitted in plaintext as single-factor authenticators for storage encryption. Also, organizations concerned about targeted attacks, such as someone stealing a particular laptop to gain access to a specific user's data, should not use only passwords for storage encryption authentication because of the relative ease of capturing a user's password (e.g., watching a password being typed).

Organizations also need to ensure that the storage encryption authenticators are protected properly. This includes both technical mechanisms, such as encrypting passwords or storing cryptographic hashes of passwords, and operational and management mechanisms. For example, policy should state and users should be made aware that authenticators should not be stored in proximity of the end user device (e.g., a password should not be on a piece of paper in a laptop case), and that for two-factor authentication, multiple authenticators should not be stored with each other (e.g., a password or PIN should not be written on the back of a hardware token).

Because authentication controls access to storage encryption keys, the loss of authenticators can prevent access to the encrypted data. Organizations should determine how the loss of authenticators (both user and administrator-level) will be handled before implementing storage encryption. Most products offer recovery mechanisms for password-based user authentication. For example, a user that has forgotten a password chooses a recovery option on the protected computer. The computer provides a special code to the user, who then uses another computer to access the organization's storage encryption recovery Web site. The user provides the code to the Web site and gives proof of identity, such as answering questions about personal preferences (e.g., favorite color) that the user has previously configured. The Web site then provides the user's password or a one-time recovery code that the user enters into the computer to regain access. A similar process can also be performed by having users call a help desk instead of accessing a particular Web site.

For user authentication methods other than password-based, recovery is often more difficult, especially if the user is not at the organization's facilities. Some storage encryption products allow the password-based authentication recovery mechanisms to be used and permit the user to temporarily use password-

[37] In many cases, single-factor authentication will not be an option because the sensitivity of the data being protected will necessitate the use of multi-factor authentication.

based authentication. However, because this is generally a reduction in the strength of authentication, many organizations do not permit its use. This means that a loss of authenticator could cause an extended loss of availability to the data, until the user can receive a new authenticator (e.g., smart card, cryptographic token) and an administrator can configure the device to use the new authenticator.

Some storage encryption products also offer protection against authentication-guessing attempts. For example, if there are too many consecutive failed authentication attempts, some products can either lock the computer for a period of time or increase the delay between attempts. In particularly high-security situations, some products can be configured so that too many failed attempts causes the product to wipe all the protected data from the device. This approach strongly favors security over functionality.

4.3 Implement and Test Prototype

After the solution has been designed, the next step is to implement a test a prototype of the design. Ideally, implementation and testing should first be performed on lab or test devices. Only implementations in final testing should be conducted on production devices. Aspects of the solution to evaluate include the following:

- **Protection.** Each type of information that needs protection should be protected in accordance with the information gathered during the Identify Needs phase. This should be verified by using forensic tools to confirm that the information is encrypted. For devices that use FDE and offer hibernation, standby, or other “suspend” modes, encryption should be verified in each mode; if the mode does not write the contents of memory out to disk and encrypt it, then the information may be readily available unencrypted.[38]

- **Authentication.** Performing robust testing of authentication is important, especially for more complex authentication solutions that depend on centralized authentication services; a loss of those services could cause a loss of storage encryption services as well.

- **OS and Application Compatibility.** The solution should not break or interfere with the use of existing OS configurations and software applications. Examples of applications that may be particularly problematic are, for FDE, disk-level software tools, asset management software, and dual-boot configurations, and for all storage encryption technologies, backup utilities, forensic tools, and other storage encryption programs.

- **Management.** Administrators should be able to configure and manage all components of the solution effectively and securely. It is particularly important to evaluate the ease of deployment and configuration. Another concern is the ability of administrators to disable configuration options so that users cannot circumvent the intended security. Management concerns should include the effects of patching/upgrading software, changing software settings (e.g., changing cryptographic algorithms or key sizes), uninstalling or disabling encryption software, changing encryption/decryption keys, and changing user or administrator passwords.

- **Logging.** The logging and data management functions should function properly in accordance with the organization’s policies and strategies.

- **Performance.** The solution should be able to provide adequate performance during normal and peak usage. Testing should incorporate a variety of devices, OSs, and applications, especially those that are most likely to be affected by performance issues, such as those that manipulate large files.

[38] This can be addressed by configuring the device not to use modes that maintain the data in an unencrypted format.

- **Security of the Implementation.** The storage encryption implementation itself may contain vulnerabilities and weaknesses that attackers could exploit. Organizations with high security needs may want to perform extensive vulnerability assessments against the storage encryption components. Another common security concern is the security of the authenticators and cryptographic keys.

- **Recovery.** The solution should be tested to determine how well it can recover from failures, such as lost or forgotten authenticators, lost keys, device hardware or software failure/damage, and power loss.

- **Interoperability.** For a solution that will protect removable media that will be used on multiple devices, the organization should ensure that information encrypted on the media by one device can be decrypted by another device after authenticating successfully.

- **Operational Impacts.** Organizations should determine how the solution might impact operations, such as impeding technical support and incident response actions involving end user devices.

Organizations should consider implementing the components in a test environment first, instead of a production environment, to reduce the likelihood of implementation problems disrupting the production environment. When the components are being deployed into production, organizations should initially use encryption on a small number of hosts. Deploying it to many hosts at once might overwhelm the management servers or identify other bottlenecks through loss of availability. Many of the problems that occur are likely to occur on multiple hosts, so it is helpful to identify such problems either during the testing process or when deploying the first hosts, so that those problems can be addressed before widespread deployment. A phased deployment is also helpful in identifying potential problems with scalability.

Actions that may be prudent to perform before installing storage encryption software on end user devices include the following:

- Ensure that any files to be encrypted can be restored. Examples include backing up user files and having a disk image for the computer’s OS.

- Replace hardware components (e.g., replace an old hard drive) or the whole device if necessary (e.g., equipment that is considered too slow or unreliable).

- Ensure that the OS is secured properly, including that it is fully patched and that other necessary security controls, such as antivirus software, are installed and configured properly. If the OS is not secured properly, the device is more likely to be compromised, which could weaken the protection provided by the storage encryption solution.

- Scan the device for malware and either remove any malware that is detected or rebuild the device.

4.4 Deploy the Solution

Once testing is complete and any issues have been resolved, the next phase of the planning and implementation model involves deploying the solution. A prudent strategy is to gradually migrate devices and users to the new solution. The phased deployment provides administrators an opportunity to evaluate the impact of the solution and resolve issues prior to enterprise-wide deployment. It also provides time for the IT staff (e.g., system administrators, help desk) and users to be trained.

Most of the issues that can occur during deployment are the same types of issues that occur during any large IT deployment. In addition to potential problems described earlier in this publication, another typical issue that is storage encryption-specific is that storage encryption technologies might not work properly on some devices because of incompatibilities with particular hardware configurations.

4.5 Manage the Solution

The last phase of the planning and implementation model is the longest lasting. Managing the solution involves operating the deployed solution and maintaining the security storage architecture, policies, software, and other solution components. Examples of typical actions are as follows:

- Testing and applying patches to storage encryption software. It is beneficial to have at least one host (one for each type of platform) that is used strictly for testing updates. This can help to identify possible conflicts between an update and the normal functions of devices. Software updates should be tested and deployed using the same practices that would be used for updating any other major security controls, such as antivirus software.
- Deploying storage encryption technologies to additional types of devices
- Configuring additional devices to use the technologies
- Performing key management duties (e.g., issuing new credentials, revoking credentials for compromised systems or departing users)
- Performing recovery actions (e.g., regaining access to encrypted data when the authenticator has been lost or the storage media has been damaged)
- Adapting the policies as requirements change. An example is switching to a stronger encryption algorithm or increasing the key size.
- Monitoring the storage encryption components for operational and security issues
- Periodically performing testing to verify that storage encryption is functioning properly
- Performing regular vulnerability assessments
- Receiving notifications from vendors of security problems with storage encryption components, and responding appropriately to those notifications
- Preparing devices for retirement or disposal. Devices and media that use storage encryption technologies should be sanitized or destroyed, even for devices using FDE. Relying on storage encryption to protect data without regularly maintaining the encryption solution is not recommended. For example, suppose that a retired device protected by FDE is sold to an attacker. If a vulnerability in the FDE software is discovered in the future, the attacker may be able to exploit it to access the protected data. Another problem with relying on storage encryption to protect data on retired or disposed devices is that all copies of all keys used for storage encryption would need to be destroyed, which may be very difficult.

User files on the device should be backed up before major maintenance actions are performed, such as installing or upgrading storage encryption software and changing encryption algorithms or key sizes.

Appendix A—Alternatives to Encrypting Storage on End User Devices

Section 3 describes commonly used technologies for storage encryption—full disk, virtual disk and volume, and file/folder encryption. Organizations should not feel compelled to use only these methods to encrypt stored information; there are many other acceptable methods. The following are some examples:

- Applications can encrypt the information that they store. For example, a commercial off-the-shelf backup utility might be capable of encrypting its backups, and a compression utility might have an option to encrypt archives that a user creates. Another example is a database that can be configured to encrypt fields that contain sensitive information. An application could also store sensitive information in an alternate format, such as cryptographic hashes of passwords instead of the passwords themselves.

- Digital rights management (DRM) can be used to restrict who can access particular files and how the files can be used.

- Sensitive information could be accessed only through a virtual machine and stored as part of the virtual machine. If the virtual machine software itself does not provide an encryption capability, the virtual machine data, which is a single file, could be protected through storage encryption software.

In some cases, organizations may decide that the best way to address the problem of protecting sensitive information on end user devices is not to store the information on the devices. Examples of how this might be implemented include the following:

- Preventing access to sensitive information from higher-risk devices, such as mobile devices

- Using a thin client solution, such as terminal services, a thin Web-based application, or a portals, to access the information, and configuring the thin client solution to prohibit file transfers of the sensitive information to the end user device

- Configuring the organization's devices (including desktop computers) to prevent writing sensitive information to removable media, such as CDs or USB flash drives, unless the information is properly encrypted

- Permitting users to access files or databases containing sensitive information only through well-secured applications that restrict access as tightly as possible. For example, suppose that an organization has a database containing thousands of records on employees' benefits. Instead of allowing a user to have full and direct access to the database, which could allow the user to transfer all the database records to the user's device, the organization could permit the user to access only the necessary records and record fields. If the user only needs access to general demographic information, and does not need to access any information related to the employees' identities, then the user would not be able to access any sensitive information.

- Removing unneeded sensitive information from files or databases.

Organizations should be aware that if sensitive information can be viewed from end user devices, the information is still at some risk of exposure, even if it is not stored there. The information could be recorded in screen captures, printed, or monitored by malware, for example.

Organizations should also be aware that the use of general access control mechanisms is typically insufficient to protect sensitive information on end user devices. For example, a password generally cannot be used instead of cryptography to protect stored information. Although requiring a BIOS password can prevent an attacker from booting a computer regularly, the attacker could still access the information by placing the storage media in a different computer. Using an OS password is also ineffective because forensic tools can examine the storage media directly. OS controls such as access control lists may deter users from accessing each other's files, but they can also be circumvented by using forensic tools. Encrypted storage cannot be decrypted by forensic tools.

Appendix B—Glossary

Selected terms used in the publication are defined below.

Container: The file used by a virtual disk encryption technology to encompass and protect other files.

End User Device: A personal computer (desktop or laptop), consumer device (e.g., personal digital assistant [PDA], smart phone), or removable storage media (e.g., USB flash drive, memory card, external hard drive, writeable CD or DVD) that can store information.

File: A collection of information logically grouped into a single entity and referenced by a unique name, such as a filename.

File Encryption: The process of encrypting individual files on a storage medium and permitting access to the encrypted data only after proper authentication is provided.

Filesystem: A mechanism for naming, storing, organizing, and accessing files on logical volumes.

Folder: An organizational structure used by a filesystem to group files.

Folder Encryption: The process of encrypting individual folders on a storage medium and permitting access to the encrypted files within the folders only after proper authentication is provided.

Full Disk Encryption (FDE): The process of encrypting all the data on the hard drive used to boot a computer, including the computer's operating system, and permitting access to the data only after successful authentication with the full disk encryption product.

Malware: A program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system.

Master Boot Record (MBR): A special region on bootable media that determines which software (e.g., operating system, utility) will be run when the computer boots from the media.

Metadata: Information regarding files and folders themselves, such as file and folder names, creation dates and times, and sizes.

Partitioning: The act of logically dividing a media into portions that function as physically separate units.

Pre-Boot Authentication (PBA): The process of requiring a user to authenticate successfully before decrypting and booting an operating system.

Residual Data: Data from deleted files or earlier versions of existing files.

Sector: The smallest unit that can be accessed on media.

Storage Security: The process of allowing only authorized parties to access stored information.

Trusted Platform Module (TPM) Chip: A tamper-resistant integrated circuit built into some computer motherboards that can perform cryptographic operations (including key generation) and protect small amounts of sensitive information, such as passwords and cryptographic keys.

Virtual Disk Encryption: The process of encrypting a container, which can hold many files and folders, and permitting access to the data within the container only after proper authentication is provided.

Volume: A logical unit of storage comprising a filesystem.

Volume Encryption: The process of encrypting an entire volume and permitting access to the data on the volume only after proper authentication is provided.

Whole Disk Encryption: See “Full Disk Encryption”.

Appendix C—Acronyms

Selected acronyms used in the publication are defined below.

AES	Advanced Encryption Standard
BIOS	Basic Input/Output System
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CMAC	Cipher-Based Message Authentication Code
CMVP	Cryptographic Module Validation Program
DRM	Digital Rights Management
EFS	Encrypting File System
EPHI	Electronic Protected Health Information
FDE	Full Disk Encryption
FIPS	Federal Information Processing Standards
FISMA	Federal Information Security Management Act
GLBA	Gramm-Leach-Bliley Act
HIPAA	Health Insurance Portability and Accountability Act
HMAC	Keyed-Hash Message Authentication Code
IT	Information Technology
ITL	Information Technology Laboratory
KB	Kilobyte
MBR	Master Boot Record
NIST	National Institute of Standards and Technology
NTFS	NT File System
NVD	National Vulnerability Database
OMB	Office of Management and Budget
OS	Operating System
PBA	Pre-Boot Authentication
PBE	Pre-Boot Environment
PDA	Personal Digital Assistant
PII	Personally Identifiable Information
PIN	Personal Identification Number
PKI	Public Key Infrastructure
RADIUS	Remote Authentication Dial In User Service
SHA	Secure Hash Algorithm
SIEM	Security Information and Event Management
SP	Special Publication
SSO	Single Sign-On
TPM	Trusted Platform Module
USB	Universal Serial Bus
VPN	Virtual Private Network

Appendix D—Tools and Resources

The lists below provide examples of tools and resources that may be helpful.

Resource Web Sites

Organization	URL
Computer Forensics Tool Testing (CFTT) Project	http://www.cftt.nist.gov/
Cryptographic Module Validation Program (CMVP)	http://csrc.nist.gov/cryptval/
FIPS-Approved Symmetric Key Algorithms	http://csrc.nist.gov/cryptval/des.htm
Full Disk Encryption mailing list and discussion group	http://www.full-disc-encryption.com/
Modes of Operation for Symmetric Key Block Ciphers	http://csrc.nist.gov/CryptoToolkit/modes/

Resource Documents

Document Title	URL
FIPS 140-2, *Security Requirements for Cryptographic Modules*	http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf
FIPS 180-2, *Secure Hash Standard*	http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf
FIPS 197, *Advanced Encryption Standard (AES)*	http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
NIST SP 800-21-1, *Guideline for Implementing Cryptography in the Federal Government*	http://csrc.nist.gov/publications/nistpubs/800-21-1/sp800-21-1_Dec2005.pdf
NIST SP 800-32, *Introduction to Public Key Technology and the Federal PKI Infrastructure*	http://csrc.nist.gov/publications/nistpubs/800-32/sp800-32.pdf
NIST SP 800-38A, *Recommendation for Block Cipher Modes of Operation-Methods and Techniques*	http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf
NIST SP 800-38B, *Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication*	http://csrc.nist.gov/publications/nistpubs/800-38B/SP_800-38B.pdf
NIST SP 800-38C, *Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality*	http://csrc.nist.gov/publications/nistpubs/800-38C/SP800-38C.pdf
NIST SP 800-38D (DRAFT), *Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) for Confidentiality and Authentication*	http://csrc.nist.gov/publications/drafts.html
NIST SP 800-53 Revision 1, *Recommended Security Controls for Federal Information Systems*	http://csrc.nist.gov/publications/nistpubs/800-53-Rev1/800-53-rev1-final-clean-sz.pdf
NIST SP 800-57, *Recommendation for Key Management—Part 1: General*	http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf
NIST SP 800-57, *Recommendation for Key Management—Part 2: Best Practices for Key Management Operation*	http://csrc.nist.gov/publications/nistpubs/800-57/SP800-57-Part2.pdf
NIST SP 800-63, *Electronic Authentication Guideline*	http://csrc.nist.gov/publications/nistpubs/800-63/SP800-63V1_0_2.pdf
NIST SP 800-67, *Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher*	http://csrc.nist.gov/publications/nistpubs/800-67/SP800-67.pdf