ENTERPRISE SYSTEM MANAGEMENT

SECURITY TECHNICAL IMPLEMENTATION GUIDE

Version 1, Release 0

29 October 2004

DRAFT

Developed by DISA for the DOD

This page is intentionally left blank.

TABLE OF CONTENTS

Page

This page is intentionally left blank.

LIST OF FIGURES

This page is intentionally left blank.

1. INTRODUCTION

This Enterprise System Management (ESM) Security Technical Implementation Guide (STIG) provides security configuration guidance for software products designed to deliver enterprise-class system management functions. While the boundaries of the ESM discipline are such that there is no authoritative definition of an ESM product, *Section 2, Enterprise System Management Overview*, provides a generic description of the elements characteristic of most ESM products. *Section 3, Enterprise System Management Security*, provides general guidance for ESM products; specific commercial products are addressed in appendices.

This document is intended be used in conjunction with the other STIGs developed by the Defense Information Systems Agency (DISA). The operating system (OS) STIGs provide crucial guidance for securing the platforms on which the ESM products run. The STIGs that cover database and web server products provide guidance to ensure that those services used by ESM products also support a secure environment.

1.1 Background

For many years, network management products have been used to monitor and manage networks that span wide areas. These products helped to automate repetitive tasks and allow remote configuration changes to be made. Management disciplines and industry standards were eventually created to promote the use and further development of the products. The success of network management products encouraged vendors to develop system management products to provide similar functions for the number of growing individual server and client machines.

While the number of network elements and individual hosts continues to increase, more consistency and efficiency is being sought for management functions. Scalability has been, and continues to be, a significant issue. To address these needs, ESM software is built on the foundation created by network and system management products. The ESM products are designed to automate and centralize the administration, monitoring, operations, and support of applications, systems, and platforms on an enterprise scale.

From the perspective of security, ESM products offer both benefits and burdens. The use of ESM products to track software usage and to deploy security updates provides obvious benefits. The extent of an organization’s vulnerability to a specific worm or virus can be quickly understood. Server and client hosts can be made more secure against attack with less effort and in a shorter period of time. The need for System Administrators to manually install updates on a large number of machines is mostly eliminated.

However, the configuration and use of powerful ESM products can increase vulnerability. Because the ESM software frequently runs with elevated privileges and has a large span of control over hosts, a compromise of ESM elements could lead to widespread problems. Part of an attack might include disabling the ESM software so that security patches cannot be deployed, leaving a large number of hosts open to further attack. Another attack strategy could include using configuration management functions in the ESM product to deliver corrupt software or to maliciously alter configuration settings on a large scale.

The goal of this document is to provide guidance that allows the power of ESM products to be used, while preventing that power from being exploited to degrade the confidentiality, integrity, or availability of the systems being managed. It should be noted that Field Security Operations (FSO) support for the STIGs, Checklists, and Tools is only available to DOD Customers.

1.2 Authority

DOD Directive 8500.1 requires that “all IA and IA-enabled IT products incorporated into DOD information systems shall be configured in accordance with DOD-approved security configuration guidelines” and tasks DISA to “develop and provide security configuration guidance for IA and IA-enabled IT products in coordination with Director, NSA.” This document is provided under the authority of *DOD Directive 8500.1*.

The use of the principles and guidelines in this STIG will provide an environment that meets or exceeds the security requirements of DOD systems operating at the MAC II Sensitive level, containing unclassified but sensitive information.

1.3 Scope

This document describes security requirements to be applied to ESM products used in DOD environments. The information is designed to assist Security Managers, Information Assurance Managers (IAMs), Information Assurance Officers (IAOs), and System Administrators (SAs) with the creation of more secure ESM configurations. As noted in the previous section, application of the requirements is intended to provide a certain level of assurance. Individual sites must determine if this level of assurance is appropriate to their environment.

This document provides both general and product-specific security guidance. As noted elsewhere in this document, vendor implementation of ESM functions does vary; and most commercial products provide only subsets of all the functions generally associated with ESM.

Specific guidance is provided for the following:

- Tivoli enterprise management products
- Microsoft Systems Management Server 2003

1.4 Writing Conventions

Throughout this document, statements are written using words such as “**will**” and “**should**.” The following paragraphs are intended to clarify how these STIG statements are to be interpreted.

A reference that uses “**will**” implies mandatory compliance. All requirements of this kind will also be documented in the italicized policy statements in bullet format, which follow the topic paragraph. This will make all “**will**” statements easier to locate and interpret from the context of the topic. The IAO will adhere to the instruction as written. Only an extension issued by the Designated Approving Authority (DAA) will table this requirement. The extension will normally have an expiration date, and does not relieve the IAO from continuing their efforts to satisfy the requirement.

A reference to “**should**” is considered a recommendation that further enhances the security posture of the site. These recommended actions will be documented in the text paragraphs but not in the italicized policy bullets. Nevertheless, all reasonable attempts to meet this criterion will be made.

For each italicized policy bullet, the text will be preceded by parentheses containing the italicized Short Description Identifier (SDID), which corresponds to an item on the checklist and the severity code of the bulleted item. An example of this will be as follows: “(*G111: CAT II*).” If the item presently has no Potential Discrepancy Item (PDI), or the PDI is being developed, it will contain a preliminary severity code and "N/A" for the SDID (i.e., “(*N/A: CAT III)*”).

1.5 Vulnerability Severity Code Definitions

Category I	Vulnerabilities that allow an attacker immediate access into a machine, allow superuser access, or bypass a firewall.
Category II	Vulnerabilities that provide information that has a high potential of giving access to an intruder.
Category III	Vulnerabilities that provide information that potentially could lead to compromise.
Category IV	Vulnerabilities, when resolved, will prevent the possibility of degraded security.

1.6 Information Assurance Vulnerability Management (IAVM)

The DOD has mandated that all IAVMs are received and acted on by all commands, agencies, and organizations within the DOD. The IAVM process provides notification of these vulnerability alerts and requires that each of these organizations take appropriate actions in accordance with the issued alert. IAVM notifications can be accessed at the Joint Task Force Global Network Operations (JTF-GNO) web site: http://www.cert.mil.

1.7 STIG Distribution

Parties within the DOD and Federal Government's computing environments can obtain the applicable STIG from the Information Assurance Support Environment (IASE) web site. This site contains the latest copies of any STIG, as well as checklists, scripts, and other related security information.

The NIPRNet URL for the IASE site is http://iase.disa.mil/. The National Institute of Standards and Technology (NIST) site is http://csrc.nist.gov/pcig/cig.html. Access to the STIGs on the IASE web server requires a network connection that originates from a **.mil** or **.gov** address. The STIGs are available to users that do not originate from a **.mil** or **.gov** address by contacting the FSO Support Desk at DSN 570-9264, commercial 717-267-9264, or e-mail to fso_spt@ritchie.disa.mil.

1.8 Document Revisions

Comments or proposed revisions to this document should be sent via e-mail to fso_spt@ritchie.disa.mil. DISA FSO will coordinate all change requests with the relevant DOD organizations before inclusion in this document.

2. ENTERPRISE SYSTEM MANAGEMENT OVERVIEW

2.1 General Overview

Enterprise System Management (ESM) can be described as the automation of activities involved in administering, monitoring, operating, and supporting multiple information systems. The systems are commonly of various platform types and connected by one or more networks. Geographical configurations may vary from many systems collocated to a few systems distributed over a wide area. ESM commonly involves automating repetitive tasks and remotely performing activities that would otherwise be performed by local System Administrators. This document provides implementation guidance for security controls that apply to ESM applications.

Because the scope of ESM functions is broad, it is not always obvious that a specific product is an ESM application. In fact, there is a close functional similarity between ESM applications and network management applications that generally preceded them. Network management applications traditionally focused on network devices and not application hosts, but that focus has gradually been softened. In some cases, ESM product suites include network, host, and application management components.

This section provides an overview of what an ESM application does and some of the elements that might be used by the application. This should provide a background to determine if a given application should be called an ESM application and should be subject to the requirements in this document.

There are two particular concepts that impact the security requirements for DOD implementation of ESM applications. The first of these is the notion of an enclave. The Committee on National Security Systems (CNSS) *National Information Assurance (IA) Glossary* defines an enclave as the “collection of computing environments connected by one or more internal networks under the control of a single authority and security policy, including personnel and physical security.” This is discussed in detail in the *Enclave Security STIG* and the NSA *Information Assurance Technical Framework*. This impacts ESM implementation because some requirements vary according to whether an ESM application spans enclave boundaries. Because applications that operate entirely within an enclave boundary are protected by the enclave perimeter defenses, less stringent requirements apply to those implementations. However, most ESM applications are designed with the capability to operate across networks that could easily span enclave boundaries, so it is necessary to determine if a specific implementation crosses enclave boundaries when applying the requirements.

The second concept impacting DOD ESM implementations is the definition of IA and IA-enabled products as discussed in *DOD Directive 8500.1* and *DOD Instruction 8500.2*. The definition is detailed in *Section 3.2.1, Application Design Characteristics*. This impacts ESM implementation because additional requirements apply to IA and IA-enabled products. Those products provide some level of security services so their implementation requires greater caution. While most ESM applications are not IA products, many are IA-enabled because they implement access controls over ESM administrative functions. Therefore it is necessary to determine if an ESM application is an IA or IA-enabled product when applying the requirements.

A final general note concerning ESM applications and the importance of their security is necessary. Certain types of ESM applications have the ability to generate very significant impact to the systems under their control. Acting as automated System Administrators, these applications may change operational status and configuration, report state information, and distribute critical security data such as suspicious event notices and virus signature updates. These functions generally take on the importance of the availability of the systems on which they are performed. In short, an ESM application may become a mission-essential application. In addition the applications have privileged access in order to accomplish their functions. Therefore, the security of these applications demands careful implementation in order to ensure the integrity and availability of the systems they manage.

The following sections describe functional areas and implementation elements that are common to ESM applications. While there is no practical consensus that these items must be found in every ESM application, there are some standards that provide a common basis for discussion.

2.1.1 ESM Functional Areas

The International Telecommunication Union (ITU) Open Systems Interconnection (OSI) standards relating to management functions provide a helpful framework in which to discuss ESM applications. Recommendation X.700, “Management framework for Open Systems Interconnection (OSI) for CCITT applications” describes five Management Functional Areas (MFAs): Fault management, Configuration management, Accounting management, Performance management, and Security management. This characterization is sometimes referred to as the FCAPS model, after the first initial of the name of each of the areas. The following figure shows these areas with some of their associated functions from the X.700 Recommendation:

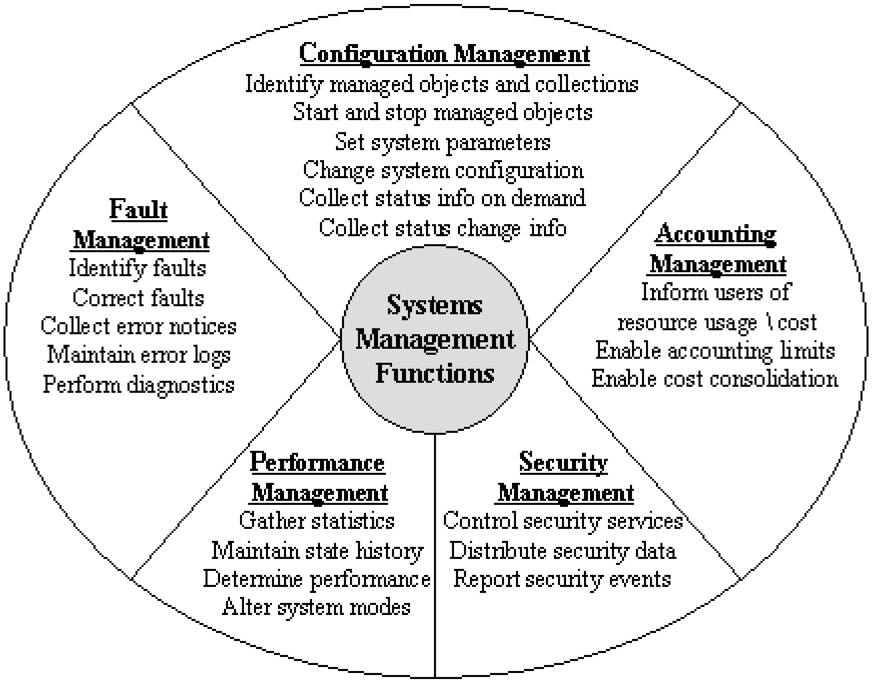

Figure 2-1. ESM Management Functional Areas

Two important notes apply to the FCAPS model:

- Though there are five functional areas, most products rely on shared mechanisms to deliver their functions. For example, a common mechanism is likely to be used to collect fault data as well as performance data.

- This same model is used to describe functional areas used for network management. ITU Recommendation M.3400, "TMN management functions" uses the same terminology in its discussion of functions.

In terms more specific to ESM, the functional areas may be implemented in the following ways:

- Fault Management - ESM applications providing centralized monitoring and reporting of function availability on client systems fit in this area. Some applications also perform diagnostic and corrective action on clients. Applications that are described as offering generic event management are generally providing fault management functions.

- Configuration Management - ESM applications providing remote management of client systems fit in this area. These applications may change the operating parameters or state of the systems under their control. Applications providing hardware and software inventory and applications that deploy software and other data also fit in this area.

- Accounting Management - ESM applications that monitor system resource usage for the purpose of limiting consumption or collecting data for cost allocation~~,~~ fit in this area.

- Performance Management - ESM applications providing longer-term pictures of client system availability and workload management data fit in this area.

- Security Management - ESM applications providing centralized monitoring and reporting of security events fit in this area. Data may be consolidated from firewalls, intrusion detection systems, or other host security services. Applications with sophisticated event correlation engines may offer the capability to change security parameters or to isolate hosts that may have been compromised.

ESM products are quite likely to span these functional areas because the mechanisms and data used are often common. Most ESM products use elements of configuration management to organize their tasks and data. With the possible exception of security management functions, the requirements for securing ESM applications are generally the same for all the functional areas.

2.1.2 ESM Implementation Elements

Enterprise System Management products are like other categories of software products in terms of the diverse ways in which they are designed and implemented. The concept of ESM has developed concurrently with open system standards, but vendors frequently choose proprietary versions of elements to deliver ESM functions. While proprietary solutions may have efficiency or performance advantages, their unique elements can create interoperability issues when combined and they certainly lead to confusion when discussed.

This section provides a very brief discussion of some of the implementation elements for ESM applications. The objective is to provide terminology that can be used to describe common elements that exist in diverse vendor implementations. Some of the organizations and the standards they promote are mentioned as context for later ESM application discussions.

The body of this document uses generic terms to discuss ESM application elements. In the vendor-specific appendices, these generic terms are related to the vendor terminology as appropriate.

- Manager - An ESM manager is an application that usually runs on a dedicated host server. The application is responsible for organizing and controlling functions that are carried out on, or for, clients within the manager’s span of control. ESM manager applications include elements that maintain data about the clients. The applications usually include support that enables remote administration and status displays.

- Agent - An ESM agent is an application, usually small, that runs on a client host that may be called a managed element. The agent application is responsible for collecting data, performing actions on the managed element, and responding to commands issued by an ESM manager. Agents may maintain continuous or periodic network connections to their associated manager.

- Console - An ESM console is an application that provides an interface to the ESM manager for an administrator to enter commands and display data. In current products, ESM consoles are usually implemented as graphical interfaces that are run on hosts remote to the manager. Some ESM consoles are implemented as web applications using a browser to provide the graphical user interface.

- Management Data Repository - An ESM management data repository is the logical database used by an ESM manager to store information about the clients on which agents are deployed. An object-oriented approach is common, representing clients as objects associated with attributes and methods that apply to them. The repository may also contain definitions of configuration settings that are to be applied to groups of clients.

There are a number of ways in which these elements can be assembled as an ESM application. The elements are arranged in a hierarchy with the ESM manager residing on a server at the top, and the clients with ESM agents at the bottom. The following figure shows two common implementations. The configuration on the left shows a basic example. On the right, an additional tier of manager servers is shown. The multiple tier management configuration is useful for large organizations that have many clients that are distributed over a wide area. The middle tier of managers can be co-located with the clients. This distributes the workload from the top tier to the middle tier servers and reduces the number of network connections to the top tier server. These characteristics make this solution scalable to very large environments.

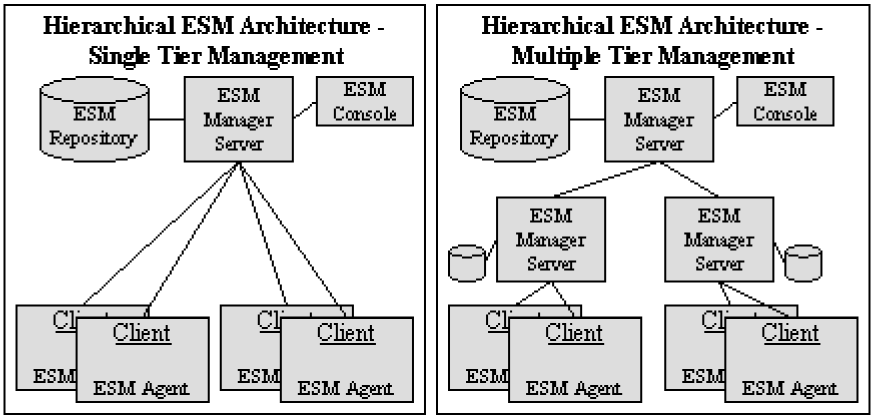

Figure 2-2. Hierarchical ESM Architectures

Another possible configuration involves the use of multiple ESM manager servers for each client with an ESM agent. The following figure shows a basic example. The manager servers operate in parallel, each with specific areas of responsibility. The managers may communicate in order to share information about the clients.

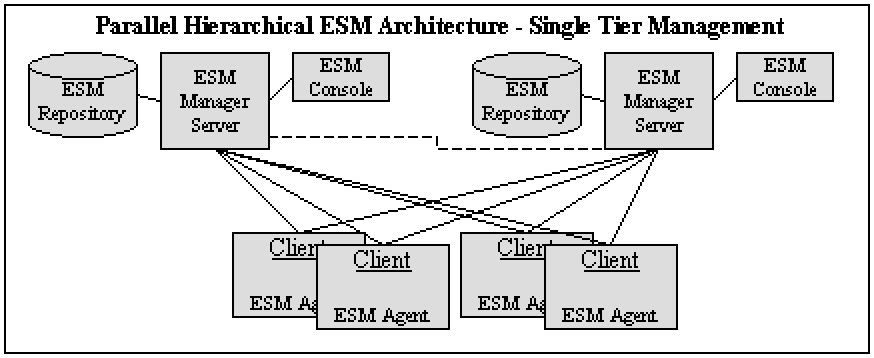

Figure 2-3. Parallel Hierarchical ESM Architecture

Common to all of these configurations is the transfer and storage of management data. This data is carried by network connections between managers and agents. It is stored in management data repositories. Several standards efforts have been undertaken in order to promote greater interoperability among products from different vendors.

A discussion of the standards related to management data can be helpful in identifying ESM applications. However, this is a complex subject that requires extensive material for explanation. A review of the documents from the Internet Engineering Task Force (IETF) (http://www.ietf.org/) and the Distributed Management Task Force (DMTF) (http://www.dmtf.org/) is strongly recommended. The following brief notes identify the most prominent standards that may be employed in ESM applications.

- SNMP - The Simple Network Management Protocol (SNMP) is defined by the IETF as "a simple protocol by which management information for a network element may be inspected or altered by logically remote users". The objective for SNMP is to "provide a simple, workable architecture and system for managing TCP/IP-based internets and in particular the Internet." Although SNMP was initially intended for management of network devices, it has been widely adopted for use in diverse management environments. SNMP is described by many documents; RFC1157 provides the basic definition.

- CMIP - The Common Management Information Protocol (CMIP) is a protocol for network management defined by the International Telecommunication Union (ITU) as part of the Open Systems Interconnection (OSI) management standards. CMIP was designed for OSI-based communication protocols and was intended to replace SNMP. CMIP Over TCP (CMOT) was defined for use with IP-based networks. RFC1189 defines a network management architecture using CMOT. CMIP and CMOT have not been widely adopted.

- DMI - The Desktop Management Interface (DMI) specification was defined by the Distributed Management Task Force (DMTF) as a framework for the management of Desktop systems and servers. The DMI architecture consists of layers that allow interoperability between management server and client implementations from different vendors. The DMTF has announced an "end of life" for the DMI standard in favor of more broadly focused management standards.

- CIM - The Common Information Model (CIM) incorporates a specification and a schema defined by the DMTF as "a conceptual information model for describing managed entities, their composition, and relationships." An object-oriented architecture is used in the definition and structure of the data. The CIM management schema is divided into a Core Model, Common Models, and extension schemas. The Core Model covers elements applicable to all management areas. The Common Models cover elements common to particular management areas including systems, applications, databases, networks, and devices. Technology specific extensions can be created through the extension schemas. DMTF incorporated the use of directories in the CIM through the Directory Enabled Networks (DEN) initiative. DEN provides a mapping for CIM to an LDAP structure to enable the use of directories to locate management information and access management data. Both the DMTF and the IETF have done work to enable the representation of policy information as an extension to the CIM. The DMTF CIM Policy Model and the IETF Policy Core Information Model (PCIM) (RFC3060 and RFC3460) are designed to allow the expression of policy in vendor and device-independent terms that can be translated by software into device-specific configuration changes that implement the policy.

- WBEM - Web-Based Enterprise Management (WBEM) is defined by the DMTF as "a set of management and Internet standard technologies developed to unify the management of enterprise computing environments". WBEM is expressed primarily by three core standards: the Common Information Model (CIM), the Representation of CIM in XML specification, and the CIM Operations over HTTP specification. The Representation of CIM in XML specification provides the format for encoding CIM declarations (classes, instances, and qualifiers) and CIM messages in Extensible Markup Language (XML). The CIM Operations over HTTP specification defines "a mapping of CIM operations onto HTTP" to support the transport of CIM messages. Vendors have developed implementations of WBEM to enable management of their products. Microsoft's Windows Management Instrumentation (WMI) is one well-known implementation of WBEM standards for Windows environments.

Concise descriptions of some of these technologies can be found in the Carnegie Mellon Software Engineering Institute's Software Technology Roadmap (http://www.sei.cmu.edu/str/).

To close the discussion of ESM implementation elements, it is reasonable to briefly raise two issues: the role of customization and the logical points of vulnerability.

The implementation of ESM requires some flexibility to account for the variety of objects and configurations that should be managed. In addition, ESM applications are frequently called on to perform ad hoc and non-standard operations on a one-time basis. Implementation flexibility is achieved by customizing ESM functions. Coding programs that are compiled into binary executables, as well as scripts that are interpreted at run time, may be used to accomplish this. In any case, these items become elements, though often temporary, of a specific ESM implementation. It is important to recognize that these “local” elements have to be considered when evaluating the security characteristics of an ESM implementation.

Finally, it is important to recognize some logical points of vulnerability that arise from elements in the ESM architecture. Briefly, these areas include:

- ESM managers and agents are applications that might be vulnerable due to errors in coding or configuration. ESM application failures can have negative impacts that are similar to administrator errors, but on a much larger scale.

- ESM data repositories are likely to hold data that represents, at a minimum, sensitive information. This is because that data includes specific configuration information. Disclosure or corruption of that data could lead to more serious vulnerabilities.

- ESM network communications are often critical to operations. Fault data that is not received by the manager or configuration commands that do not arrive at the agent can result in a loss of function that might be critical to maintaining system availability.

Reducing and eliminating vulnerabilities in ESM applications requires correct configuration and attention to vulnerability information for all of its elements. During 2002 serious vulnerabilities were discovered in a wide variety of vendor implementations of SNMP. This occurrence provided a good example of the potential impact of vulnerabilities in management applications. A failure to install the patches for these vulnerabilities could have seriously degraded the availability of a significant portion of the network management infrastructure. Avoiding such problems is essential to support DOD’s net-centric operations.

3. ENTERPRISE SYSTEM MANAGEMENT SECURITY

3.1 Introduction

Enterprise System Management applications commonly perform sensitive functions, requiring elevated privileges, on multiple hosts. Some of these functions, such as security patch management, are essential to operations because they help to maintain secure environments. The conclusion from these facts is that security controls on ESM applications are critical.

While the impact can vary significantly, a compromise of an ESM application could result in a serious loss of the confidentiality or integrity attributes of hosts. Beyond that, the loss of a particular ESM function could result in a loss of an essential information assurance function that would effectively render the affected hosts inaccessible.

This section provides the general security requirements for the implementation of ESM products. Because of the diversity of products, these requirements are somewhat generic in nature. The product sections of this document further explain the requirements as they relate to specific implementations.

This section is broken into subsections that align with the Information Assurance (IA) control subject areas defined in *Department of Defense (DOD) Instruction 8500.2, Information Assurance (IA) Implementation*. These subject areas are as follows:

- Security Design and Configuration
- Identification and Authentication
- Enclave and Computing Environment
- Enclave Boundary Defense
- Physical and Environmental
- Continuity
- Vulnerability and Incident Management.

Some of these areas are further divided to provide a more cohesive presentation. The Personnel subject area is not included as there are no controls in that subject area that are addressed in an ESM product security review.

It is important to understand some of the specific terminology used in this section:

- ESM Administrator - An ESM administrator is responsible for configuring and operating one or more ESM products. An ESM administrator, using an ESM product, may perform some or all of the functions of a host System Administrator (SA).

- I&A Services - Identification and authentication (I&A) services are used to establish and verify the identity of an entity. Secure host operating systems always perform I&A services. In determining the privileges that an individual ESM administrator can exercise, some ESM products rely on host I&A services while others implement their own versions. If an ESM product manages some form of password file, that product provides I&A services.

3.2 Security Design and Configuration

This section describes ESM security requirements based on applicable IA controls in the Security Design and Configuration subject area. These requirements address three general areas: design characteristics, implementation and configuration, and network access.

3.2.1 Application Design Characteristics

The most effective security characteristics of an application are integrated into its design. In order to ensure objectively that the desired characteristics are present and properly implemented, a formal evaluation and validation process is necessary. The requirements in this section are designed to achieve two goals:

- To ensure that IA and IA-enabled ESM applications have been through the standardized evaluation processes and meet the requirements for products used in DOD information systems

- To ensure that software for which there is no formal review, extension, or repair process is not used unless specific safeguards are employed.

The National Security Telecommunications and Information Systems Security Committee (NSTISSC), now known as the Committee on National Security Systems (CNSS), issued *National Security Telecommunications and Information Systems Security Policy (NSTISSP) No. 11* to specify evaluation and validation requirements related to the acquisition of IA and IA-enabled IT products. DOD Instruction (DODI) 8500.2 reiterates the requirements. A detailed discussion of the requirements can be found in the *Enclave Security STIG*.

It must be noted that some of the requirements in this section apply to ESM applications that are classified as IA or IA-enabled products. *DODI 8500.2* defines these terms as follows:

- An IA product is a product or technology whose primary purpose is to provide security services (e.g., confidentiality, authentication, integrity, access control, non-repudiation of data); correct known vulnerabilities; and/or provide layered defense against various categories of non-authorized or malicious penetrations of information systems or networks.

- An IA-enabled product is a product or technology whose primary role is not security, but which provides security services as an associated feature of its intended operating capabilities.

A few ESM applications, such as those that perform provisioning of user identities, are IA products. Most ESM applications are IA-enabled because they provide security services such as controlling the authorization of ESM administrators to perform privileged functions. Both types of products are required to meet the requirements that follow.

- *(EGA.0010: CAT III) The IAM will ensure that the acquisition of all IA and IA-enabled Government-Off-the-Shelf (GOTS) ESM products meets the applicable NSA evaluation and validation requirements specified in NSTISSP No. 11 and DODI 8500.2.*

- *(EGA.0020: CAT III) The IAM will ensure that the acquisition of all IA and IA-enabled Commercial-Off-the-Shelf (COTS) ESM products meets the applicable Common Criteria, NIAP, or FIPS evaluation and validation requirements specified in NSTISSP No. 11 and DODI 8500.2.*

Specific system configuration data that is collected, transmitted, and stored by ESM applications could be useful to a potential intruder trying to exploit vulnerabilities on the subject systems. To deter this threat, the ESM data needs to be protected at the same confidentiality level as the systems on which that data is based. IA and IA-enabled COTS and GOTS ESM products provide a level of protection through their administrator access controls. In order to ensure that those controls are strong enough for the confidentiality level of the data, the products must meet the appropriate high, medium, or basic robustness levels as defined in *DODI 8500.2*.

- *(EGA.0030: CAT III) The IAM will ensure that all IA and IA-enabled COTS and GOTS ESM products that manage data from sensitive systems meet the medium robustness requirements defined in DODI 8500.2 when any of the following is true:*

  - *ESM data traverses public networks*
  - *ESM data resides on systems that are accessible by individuals not authorized to access the information.*

Software that is classified as public domain, freeware, or shareware represents a risk to information systems because the Government does not have access to the original source code to review, extend, or repair it when needed. To minimize this risk, specific conditions must be met before software in these classes is used.

- *(EGA.0040: CAT III) The IAM will ensure that binary or machine executable public domain software and other software with limited or no warranty (such as those known as freeware or shareware) is not used to fulfill an ESM function unless the following conditions are met:*

  - *The software is necessary for mission accomplishment and there are no alternative IT solutions available.*

  - *The software is assessed for information assurance impacts and approved for use by the DAA.*

It should be noted that this specific restriction does not apply to open source software. It is permissible to use open source software as long as it conforms to the same DOD policies that govern COTS and GOTS software. This includes those requirements relating to IA and IA-enabled components. Specific guidance is found in the DOD Memorandum, *Open Source Software (OSS) in the Department of Defense (DOD).*

3.2.2 Application Implementation and Configuration

Even a well-designed application can open or be subject to security vulnerabilities if it is not implemented properly. In this instance, implementation refers to how an application is installed and maintained in the environment as well as the configuration settings for the application. The requirements in this section are designed to ensure that ESM applications are installed and configured in a secure manner and that documented processes are used to maintain that secure configuration.

Some ESM applications perform management functions by having a server transmit and schedule the execution of code on client hosts. When used with appropriate controls such as mutual authentication, this practice is acceptable. However, the use of certain types of mobile code is prohibited as described by the following requirements defined in *DODI 8500.2*:

- Emerging mobile code technologies that have not undergone a risk assessment by NSA and been assigned to a Risk Category by the DOD CIO must not be used.

- Category 1 mobile code must be signed with a DOD-approved PKI code-signing certificate; use of unsigned Category 1 mobile code is prohibited; use of Category 1 mobile code technologies that cannot block or disable unsigned mobile code (e.g., Windows Scripting Host) is prohibited.

- Category 2 mobile code, which executes in a constrained environment without access to system resources (e.g., Windows registry, file system, system parameters, network connections to other than the originating host), may be used.

- Category 2 mobile code, which does not execute in a constrained environment, may be used when obtained from a trusted source over an assured channel (e.g., SIPRNET, SSL connection, S/MIME, code signed with a DOD-approved code signing certificate).

- Category 3 mobile code may be used.

The *Enclave Security STIG* provides a general description of mobile code. DOD Memorandum, *Policy Guidance for Use of Mobile Code Technologies in Department of Defense (DOD) Information Systems*, provides specific details on the assignments of technologies to individual mobile code categories.

- *(EGA.0050: CAT III) ESM administrators will ensure that remote management functions used in ESM operations are in compliance with the mobile code requirements of DODI 8500.2 and the Enclave Security STIG.*

As documented in other sections of this document, certain types of ESM data require cryptographic protection. Encryption, key exchange, digital signature, and hash algorithms are used in various cryptographic services to provide this protection. Proprietary or less robust commercial algorithms cannot be used because their level of protection may be too weak. Product implementations that have not been appropriately evaluated and validated might not provide the intended protection. The use of NIST-approved implementations ensures the appropriate strength and correct implementation of a cryptographic service.

- *(EGA.0060: CAT II) ESM administrators will ensure that ESM software is configured to use FIPS 140-2 approved encryption, key exchange, digital signature, and hash algorithms for data storage and transmission.*

Most ESM applications provide automation of configuration management tasks for ESM clients as a primary function. However, the products are tools that simplify tasks; they cannot perform impact evaluation, scheduling, and strategic direction setting responsibilities that are required for a complete configuration management process. If configuration changes are performed without satisfying these responsibilities, the availability, integrity, and confidentiality of the ESM clients might be compromised.

DODI 8500.2 specifies the implementation of a configuration management process that includes:

- Formally documented CM roles, responsibilities, and procedures to include the management of IA information and documentation;

- A configuration control board that implements procedures to ensure a security review and approval of all proposed DOD information system changes, to include interconnections to other DOD information systems;

- A testing process to verify proposed configuration changes prior to implementation in the operational environment; and,

- A verification process to provide additional assurance that the CM process is working effectively and that changes outside the CM process are technically or procedurally not permitted.

- *(EGA.0070: CAT III) The IAM will ensure that ESM software that implements configuration changes is used only in conjunction with a documented configuration management (CM) process.*

ESM applications generally provide functions that ensure continuing and efficient operation of information systems. Applications that provide identification and authentication (I&A) services may even be part of the operational software required to control access to a system. Because of these operational roles, an inventory of the ESM products being used is essential to adequate configuration management and disaster recovery planning.

- *(EGA.0080: CAT III) The IAM will ensure that a current and comprehensive baseline inventory that includes all ESM software is maintained by the CCB and as part of the C&A documentation and that a copy of the inventory is stored in a fire-rated container or otherwise not collocated with the original.*

Because ESM applications typically have the ability to impact the operational condition or monitoring capability for client systems, it is critical to ensure that libraries containing the software elements and configuration controls are protected from malicious or unintentional change. Without proper access controls, negative impacts ranging from denial of service to the introduction of malicious code are possible.

Although it is of less immediate concern for operations, it is also important to ensure that proprietary vendor code and data is protected in accordance with applicable licenses or agreements. For example, use of an application by more than the agreed-upon number of users could expose the Government to additional costs or the loss of rights to use the product.

ESM software often includes one or more programs with the ability to perform privileged functions. These functions may directly impact system availability or the integrity or confidentiality of data on client systems. To prevent unauthorized use, access to the programs providing the privileged functions has to be restricted.

The following requirements ensure that ESM software libraries and configuration files retain their integrity and that access to privileged programs is properly restricted.

- *(EGA.0090: CAT II) The IAO will ensure that access to ESM software libraries (including executable and configuration files) is limited so that:*

  - *Only authorized ESM application processes, ESM administrators, or SAs that require it, have update access.*

  - *Only authorized ESM application processes, ESM administrators, SAs, or other users that require it and are within applicable license agreements, have read access.*

- *(EGA.0100: CAT II) The IAO will ensure that access to ESM software libraries is limited so that only authorized ESM application processes, ESM administrators, or SAs can execute privileged programs.*

As with all applications, a coding error may be found in ESM software that results in security vulnerabilities. Policies vary, but if the affected software is no longer supported, the vendor does not provide fixes to address the problem. In addition, the vendor may not even confirm whether a security vulnerability exists in an unsupported release, leaving users without assurance that their configurations are secure.

To address these issues, it is necessary to ensure that only supported software is used. When a vendor announces that support is being discontinued, an upgrade or removal plan must be developed.

- *(EGA.0110: CAT I) The IAO will ensure that ESM software is removed or upgraded prior to the vendor dropping support.*

- *(EGA.0120: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading ESM software prior to the date the vendor drops security patch support.*

Many ESM products incorporate data repositories such as those used to store hardware and software inventory data. The ESM product architecture may allow partitioning of the repository management from user interfaces used to display and report the data. In practical terms this means that programs that display the data execute on one host and the database management system (DBMS) holding the data can reside on a different host. This type of architecture can provide security benefits in two ways. It provides a more controlled access path to the data and it allows for further restrictions of access to the platform on which the data resides.

The *Database Security Technical Implementation Guide* recommends that any DBMS be installed on a host system dedicated to its support. By separating the DBMS sever, access to that platform can be more finely controlled, resulting in reduced exposure to vulnerabilities in the DBMS software. *DODI 8500.2* requires a physical or logical separation of user interface services from data storage and management services.

- *(EGA.0130: CAT III) ESM administrators will ensure that ESM components are implemented to logically or physically separate the user interface elements from data storage and management elements.*

It is recognized that some products and environments do not support this type of partitioning. These cases are addressed through product-specific guidance in this document.

For the ESM products that provide specific IA functions, a special requirement exists to isolate the security support structure. Products such as those that perform provisioning of user identities or enterprise resource access control require special attention due to the potential impact of compromises of the integrity or availability of the security services.

Isolation of the security support structure is best provided through the use of separate partitions or domains that allow control of access to, and integrity of, the hardware, software, and firmware used. In operational terms this can mean dedicated hosts for ESM IA applications. However, this requirement applies only where the nature or architecture of a specific application supports it.

- *(EGA.0140: CAT II) ESM administrators and IAOs will ensure that ESM hardware and software components that perform security functions, including but not limited to user account administration and resource access control, are isolated by logical or physical means.*

3.2.3 Network Access

The use of networks is inherent to ESM applications in their performance of remote monitoring, reporting, and configuration functions. Unfortunately, the design of some older applications was based on the assumption that data transmission would occur over closely controlled or closed networks with minimal security exposure. This design assumption has led to serious security issues when applied to current networks, especially those with higher levels of interconnection and associated risk.

The requirements in this section are intended to ensure that the combination of ports, protocols, and services (PPS) used by ESM applications is consistent with secure practices identified in DOD network security guidance, including *DOD Instruction 8551.1, Ports, Protocols, and Services Management (PPSM)* and the associated *Ports, Protocols, and Services (PPS) Assurance Category Assignments List*. The *PPS Assurance Category Assignments List* provides detailed guidance for specific services and port numbers. The following are extracts of the PPS usage principles from the PPSM instruction:

- PPS are assessed for vulnerabilities and assigned to one of three assurance categories: RED, YELLOW, or GREEN.

- Ports, Protocols, and Services designated as RED have a low level of assurance. These PPS implemented in applications expose DOD networks to an unacceptable level of risk for routine use. A RED PPS will only be allowed when approved by the DISN DAAs for a specific DOD information system under defined conditions and restrictions and if no suitable alternative exists.

- Ports, Protocols, and Services designated as YELLOW have a medium level of assurance. These PPS expose DOD networks to an acceptable level of risk for routine use only when implemented with the required mitigation strategy and approved by the DISN DAAs for a specific DOD information system.

- Ports, Protocols, and Services designated as GREEN have a high level of assurance. These PPS are considered best security practices and are recommended for use when implemented with the required mitigation strategy and approved by the DISN DAAs for a specific DOD information system.

- *(EGA.0150: CAT II) The IAO will ensure that ESM use of specific network ports, protocols, and services conforms to the requirements of the DOD Ports, Protocols and Services Management (PPSM) instruction and the Ports, Protocols, and Services (PPS) Assurance Category Assignments List.*

- *(EGA.0160: CAT III) If an ESM application traverses a DOD enclave boundary, the ESM administrators will ensure that the ESM application is registered through the designated DOD Component POC as defined by the PPS homepage at www.cert.mil/portsandprotocols.*

Although some COTS ESM products do not conform to the specified PPS restrictions, there are some potential mitigating implementations:

- The requirements apply unconditionally to ESM implementations that traverse enclave boundaries. However, for implementations in which network traffic is contained within a DOD Component enclave (e.g., does not traverse a backbone network such as the NIPRNET), the requirements are reduced to strong recommendations.

- The requirements also drop to the level of strong recommendations where ESM products are deployed in a closed network such as the DISA Computing Services Out-of-Band (OOB) network. Such a network would have to be equipped with appropriate security controls to limit protocol access to management servers and clients and authenticated Virtual Private Networks (VPN).

The product specific sections of this document provide requirements for individual applications, but there is one additional issue to note. Some ESM applications may provide for remote installation using PPS that are normally not permitted. It is strongly recommended that products with this design be installed locally rather than remotely. However, the use of a coordinated installation window that uses the restricted PPS for a period of a few hours is not prohibited by this document as long as the traffic can be restricted to the network addresses of the source and target hosts.

3.3 Identification and Authentication

This section describes ESM security requirements based on applicable IA controls in the Identification and Authentication subject area. These requirements address items that are used to identify and authenticate a user and as input to encryption processes. These items include account IDs, PKI certificates, authenticators such as passwords, and symmetric and asymmetric keys.

The concept of identification and authentication (I&A) services was defined earlier, but deserves a note here. An I&A service establishes a user’s identity and verifies that the requesting user is correctly associated with that identity. Because of the sensitive nature of this service, some of the requirements stated here apply specifically to ESM applications that provide that service.

A final general note on I&A services involves the choice where multiple options exist. Some products allow a choice between using host or application-based I&A services. A single point of security control is generally better because it simplifies account maintenance and auditing, and it provides an integrated point for assigning privileges. Therefore, when the option exists and there are no significant security advantages to the contrary, host I&A services should be implemented instead of the equivalent application services.

3.3.1 Individual Identification

The concept of individual identification is basic to access control decisions and to auditing. With the possible exception of data or systems that are assigned a confidentiality level of public, individual identification is always necessary. Although in some cases it is acceptable to make access control decisions at a group level, meaningful auditing always requires individual identification.

In almost every case, ESM administrator accounts have access to sensitive data and privileged functions. If this access is not sufficiently controlled, the confidentiality, integrity, and availability attributes of one, many, or all of the hosts executing components of an ESM application could be compromised. If access to sensitive data or a privileged function cannot be attributed to a specific user, it may be impossible to determine the source of a compromise or attack. An individual identifier, along with an associated authenticator, provides access control and audit capability.

- *(EGB.0010: CAT II) The IAO will ensure that each ESM user account is associated with an individual identifier, such as a unique token or user login ID, and a password.*

Although user IDs and passwords may currently be the most common form of identification and authentication, newer technologies with greater strength are being widely implemented. The DOD Class 3 and Class 4 PKI implementations are a strategic option for positive authentication for access to information systems. While it is recognized that older ESM applications may not be PKI-capable, all new acquisitions and upgrades should incorporate PKI technologies that are compatible with the DOD PKI or other technologies implemented through an NSA-certified product.

- *(EGB.0020: CAT III) If an ESM application provides internal I&A services, the IAO will ensure that I&A is accomplished using a DOD PKI Class 3 or 4 certificate and hardware security token (when available), or an NSA-certified product.*

A group authenticator is any mechanism for authentication, shared by members of a group. A common example of the use of a group authenticator is referred to as a group ID; it allows access to systems or data by all the members of the group who have access to (i.e., knowledge of) a single password. Using a group authenticator definitely provides ease of use. However, if used alone, it also eliminates the ability to attribute a specific action to a unique individual and therefore significantly reduces the value of audit data.

Some group authenticators are inherently less secure because it is more difficult to control them. In the example of a shared password, one authorized user can easily disclose the password to additional, unauthorized users and thus reduce or eliminate the security value of the password. Authenticators such as hardware tokens are more secure because they are not, if correctly implemented, easily shared or duplicated.

To diminish the problems with group authenticators, while still taking advantage of their benefits, some mitigating controls are needed. By requiring that an individual authenticator also be used as part of processes that use a group authenticator, it can become possible to track actions back to an individual user. By ensuring that group authenticators are based on an implementation that has been reviewed and determined to be sufficiently robust, it is possible to have an adequate level of trust in their security.

- *(EGB.0030: CAT II) If an ESM application provides internal I&A services, the IAO will ensure that ESM group authenticators are used only in conjunction with an individual authenticator.*

- *(EGB.0040: CAT II) If an ESM application provides internal I&A services, the IAO will maintain documentation verifying that any definition of group authenticators not based on the DOD PKI has been explicitly approved by the DAA.*

3.3.2 Authenticator Strength and Protection

Authenticators are the means for proof of identity. If an authenticator is compromised, access control over the resources available to the associated user account is lost. Therefore authenticators must be strong enough to resist circumvention and must be protected from unauthorized disclosure.

Passwords are the most common form of authenticator. The chief means to strengthen passwords is by controlling their composition, expiration interval, and change options. Requiring that passwords are composed of multiple character types and that they are changed regularly helps to deter attacks based on simple guessing, dictionary, or brute force techniques.

- *(EGB.0050: CAT II) If an ESM application provides internal I&A services and passwords are used, the IAO will ensure that the ESM application is configured to the extent system capabilities permit to enforce the following password composition, automatic expiration, and reuse requirements:*

    - *Passwords are at least eight characters long.*
    - *Passwords are composed of a case sensitive mix including at least one upper case letter, lower case letter, number, and special character.*
    - *Passwords are not the same as the associated ID.*
    - *At least four characters are changed when a new password is created.*
    - *Passwords cannot be changed more than once in any 24-hour period without the intervention of the IAO.*
    - *Passwords expire automatically at least every 90 days.*
    - *The last 10 passwords are not reused.*

The following additional guidance should be implemented where resources require more robust protection:

- Passwords are 12 to 16 characters long.
- Repeating characters are not used.
- Passwords do not include dictionary words, names, dates, or phone numbers.
- Passwords for privileged users expire automatically at least every 30 days.

The strength of an authenticator becomes largely irrelevant if unauthorized users or processes are able to extract it from a file or intercept it as it traverses a network. For that reason it is essential to protect authentication data both at rest and in transit. In additional to any physical means of protection such as partitioning, encryption services are needed to adequately protect authentication data.

- *(EGB.0060: CAT I) If an ESM application provides internal I&A services, the IAO will ensure that the ESM password repository is encrypted.*

- *(EGB.0070: CAT II) The IAO will ensure that the transmission of authentication data for access to ESM applications is encrypted.*

In addition to the mechanisms enforced by the ESM application, there are some procedural requirements that enhance the protection of authenticators. The first is meant to ensure that authenticators stay exclusive to the rightful owner.

- *(EGB.0080: CAT II) The IAO and ESM administrators will ensure that authenticators (e.g., passwords) for ESM accounts are not shared, are not embedded in access scripts, and are not stored on function keys.*

The other procedural requirement is related to software implementation. For simplicity, some vendors provide installation procedures that include or specify the use of particular IDs and passwords. These are referred to as the factory set, default, or standard values. While there are some cases where deviating from the specified values renders an application inoperable or enormously increases the installation complexity, most vendors allow the password or both the ID and password to be chosen by the installer. While the security vulnerability of using default values may seem obvious, this is a common security problem. If not eliminated for an ESM application, this vulnerability might allow anyone with knowledge of the vendor documentation to have privileged access to a system. When performed at the appropriate time in the implementation cycle, eliminating this vulnerability is usually easy and the benefit is significant.

- *(EGB.0090: CAT III) The IAO will ensure that all factory set, default, or standard user IDs and passwords for the ESM application are removed or changed.*

3.3.3 Key Management

Various cryptographic operations use keys as input to their algorithms. Since knowledge of some key data can effectively negate the security value of encryption, hash, and digital signing operations, the importance of secure key management is clear.

The use of keys in ESM applications can include symmetric, asymmetric, or both key types. Symmetric keys, also called secret keys, have to be shared among the entities using them. Due to this characteristic, symmetric key lengths must be more carefully selected and all the key data securely managed. Asymmetric keys involve a private and public key pair and are the basis for Public Key cryptography. Although the public key portion of the key pair can be disclosed, the corresponding private key data must be securely managed.

A Key Management Infrastructure (KMI) is used to manage both symmetric and asymmetric key types. DODI 8500.2 defines KMI services. “The KMI provides a common unified process for the secure creation, distribution, and management of cryptographic products, such as asymmetric keys (e.g., PKI) and traditional symmetric keys (e.g., Electronic Key Management System (EKMS)) that enable security services for DOD information systems.”

Secure key management for ESM applications is achieved by conforming to the policies and procedures implemented through the NSA-managed DOD KMI. This currently includes symmetric key management provided by EKMS and asymmetric key management provided by the DOD PKI.

- *(EGB.0100: CAT III) If an ESM application utilizes symmetric keys, the IAO will ensure that those keys are produced, controlled, and distributed using NSA-approved key management technology and processes.*

- *(EGB.0110: CAT III) If an ESM application utilizes asymmetric keys, the IAO will ensure that those keys are produced, controlled, and distributed using DOD PKI Class 3 or Class 4 certificates and hardware security tokens that protect the user’s private key.*

- *(EGB.0120: CAT I) If an ESM application utilizes symmetric or asymmetric keys on a system processing classified information, the IAO will ensure that those keys are produced, controlled, and distributed using NSA-approved key management technology and processes.*

3.4 Enclave and Computing Environment

This section describes ESM security requirements based on applicable IA controls in the Enclave and Computing Environment subject area. These requirements address five general areas: data protection, user account management, application customization, auditing, and network access.

3.4.1 Data Protection

ESM applications may capture or create a variety of data including current availability and event data; historical availability, performance, and accounting data; and hardware and software configuration data. This data is a significant asset for several reasons:

- Current availability and event data represent the operational status of one or more information systems. The information may be used as the basis for reallocating resources or for corrective action.

- Historical data provides usage information for short and long term resource planning, cost allocation, and forensic investigations.

- Hardware and software configuration data are used for asset inventory and deployment activities.

It was noted earlier that ESM application data needs to be protected at the same confidentiality level as the systems on which that data is based. This reflects the fact that unauthorized access to this data might have several negative effects:

- Invalid data, used as the basis for configuration changes or corrective action, could result in a loss of availability.

- Specific configuration data might expose a subject system to specific attacks based on known vulnerabilities.

- The loss of logging data for ESM actions could make it impossible to identify the source of malicious activity.

The following requirements ensure that access to ESM data is appropriately restricted and that access is recorded for subsequent review if needed.

- *(EGC.0010: CAT II) The IAO will ensure that access to data created by ESM applications is limited so that only authorized ESM application processes and users can read or update it.*

- *(EGC.0020: CAT II) ESM administrators will ensure that ESM actions that access or change data are logged and that the logs are reviewed periodically or immediately upon system security events.*

It should be noted that the requirement for data access and change logging can be satisfied using OS facilities as well as ESM application functions.

ESM data can itself be sensitive or classified if it includes specific hardware or software configuration data for systems processing sensitive or classified data. As this data traverses networks, it must be protected from disclosure according to its confidentiality level and in accordance with need-to-know requirements. In this case protection is mandated in the form of data encryption.

- *(EGC.0030: CAT II) ESM administrators will ensure that ESM data that includes unclassified, sensitive information (including system hardware or software configuration data) that traverses a commercial or wireless network is encrypted, at a minimum, using NIST-certified cryptography.*

- *(EGC.0040: CAT I) ESM administrators will ensure that ESM data that includes classified systems' hardware or software configuration data that traverses a network cleared to a lower level than the ESM data is encrypted using NSA-approved cryptography.*

3.4.2 User Account Management

User accounts defined for access to ESM applications grant privileges that allow various interactions with the systems running the applications. These interactions can be relatively benign, such as access to availability information, or quite powerful, such as reconfiguration or even restart of multiple systems. Span of control is also a factor. Some ESM account privileges are intended to allow an end user to perform maintenance only on their own system. While this can be cost effective, it must be defined carefully so that privileges over other systems are not granted unintentionally. It is important that privileged accounts are used only when required in order to closely document and control the use of privileges.

Access controls over ESM programs and data are directly related to privileges granted through ESM account definitions. The controls provide little protection if the practices used for account management are weak. In large organizations, careful account management can be a significant administrative burden. One mechanism for dealing with that burden is the use of role based access controls in applications that support it. This helps to control privileges according to a user's functional need for them and should simplify account maintenance.

The following requirements enforce the principles of least privilege and separation of duties for ESM accounts. This means that privileges are not granted unless necessary and not used unless intended. The requirement to use role based access control, where feasible, helps to reduce the complexity and potential errors associated with privileged account maintenance.

- *(EGC.0050: CAT II) The IAO will ensure that ESM administrator accounts are assigned the minimum privileges required for the user's job function.*

- *(EGC.0060: CAT III) The IAO will ensure that ESM administrator accounts are not used for non-privileged functions.*

- *(EGC.0070: CAT III) The IAO will ensure that ESM administrator accounts are established and administered in accordance with a role-based access scheme to the maximum extent technically feasible within the ESM software.*

Errors or omissions in account management may be the result of process problems. Processes that are documented and implemented are helpful in avoiding problems that could weaken security over ESM accounts. Formal documentation of privilege assignment is an essential part of the management process.

- *(EGC.0080: CAT II) The IAM will ensure that a process is documented and implemented for the management of ESM accounts.*

- *(EGC.0090: CAT II) The IAM will maintain documentation of the assignment of ESM accounts and roles.*

The account management process must incorporate manual or automated steps to handle situations that would otherwise allow continued privileged access after it is no longer authorized. The following requirements provide for disabling and deleting accounts that are no longer needed.

- *(EGC.0100: CAT II) The IAO will ensure that the account management process applicable to ESM accounts includes manual or automated procedures to enforce the following for inactive, suspended, and terminated accounts:*

    - *Accounts for which unauthorized activity is identified are disabled immediately.*
    - *Accounts to be terminated due to user re-assignment or departure are disabled or deleted within two days of notification by the user or user's supervisor.*
    - *Accounts inactive for more than 35 days are disabled.*
    - *Accounts disabled for more than 180 days are deleted.*

3.4.3 Application Customization

It is sometimes necessary to customize COTS or GOTS applications to get functions to work properly or efficiently in a specific environment. Customizations may be in the form of changes to existing programs or the addition of new programs.

Because ESM applications often run as privileged processes on many systems, invalid or malicious changes to those applications have the potential to cause significant problems over an extended area. The installation and propagation of improperly modified ESM programs could cause serious compromises of confidentiality, integrity, and availability to a single site or an entire enclave.

Part of the defense against such compromises is the use of a formal configuration management process. Such a process includes review and approval of change requests and controls to allow only authorized personnel to implement changes. In this way, ESM application changes are adequately reviewed before implementation and the changes are not implemented without the explicit knowledge and consent of the appropriate parties.

- *(EGC.0110: CAT III) If an ESM application is altered by the addition of locally written programs or changes to COTS or GOTS programs, and if those added or changed programs are executed during ESM operation by a privileged process or user, the IAO will ensure that a documented configuration management (CM) process exists for the implementation of those added or changed programs.*

3.4.4 Auditing

Auditing for information systems involves the collection and retention of data so that it is possible to assess the adequacy of system controls and the degree of compliance with policies and procedures. Audit data provides information needed to evaluate the source, scope, and impact of a security incident.

As with other areas of concern, auditing for ESM applications has additional significance because the applications often execute as privileged processes. Certain user actions, performed within a limited time period and in certain patterns, can be signs of preparation or attempts to exploit system vulnerabilities that involve privileged access. These actions include attempts to access security files and attempts to access an interactive application. Certain actions taken by an application, in response to a perceived threat, are also potential signs of an attack. These actions include denying access due to successive invalid password entries; disabling IDs, network ports, or other access mechanisms; or otherwise flagging actions that appear to be malicious.

Taken individually, these events are not absolute indicators and any response to them could be premature. However, if the execution of the actions is not recorded, it becomes impossible to recognize later the pattern that confirms the occurrence of an attack. Therefore it is necessary to capture this information as the events occur.

- *(EGC.0120: CAT II) To the extent technically feasible from and applicable to the ESM application, the IAO will ensure that audit data containing user ID, date and time of event, type of event, and success or failure of event are written for the following:*

  - *Successful and unsuccessful attempts to access security (e.g., account or permission) files*
  - *Successful and unsuccessful logon to (attempt to access) the application*

- *Denial of access resulting from excessive number of logon attempts*
- *Blocking or blacklisting a user ID, terminal, or access port, and the reason*
- *Activities that might modify, bypass, or negate safeguards controlled by the application.*

As mentioned, it may take a collection of data over time to recognize an attack. Because the time span of an attack can be lengthy, some period of data retention has to be selected. Also, because investigations can take extended periods of time, it is necessary to be sure that important evidence is not inadvertently lost.

- *(EGC.0130: CAT II) The IAO will ensure that audit data generated by an ESM application is retained for at least one year.*

Some ESM applications offer built-in facilities for the collection and reporting of audit data, or even for generating warnings based on the data. While such features may not appear to be immediately important from an operational view, waiting until malicious activity is suspected to deploy the features can mean that the data is lost or is not processed in time to detect an attack in progress. An ESM application might also have the ability to take defensive action such as disabling user access. Deploying these product capabilities early ensures that they will be available when needed.

- *(EGC.0140: CAT II) ESM administrators will deploy tools that provide audit data review and report capabilities.*

- *(EGC.0150: CAT II) To the extent technically feasible from and applicable to ESM applications, ESM administrators will ensure that an automated, continuous on-line monitoring and audit trail creation capability is deployed with the capability to immediately alert personnel of any unusual or inappropriate activity with potential IA implications and with a user configurable capability to automatically disable the application if serious IA violations are detected.*

A primary benefit of collecting audit data is lost if the data is only reviewed when a security incident has been confirmed. A proactive approach to reviewing the data on a regular basis is important for early detection and for prevention of attacks.

- *(EGC.0160: CAT II) The IAO will ensure that audit data generated by ESM applications are reviewed at least weekly for indications of inappropriate or unusual activity and that suspected IA policy violations are analyzed and reported.*

Protecting audit data requires safe physical storage and appropriate access control. To accomplish this, the data is backed up and access control definitions are implemented to limit updates to the original and backup copies. The destruction of audit data due to media failures or the absence of update access controls represents a double loss. The ability to recognize and possibly recover ESM data that was tampered with is lost and the system resources expended to collect, process, and store the data are effectively lost.

- *(EGC.0170: CAT II) The IAO will ensure that ESM audit data is backed up not less than weekly onto a different system or media than the system on which the ESM application executes.*

- *(EGC.0180: CAT II) The IAO will ensure that access to ESM audit data, including backup copies, is limited so that only authorized ESM application processes, ESM administrators, or SAs can read, update, or delete it.*

3.4.5 Network Access

It was noted earlier that the use of networks is inherent to ESM applications. The requirements in this section are designed to protect three aspects of network access for ESM applications:

- ESM administrators and other ESM users may access ESM servers under the control of I&A services provided by an ESM application. As noted in *Section 3.1, Introduction*, these services establish and verify the identity of the user or process.

- ESM servers are necessarily part of a network that connects many hosts. The presence of a network connection on any application host warrants protective measures.

- In one form or another, virtually all ESM data transits a network. The nature of this data and the possible impact of corruption or interception indicate a need for supplementary integrity controls.

I&A services provided by ESM applications must include protective controls equivalent to other providers (such as operating systems) of those services. These controls address repeated logon attempts, concurrent sessions, and notification of privacy and security conditions. If the controls are not implemented, the system is more vulnerable to password attacks, continuing use of compromised accounts, and the loss of the right to use audit data.

- *(EGC.0190: CAT II) If an ESM application provides internal I&A services, ESM administrators will ensure that successive logon attempts are controlled using one or more of the following:*

  - *Access is denied after multiple unsuccessful logon attempts.*
  - *The number of access attempts in a given period is limited.*
  - *A time delay control system is employed.*

- *(EGC.0200: CAT II) If an ESM application provides internal I&A services and it supports multiple logon sessions for each user ID, ESM administrators will ensure that a session maximum is defined.*

The requirement for a warning banner is essential because it preserves the Government’s right to monitor, record, and audit session activities. If this requirement is not fulfilled, data that is collected may have to be excluded from any prosecution of an intruder.

- *(EGC.0210: CAT III) The IAO will ensure that access to an ESM application includes the presentation of a warning banner from the ESM host. The banner will consist of statements that advise the user of the following elements:*

  - *The system is a DOD system.*
  - *The system is subject to monitoring, recording, and auditing.*
  - *Monitoring is authorized in accordance with applicable laws and regulations and conducted for purposes of systems management and protection, protection against improper or unauthorized use or access, and verification of applicable security features or procedures.*
  - *Use of the system constitutes consent to monitoring.*
  - *The system is for authorized U.S. Government use only.*

It should be noted that the requirement for a warning banner may be satisfied using OS facilities on the ESM host as well as ESM application functions.

Although technically possible, the use of ESM applications on systems operating at different classification levels is not permitted. The use of ESM applications on systems connected by non-DOD networks or on a mix of DOD and non-DOD systems is also not permitted. Those configurations would require a controlled interface and it has not been determined that such a configuration for ESM applications could be adequately secured. The primary issue with those configurations would be the lack of ability to segregate the ESM data to the level of assurance required.

- *(EGC.0220: CAT II) The IAO and ESM administrators will ensure that ESM applications are not implemented across DOD information systems operating at different classification levels or across DOD and non-DOD systems or networks.*

The sensitive nature of ESM applications and data necessitate increased protection for the ESM application hosts. One tool to accomplish this is a host-based intrusion detection system (HIDS). A discussion of HIDS functions can be found in the *Enclave Security STIG*. The requirements in this document for encryption of ESM data traversing a network are a primary driver to the need for a HIDS on ESM servers.

- *(EGC.0230: CAT II) The IAO will ensure that ESM application hosts are protected by a host-based intrusion detection system (HIDS).*

There are a number of problems that can be introduced when applications depend on the transfer of data across networks. Some of these are:

  - Data traversing any network may be subject to corruption from hardware problems.
  - Deliberate tampering during transmission might corrupt data.
  - If a host’s network identity is spoofed, counterfeit data might be introduced into the transmission and corrupt data on the receiving host.

Integrity checking and host authentication mechanisms help to combat data corruption. Integrity checking mechanisms can operate at the file level and at the message level. Host authentication can occur at the session or message level.

Some ESM applications that transmit data incorporate hash checking or similar integrity checks at the file level. An entire file on the sending host is passed through a hash algorithm and the hash value is sent with the file. On the receiving host, the file is passed through the same algorithm and the receiver's hash value is compared with the sender's. If the values do not match, an error is recorded and the file is retransmitted. Hash checking or similar integrity mechanisms that are applied to entire files are embedded features of the ESM application.

Integrity checking at the message level and host authentication are frequently related to network sessions. One common implementation is the use of the Secure Sockets Layer (SSL) or a follow-on protocol. Session protocols like SSL offer hash checking as messages are exchanged to validate the contents of each message. Host authentication occurs at session startup and periodically thereafter. By requiring both the server and client to authenticate each other, the data source is established with higher assurance than simple network address validation. The use of session protocols requires some integration by the ESM vendor, but offers a standards-based solution that can be implemented using elements of the DOD PKI.

Another common way for ESM applications to implement data integrity and host authentication would be the use of components that adhere to the Simple Network Management Protocol version 3 (SNMPv3) User-based Security Model (USM). This standard is documented in RFC 3414. The SNMPv3 USM specification requires the implementation of an authentication protocol that is responsible for data integrity and data origin authentication. The *Network Infrastructure Security Technical Implementation Guide* requires network management tools using SNMP to implement the SNMPv3 Security Model.

ESM applications may provide varying options for ensuring data integrity over a network. The requirements here are intended to enforce the implementation of those options in the ESM products that have these capabilities.

- *(EGC.0240: CAT II) ESM administrators will ensure that ESM applications are configured to use integrity mechanisms such as parity checks, cyclic redundancy checks, or hash checks to ensure the integrity of transmitted data.*

- *(EGC.0250: CAT II) ESM administrators will ensure that ESM applications are configured to assure session integrity and to detect or prevent session hijacking through the use of mutual (both server and client) authentication mechanisms such as those available through Secure Sockets Layer (SSL) and successor protocols.*

3.5 Enclave Boundary Defense

This section describes ESM security requirements based on applicable IA controls in the Enclave Boundary Defense subject area. These requirements address remote access to ESM applications. For the purposes of this document, remote access is described as any access to an ESM application from a host outside of the enclave in which the ESM application host resides.

All remote access to DOD information systems, including privileged and unprivileged, requires a restricted access path that includes encryption and strong authentication. The following requirements address those security measures for any remote ESM application access.

- *(EGD.0010: CAT II) The IAO will ensure that remote access to ESM applications is secured through the following:*

    - *Use of a managed access control point such as a remote access server in a DMZ*
    - *Session encryption using, according to the data classification, NIST-certified or NSA-approved cryptography*
    - *Strong user authentication that resists spoofing, such as a two-factor system.*

ESM applications enable privileged functions and provide access to data that is likely to be sensitive, and may be classified. Therefore, in practically all cases, ESM administrators are identified as privileged users. A discussion of privileged user remote access can be found in the *Enclave Security STIG*. Because of the potential for serious impact from the compromise of privileged access, additional security for sessions and special attention to auditing to those sessions is necessary.

The following requirements address the need to protect remote ESM administrator access and to review use of that access because it represents remote user privileged access.

- *(EGD.0020: CAT II) The IAO will ensure that remote access for ESM administrators uses:*

    - *Session security measures such as a VPN configured in blocking mode to discard all but authorized traffic*
    - *A process that creates an audit log for each remote session.*

- *(EGD.0030: CAT II) The IAM/IAO will review the audit log for every remote session of an ESM administrator.*

While virtual private network (VPN) implementations do provide desirable session protection, they can also be used to conceal malicious traffic. A network-based intrusion detection system (IDS) is a tool to address this risk. A discussion of network IDS functions can be found in the *Enclave Security STIG*. The requirement in this document enforces the specific need for remote ESM administrative traffic that uses a VPN to be examined for intrusive behavior.

- *(EGD.0040: CAT II) The IAO will ensure that VPN traffic for remote ESM administrator sessions is visible to a network intrusion detection system (IDS).*

3.6 Physical and Environmental

This section describes ESM security requirements based on applicable IA controls in the Physical and Environmental subject area. These requirements address a specific need for the physical security of ESM application hosts.

Some ESM applications perform functions that monitor or reconfigure the operational status of other information systems. It has also been noted that some ESM applications perform privileged functions. If the availability or integrity of the ESM application is negatively impacted, many or all of the other information systems may also be negatively impacted. Tampering with the physical configuration or environment of the host system is a simple way to cause availability and integrity issues for ESM applications.

Though all application hosts require physical access controls, the operational role, privileged functions, and potential far-reaching impact of problems with ESM applications establish a greater need to provide appropriate physical security for the ESM host servers.

- *(EGE.0010: CAT II) The IAO will ensure that physical access to ESM application hosts is restricted to specifically authorized personnel.*

It is expected that the physical access controls for ESM hosts will be of a more robust nature than those that might be used for less critical resources such as local file and print servers. This could be implemented by more restricted physical isolation or other measures. The specific nature of this robustness is at the discretion of the responsible IAO.

3.7 Continuity

This section describes ESM security requirements based on applicable IA controls in the Continuity subject area. These requirements address the need for steps that will enable recovery of ESM application functions when some event causes the primary processing resources to be damaged or lost.

The need for a recovery capability is driven by the fact that some ESM applications provide information assurance functions (such as I&A services and security patch management) or monitoring and reconfiguration functions that are required to maintain operational status. Their loss would degrade or seriously reduce overall capability to meet mission objectives.

The following requirements are related to the need for specific attention to ESM applications in disaster recovery planning. The objective is to see that ESM applications that provide a business or mission essential function to an environment are not overlooked.

- *(EGF.0010: CAT II) The IAM will ensure that the disaster recovery plan for the enclave includes appropriate provision for the continuity of ESM applications that provide I&A services, other essential information assurance functions such as security patch management, or other services required to monitor or maintain operational status.*

  - *For ESM applications serving MAC III systems, resumption within five days of activation.*
  - *For ESM applications serving MAC II systems, resumption within 24 hours of activation.*
  - *For ESM applications serving MAC I systems, transfer to an alternate site for the duration of an event with little or no loss of operational continuity.*

- *(EGF.0020: CAT II) The IAM will ensure that ESM applications that provide I&A services, other essential information assurance functions such as security patch management, or other services required to monitor or maintain operational status are identified for priority restoration planning.*

Note that separate plans for ESM applications are not required. On the contrary, the best implementation would be an integrated plan for the enclave that included ESM and other essential applications.

In addition to facilities and hardware required for recovery, it is necessary to have copies of the data and software used by ESM applications. The following requirements specify what is necessary, while recognizing that different recovery capabilities are needed for ESM services for systems with different availability requirements.

- *(EGF.0030: CAT II) The IAO will ensure that ESM application data is managed appropriately to the MAC of the systems served by the ESM application.*

  - *For ESM applications serving MAC III systems, ESM data is backed up at least weekly.*
  - *For ESM applications serving MAC II systems, ESM data is backed up daily and the recovery media is stored at an off-site location that affords protection in accordance with the mission assurance category and confidentiality level of the data.*
  - *For ESM applications serving MAC I systems, ESM data is backed up by maintenance of a redundant secondary system, not collocated, that can be activated without loss of data or disruption to the operation.*

- *(EGF.0040: CAT II) The IAO will ensure that backup copies of software for ESM applications that provide I&A services, other essential information assurance functions such as security patch management, or other services required to monitor or maintain operational status are stored in a fire-rated container or otherwise not collocated with the operational software.*

3.8 Vulnerability and Incident Management

This section describes the ESM security requirements based on applicable IA controls in the Vulnerability and Incident Management subject area. These requirements address the need to adopt and implement a process to mitigate hardware and software vulnerabilities as they are identified.

It appears that no vendor's products are immune from the identification and exploit of vulnerabilities. Many vulnerabilities have been linked to deficient programming and, although those issues have received a lot of attention, serious problems are still discovered. The proliferation of vulnerability and exploit information on the Internet exacerbates these problems by making the information widely and easily accessible. Unfortunately, mitigating action is often not taken, even when a fix has been identified and made available. These facts indicate the necessity for a process to make sure that efficient and rapid mitigating action is taken.

An effective vulnerability management process includes the elements needed to identify and take mitigating action for vulnerabilities. Automated assessment tools can speed identification. An ESM configuration management application can enable rapid deployment of mitigating actions. When these or equivalent manual procedures are part of an enclave's documented work processes, successful vulnerability management is much more likely.

To make certain that vulnerabilities are addressed, a formal commitment to security patch implementation is essential. It is not necessary to have a unique policy for ESM resources, only to have a policy that covers ESM resources. Manual or automated documentation indicating that patches have been applied provides auditable evidence that mitigating action has been taken.

- *(EGG.0010: CAT II) The IAM will ensure that a vulnerability management process that encompasses ESM applications and server hardware is documented and implemented.*

- *(EGG.0020: CAT II) The IAO will ensure that all security related patches to ESM applications are applied and that completion is documented for each applicable asset.*

This page is intentionally left blank.

APPENDIX A. RELATED PUBLICATIONS

Government Publications:

Chairman of the Joint Chiefs of Staff (CJCS) Manual 6510.01, “Defense-in-Depth: Information Assurance (IA) and Computer Network Defense (CND),” 25 March 2003

Committee on National Security Systems (CNSS) Instruction No. 4009, “National Information Assurance (IA) Glossary,” May 2003

Committee on National Security Systems (CNSS), “National Security Telecommunications and Information Systems Security Policy (NSTISSP) No. 11,” July 2003

Department of Defense Directive 8500.1, “Information Assurance (IA),” 24 October 2002

Department of Defense Instruction 5200.40, “DOD Information Technology Security Certification and Accreditation Process (DITSCAP),” 30 December 1997

Department of Defense Instruction 8500.2, “Information Assurance (IA) Implementation,” 6 February 2003

Department of Defense Instruction 8520.2, "Public Key Infrastructure (PKI) and Public Key (PK) Enabling", 1 April 2004

Department of Defense Instruction 8551.1, “Ports, Protocols, and Services Management (PPSM),” 13 August 2004

Department of Defense Memorandum, “Department of Defense (DOD) Information Assurance Vulnerability Alert (IAVA),” 30 December 1999

Department of Defense Memorandum, “Open Source Software (OSS) in the Department of Defense (DOD)”, 28 May 2003

Department of Defense Memorandum, “Policy Guidance for Use of Mobile Code Technologies in Department of Defense (DOD) Information Systems,” 7 November 2000

Defense Information Systems Agency (DISA), “Database Security Technical Implementation Guide,” Version 6, Release 1, 7 July 2003

Defense Information Systems Agency (DISA), “Enclave Security” Security Technical Implementation Guide,” Version 2 Release 1, 1 July 2004

Defense Information Systems Agency (DISA), “Network Infrastructure Security Technical Implementation Guide,” Version 5, Release 2, 29 September 2003

Defense Information Systems Agency (DISA), “OS/390 Security Technical Implementation Guide,” Version 4, Release 1, 4 August 2003

Defense Information Systems Agency (DISA), “UNIX Security Technical Implementation Guide,” Version 4, Release 4, 15 September 2003

Defense Information Systems Agency (DISA), “Web Server Security Technical Implementation Guide,” Version 4, Release 1, 29 August 2003

Defense Information Systems Agency (DISA), “Windows NT/2000/XP Addendum,” Version 4, Release 1, 26 February 2004

National Security Agency, “Information Assurance Technical Framework (IATF),” Release 3.1, September 2002

Vendor Publications:

International Business Machines, “Tivoli Business Systems Manager Administrator’s Guide,” Version 2.1.1, with Fix Packs 1-10 GC32-0799-01

International Business Machines, “Tivoli Business Systems Manager Getting Started,” Version 2.1.1, with Fix Packs 1-10, GC32-0801-01

International Business Machines, “Tivoli Business Systems Manager Installation and Configuration Guide,” Version 2.1.1, Re-released with Fix Packs 1–10 GC32-0800-02

International Business Machines, “Tivoli Business Systems Manager Release Notes,” Version 2.1.1, SC23-4841-01

International Business Machines, “Tivoli Enterprise Console Adapters Guide,” Version 3.9, SC32-1242-00

International Business Machines, “Tivoli Enterprise Console Command and Task Reference,” Version 3.9, SC32-1232-00

International Business Machines, “Tivoli Enterprise Console Installation Guide,” Version 3.9, SC32-1233-00

International Business Machines, “Tivoli Enterprise Console Release Notes,” Version 3.9, SC32-1238-00

International Business Machines, “Tivoli Enterprise Console Rule Developer’s Guide,” Version 3.9, SC32-1234-00

International Business Machines, “Tivoli Enterprise Console Rule Set Reference,” SC32-1282-00

International Business Machines, “Tivoli Enterprise Console User’s Guide,” Version 3.9, SC32-1235-00

International Business Machines, “Tivoli Monitoring for Business Integration Installation and Setup Guide,” Version 5.1.1 SC32-1402-00

International Business Machines, “Tivoli NetView for UNIX Release Notes,” Version 7.1.4, SC32-1239-00

International Business Machines, “Tivoli NetView for Windows Release Notes,” Version 7.1.4, SC32-1240-00

Microsoft Corporation, “Microsoft Systems Management Server 2003 Concepts, Planning, and Deployment Guide”

Microsoft Corporation, “Microsoft Systems Management Server 2003 Operations Guide”

Microsoft Corporation, “Scenarios and Procedures for Microsoft Systems Management Server 2003: Security”

Tivoli Systems, “Application Development Environment Release Notes,” Version 3.6.5, 8 January 2001

Tivoli Systems, “Application Development with TME 10 ADE,” Version 3.6, September 1998

Tivoli Systems, “Tivoli Enterprise Firewall Security Toolbox User’s Guide,” Version 1.3.1, GC23-4826-01

Tivoli Systems, “Tivoli Enterprise Installation Guide,” Version 4.1.1, GC32-0804-01

Tivoli Systems, “Tivoli Event Integration Facility Reference,” Version 3.9, SC32-1241-00

Tivoli Systems, “Tivoli Inventory Release Notes Version 3.6.2,” December 1999

Tivoli Systems, “Tivoli Management Framework, Version 4.1.1. Documentation Road Map,” November 2003, GI11-0891-01

Tivoli Systems, “Tivoli Management Framework Maintenance and Troubleshooting Guide,” Version 4.1.1, GC32-0807-01

Tivoli Systems, “Tivoli Management Framework Planning for Deployment Guide,” Version 4.1.1, GC32-0803-01

Tivoli Systems, “Tivoli Management Framework Release Notes,” Version 4.1.1, GI11-0890-01

Tivoli Systems, “Tivoli Management Framework User’s Guide,” Version 4.1.1, GC32-0805-01

Tivoli Systems, “Tivoli Manager for MQSeries Revised February 14, 2003, Release Notes,” Version 2.4.0, GI10-3059-06

Tivoli Systems, “Tivoli NetView for UNIX Administrator’s Guide, Version 7 1, SC321246-00

Tivoli Systems, “Tivoli NetView for Windows User’s Guide,” Version 7, Release 1.4, SC32-1245-00

Tivoli Systems, “TME 10 ADE Application Services Manual, Volume I,” Version 3.6, September 1998

Tivoli Systems, “TME 10 ADE Application Services Manual Volume II,” Version 3.6, September 1998

Tivoli Systems, “TME 10 AEF User’s Guide,” Version 3.6, September 1998

Tivoli Systems, “TME 10 AEF Release Notes,” Version 3.6, September 1998

Tivoli Systems, “TME 10 Inventory User’s Guide Version 3.6,” September 1998

Tivoli Systems, “TME 10 Software Distribution AutoPack User’s Guide Version 3.6,” September 1998

Tivoli Systems, “TME 10 Software Distribution Release Notes Version 3.6,” September 1998

Tivoli Systems, “TME 10 Software Distribution User’s Guide Version 3.6,” September 1998

Other Publications:

International Telecommunication Union (ITU), “CCITT Recommendation X.700 (09/92), Management Framework for Open Systems Interconnection (OSI) for CCITT Applications”

International Telecommunication Union (ITU), “ITU-T Recommendation M.3400 (02/2000), TMN Management Functions”

Web Sites:

Carnegie Mellon Software Engineering Institute's Software Technology Roadmap	http://www.sei.cmu.edu/str/
Distributed Management Task Force (DMTF)	http://www.dmtf.org/
DOD Ports and Protocols Program	http://www.cert.mil/portsandprotocols/
IBM Tivoli Documentation	http://publib.boulder.ibm.com/tividd/td/tdprodlist.html
Information Assurance Support Environment	http://iase.disa.mil/
Information Assurance Technical Framework (IATF) Forum	http://www.iatf.net/
Internet Engineering Task Force (IETF)	http://www.ietf.org/
Microsoft Security Bulletin Search	http://www.microsoft.com/technet/security/current.aspx
Microsoft SMS Home	http://www.microsoft.com/smserver/
Microsoft SMS 2003 Toolkit 1	http://www.microsoft.com/smserver/downloads/2003/tools/toolkit.asp
Microsoft TechNet: Systems Management Server	http://www.microsoft.com/technet/prodtechnol/sms/default.mspx
National Institute of Standards and Technology (NIST) Computer Security Resource Center (CSRC) (Add Microsoft Security Bulletin Search web site)	http://csrc.nist.gov/pcig/cig.html

This page is intentionally left blank.

APPENDIX B. TIVOLI

B.1. Tivoli

B.1.1 Introduction

The evolution of open systems has led to the development of large enterprises. Tivoli considers enterprises to be large collections of heterogeneous hardware and software connected by different sized networks that support the distributed computing requirements of an organization. Because of the use of diverse hardware and software different individuals perform many procedures and responsibilities within enterprises differently.

This has caused the need for a solution to enterprise management. Tivoli has developed a suite of integrated products specifically designed to address many of the management, administrative, and security requirements of Enterprises.

B.1.2 Tivoli Enterprise Architecture

Tivoli's approach to enterprise architecture is to break the enterprise into smaller sections referred to as Tivoli Management Regions (TMRs). Each region is a self-contained entity. In early releases of Tivoli the TMRs were two tier constructs having a management server and a series of clients, now referred to as endpoints. Because the TMR Server was limited to the number of clients/endpoints that could be connected and serviced, gateway servers, which work as proxy servers were added. Gateway servers absorbed some of the responsibilities of the management server and enabled more endpoints to be connected. In addition, Tivoli Management Agents (TMAs) were added to receive distributions, execute tasks, run monitors, and send events. TMAs are used to manage many of the Tivoli systems as opposed to Managed Nodes or PCs running the PC agent (PC managed nodes).

By standardizing much of the management, administrative, and security procedures, TMRs can be interconnected across networks thus creating the Tivoli enterprise architecture.

- *(TME.0001: CAT II) The IAO will ensure that a current configuration document exists describing the Tivoli enterprise architecture, to include such items as TMR server(s), gateways, managed nodes, firewalls, and endpoints.*

B.1.3 Tivoli Management Region (TMR)

As mentioned earlier, TMRs are the most basic element in a Tivoli enterprise. A TMR consists of a Tivoli Management Server(s), one or more Tivoli Gateways, and one or more Tivoli endpoints. TMRs can exist independently in conjunction with other TMRs or interconnected either wholly or partially to other TMRs.

Because of the possible configurations that can exist in a Tivoli Enterprise regarding TMRs, functionality and support can be centralized or decentralized. Centralized support uses a top down structure where everything is initiated at the TMR server level and flows downward through the gateways to the endpoints. Decentralized support is different in that much more of the support is initiated at the local gateway layer and distributed accordingly. In either case, since a standardized methodology is used and a common interface provided implementation, communication and problem resolution can be easier.

B.1.4 Tivoli Servers

In a Tivoli Management Enterprise there can be various types of Tivoli servers. They can be a TMR server, Managed Node(s), Tivoli Gateway(s), and Tivoli Endpoint Gateway(s). In all cases, Tivoli servers run the Tivoli Management Framework software and have the libraries, binaries, data files, and the graphical user interface (GUI) (the Tivoli Desktop) needed to support their portion of the Tivoli Environment. A typical Tivoli server contains the following components:

- An object database is used to maintain object data for either the entire Tivoli region or a specific portion of the Tivoli region.

- The object dispatcher, oserv, is used to coordinate communications with managed nodes and gateways.

- The endpoint manager is used to manage all of the endpoints in the TMR.

B.1.4.1 TMR Server

In order for a TMR to exist, a TMR server must exist. There can be only one TMR server in a TMR and it must be created first. A TMR server is the central controlling element in a Tivoli Management Region. TMR servers are placed on either UNIX or Windows platforms. The TMR server is responsible for maintaining the TMR server’s object database. It contains information about all of the resource components of a TMR to include security and functionality of the resources. TMR servers are capable of creating the other Tivoli servers in a TMR.

TMR Servers communicate with other TMR servers, gateways or managed nodes, by using the oserv daemon, which is an object dispatcher. The oserv daemon normally initiates execution when the TMR server starts up and terminates when the server comes down. In order to manage the endpoints, TMR servers use an endpoint manager service, which also runs on the TMR server.

Communication with a TMR Server by an administrator is performed through a graphical user interface GUI, known as the Tivoli Desktop. The TMR server is responsible for performing all authentication and verification necessary to ensure the security of TMR data. All major TMR administration, security, and management functions are generally performed at the TMR server level.

- *(EGE.0010: CAT II) The IAO will ensure that physical access to ESM servers is restricted to authorized ESM administrators and SAs.*

B.1.4.1.1 General Security Considerations for TMR Servers

As mentioned earlier, the TMR Server is the central controlling element in a TMR. As a result the security of the TMR Server is critical to the TMR. TMR servers can exist on different types of mid tier platforms. The following general areas apply to all TMR Servers and must be addressed when securing a TMR Server:

- Restrict the Tivoli libraries, binaries, X11 resource files unauthorized access and update.
- Restrict the Command reference pages (manual pages) from unauthorized access and update.
- Restrict the Message catalogs from unauthorized access and update.
- Restrict the object database from unauthorized access and updates in accordance with the *Database STIG*.
- Restrict the Environment setup files from unauthorized access and update.
- Limit all outside remote access to the TMR server.
- Limit port utilization to approved ports only.
- Restrict Tivoli commands from unauthorized access and update and usage.
- Limit the number of Tivoli System Administrator roles to a minimum of three authorized personnel.
- Set encryption level to DES.
- Restrict Tivoli administrators to read access to the Tivoli directories.
- Restrict the TMR server and managed nodes to a group of static IP addresses if using DHCP.

- *(TME.0008: CAT I) The IAO will maintain access-based documentation on approved personnel security investigations/security clearances, need-to-know, and written authorization in accordance with DODI 8500.2.*

- *(TME.0009: CAT I) The IAO will maintain written procedures for handling the introduction of classified information into the Enterprise/TMR.*

- *(TME.0010: CAT I) The IAO will maintain a written contingency processing plan for the Tivoli Enterprise or TMR.*

- *(TME.0011: CAT III) The IAO will have documentation on file demonstrating the performance of annual contingency testing on the Tivoli Enterprise or TMR.*

- *(TME.0012: CAT II) The IAO will ensure that all IAVA patches are applied.*

- *(TME.0013: CAT II) The IAO will ensure that encryption is implemented in the Tivoli Enterprise in accordance with DODI 8500.2.*

- *(TME.0014: CAT II) The IAO will ensure that unauthorized / undocumented software does not exist on the TMR server.*

- *(TME.0015: CAT I) The IAO will ensure that unsupported software is removed or upgraded prior to a vendor dropping support.*

- *(TME.0016: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading software prior to the date the vendor drops security patch support.*

B.1.5 General Security Considerations for Tivoli Servers

As mentioned earlier, there are various types of Tivoli servers, their functionality determines the type of server they are. The following general security considerations apply to all Tivoli servers:

- Access to Tivoli Server functions is controlled through the assignment of Tivoli authorization roles to Tivoli users. Without proper assignment of these roles, the integrity of Tivoli software could be compromised.

- All Tivoli Servers require Tivoli software to be installed. As with most software products, many are composed of program and data files for which proper access controls are essential. Although most of the product files will reside in directories subject to access controls required for the Tivoli Management Framework software, there are some Tivoli product directories and files that will require specific access controls.

- If web interface features are used, user transactions are processed through HTML pages. The integrity of the HTML files must be assured to maintain transaction security.

- Since various DBMSs are used to support the Tivoli products on the Tivoli Servers, their integrity and availability must be assured. The *Database STIG* must be considered when applying restrictions.

- Unauthorized remote access to the Tivoli Servers can have a major impact on the integrity of the TME. As a result, careful consideration as to firewalls, ports, and physical locations of the Tivoli Servers must comply with the *Enclave* and *Network STIGs*.

  - Since most Tivoli products require role-based access, all assignments to Tivoli Servers should be limited minimally based on job function and responsibilities.

  - Encryption must be implemented to assure confidentiality and data integrity within the TME.

  - Endpoint access to a TMR is better controlled by assigning Tivoli servers that are using DHCP Dynamic Host Configuration Protocol DHCP to a group of static IP addresses.

- *(TME.0002: CAT I) The IAO will maintain written procedures for handling the introduction of classified information into the Enterprise/TMR.*

- *(TME.0003: CAT I) The IAO will maintain a written contingency processing plan for the Tivoli Enterprise or TMR.*

- *(TME.0004: CAT III) The IAO will have documentation on file demonstrating the performance of annual contingency testing on the Tivoli Enterprise or TMR.*

- *(TME.0005: CAT II) The IAO will ensure that encryption is implemented in the Tivoli Enterprise in accordance with DODI 8500.2.*

- *(TME.0006: CAT II) The IAO will ensure that unauthorized/undocumented software does not exist on the TMR server.*

- *(TME.0007: CAT I) The IAO will ensure that unsupported software is removed or upgraded prior to a vendor dropping support.*

- *(TME.0008: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading software prior to the date the vendor drops security patch support.*

- *(TME.0009: CAT I) The IAO will ensure that platform SRRs will be completed on each endpoint within the Tivoli Enterprise (tmersrvd).*

- *(TME.0010: CAT II) The IAO will ensure that documentation exists justifying the platforms that use the tmersrvd account.*

B.1.6 Tivoli Managed Nodes

A Tivoli Managed node is a server or a PC that runs the Tivoli Management Framework software. Tivoli Managed nodes also run the oserv daemon and maintain a local database. In addition, Tivoli Managed nodes allow for the use of the Tivoli Desktop facility.

The primary difference between a TMR server and a managed node is that the TMR server database contains all information about the entire TMR and a managed node's database contains information about the managed node and its subordinates. In order for a Tivoli Managed node to exist, it must run the Tivoli Management Framework software. There can be multiple Manage Nodes in a TMR, all of which are subordinate to the TMR server.

B.1.6.1 General Security Considerations for Tivoli Managed Nodes

A Tivoli Managed Node runs the Tivoli Management Framework software. It is capable of performing many of the same functions as the TMR. Some examples of a Tivoli Managed Node are TMR Servers from other TMRs, gateways, or endpoint gateways. The following general areas apply to most Tivoli Managed Nodes:

- All Tivoli endpoints require Tivoli software to be installed. As with most software products, many are composed of program and data files for which proper access controls are essential.

- The restriction of administrator commands assures that unauthorized personnel will not be able to perform unauthorized functions.

- Port restriction is critical in assuring that endpoints access the correct TMR.

- Encryption of endpoint transmission will ensure confidentiality and integrity.

- Because the tmersrvd account performs system activities, it will need system authority on endpoints.

- Allow the tmersrvd accounts to utilize non-expiring passwords will eliminate the possibility of endpoints losing functionality and accessibility.

- Restrict the Tivoli libraries, binaries, X11 resource files unauthorized access and update.

- Restrict the Command reference pages (manual pages) from unauthorized access and update.

- Restrict the Message catalogs from unauthorized access and update.

- Restrict the database from unauthorized access and updates in accordance with the *Database STIG*.

- Restrict the Environment setup files from unauthorized access and update.

- Limit all outside remote access to the Tivoli Manage Node if it is a TMR server.

- Limit port utilization to approved ports.

- Restrict Tivoli commands from unauthorized access and update.

- Limit the number of Tivoli System Administrator roles to a minimum.

- Apply Tivoli security guidelines at the policy region level.

- Set encryption level to DES.

- Restrict managed nodes to a group of static IP addresses if using DHCP.

- *(TME.0017: CAT II) The IAO will ensure that the TMR Managed Nodes are located in a controlled area accessible to authorized personnel only in accordance with DODI 8500.2.*

B.1.6.2 Gateways

A gateway is a managed node that resides between the TMR server and endpoints within a TMR. Gateways are proxy servers and are responsible for controlling communication between the TMR server and endpoints. Gateways perform operations on the endpoints on behalf of the TMR server. TMR programs can be launched by Gateways to run on endpoints and the results reported to the TMR server by the gateways. Depending on the Tivoli products running on the gateways and their processing capabilities, gateways can manage large numbers of endpoints, thus relieving the TMR server to perform other services on behalf of the TMR. Finally, Gateways provide an additional benefit in that they can be organized to support specific subsections of a TMR thus allowing for local management, administration, and security of sections of a TMR.

B.1.6.2.1 General Security Considerations for TMR Gateways

A Tivoli Gateway runs the Tivoli Management Framework software. The difference between a Tivoli Gateway and a TMR Server is the amount of control. Gateways control the endpoints subordinate to them. TMR Servers control the entire TMR. As a result, the following general areas apply to most Tivoli Gateways:

- Restrict the Tivoli libraries, binaries, X11 resource files unauthorized access and update.

- Restrict the Command reference pages (manual pages) from unauthorized access and update.

- Restrict the Message catalogs from unauthorized access and update.

- Restrict the database from unauthorized access and updates in accordance with the *Database STIG*.

- Restrict the Environment setup files from unauthorized access and update.
- Limit port utilization to approved ports.

- Restrict Tivoli commands from unauthorized access and update and usage to authorized personnel.

- Limit the number of Tivoli System Administrator roles to a minimum.

- Create Tivoli administrators for Tivoli policy regions.
- Restrict TMR gateways to a group of static IP addresses if using DHCP.

- Set encryption level to DES.

- *(TME0018: CAT II) The IAO will ensure that the TMR Gateways will be located in a controlled area accessible to authorized personnel only in accordance with DODI 8500.2.*

B.1.7 Endpoints

In a Tivoli environment Endpoints make up the lowest level of a TMR. They are systems that ultimately receive Tivoli operations. Endpoints are capable of receiving distributions, executing tasks, running monitors, and sending events. Tivoli Endpoints can be Windows, UNIX, OS/2, and NetWare Systems.

B.1.7.1 General Security Considerations for Endpoints

Because Tivoli Management Agent endpoints do not require the software, much of the security is based on the platform. The following generals areas apply when security Tivoli Endpoints:

- Restrict Tivoli endpoint libraries, binaries, and resource files unauthorized access and update.

- Restrict administrator commands from unauthorized access and update and use.

- Limit port utilization to approved ports.

- Set encryption level to DES.

- Allow the tmrservd account to utilize system authority on endpoints.

- Allow the tmrservd account to utilize non-expiring passwords.

- *(TME.0019: CAT I) The IAO will ensure that platform SRRs will be completed on each endpoint within the Tivoli Enterprise (tmersrvd).*

- *(TME.0020: CAT II) The IAO will ensure that documentation exists justifying the platforms that use the tmersrvd account.*

B.2. Tivoli Management Framework

B.2.1 Overview

The Tivoli Management Framework (TMF) is the backbone of the Tivoli Enterprise Management Software. It was designed to provide administration and management services for the Tivoli products. The TMF provides such capabilities and services as an administrator facility, a system policy facility, a notification facility, a graphical user interface (GUI), and a command line interface (CLI). In addition, the TMF provides programs, commands, policies and resources that are used by all of the Tivoli products.

In order to manage and administer specific areas and resources of an Enterprise, other Tivoli products are integrated on top of the TMF software. The other Tivoli products provide their own specialized capabilities and services, which complement those provided by the TMF. The TMF software does not affect the configuration of the platform on which it is installed, whereas other Tivoli products that are installed on top of the TMF software may.

A Tivoli Management Region (TMR) is a collection of resources that are controlled by a TMR Server. The TMF software is installed on the TMR server and on each managed node in the TMR. The only difference between the TMF software installed on the TMR server and the TMF software installed on the Gateways and managed nodes is the resources controlled. The TMR server controls resources globally across the TMR. Gateways and managed nodes control locally assigned resources within Policy Regions. Because the TMF software plays such a critical role in the management and administration of the TMR and providing the backbone support for the other Tivoli products, it must be highly protected.

- *(TMF.0001: CAT II) The IAO will ensure that TMF software is restricted from unauthorized update and access in accordance with the platform STIGs, this STIG, and DODI 8500.2.*

- *(TMF.0002: CAT II) The IAM will ensure that TMR server is located in an area, which is restricted from unauthorized access.*

- *(TMF.0003: CAT II) The IAM will ensure that TMR gateways are located in areas, which are restricted from unauthorized access.*

- *(TMF.0004: CAT II) The IAM will ensure that TMR managed nodes are located in areas, which are restricted from unauthorized access.*

B.2.2 General Considerations

B.2.2.1 Tivoli Interfaces

The TMF when it is installed provides two types of interfaces. They are the Tivoli Desktop and the Command Line Interface CLI. The following sections describe each.

B.2.2.1.1 Tivoli Desktop

The Tivoli Desktop is the standard gui interface application for communicating with the TMF and the other Tivoli products. The Tivoli Desktop enables administrators to communicate with objects by manipulating dialogs. The Desktop consists of a set of services and components. They are:

- DSL language and compiler are used to specify dialog layout and the method of interaction
- Desktop, which is the display manager and command/callback engine
- Desktop services libraries that contain the run-time functions
- Gadget library which is the object-oriented abstraction for manipulating dialogs

The Tivoli Desktop enables administrators to communicate with Tivoli application products by using either method invocation or drag-and-drop. Method invocation occurs when dialog controls, known as gadgets, are executed. Gadgets display information and perform operations for administrators.

Drag-and-drop enables administrators to copy or move icons from one dialog to another. Icons represent a collection of resources that are modeled as objects. A collection is a group of common objects. When expanded, icons are replaced by dialogs that enable administrators to communicate with the associated resource. Resource dialogs allow administrators to change system configurations and create new resources in the distributed environment. When an icon is moved from one dialog to another, it is added to the collection represented on the second dialog and removed from the collection represented on the first dialog.

Because the Desktop provides a standardized format, the types of activities an administrator may perform and the resources displayed on the Desktop are controlled by the resource role(s) assigned to the administrator. As additional Tivoli products are added on to the TMF, new features, icons, resources, resource roles and functionality may be added to the Tivoli Desktops. This again depends on the resource role(s) assigned the administrators. In order for an administrator to access the Tivoli Desktop application, the administrator must have a user role assigned. The user role will only allow the administrator to display the resources that exist in the administrator’s collection. In order to perform any management function, the administrator will be required to be assigned additional roles. When the Desktop is started, by default, the GUI displays the contents of the collection the administrator is able to either view or manage.

Depending on the platform the TMF software is installed the Desktop will be either installed automatically, such as under UNIX, or separately as under Windows.

- *(TMF.0005: CAT II) The IAO will ensure that all Tivoli Desktops are located in an area, which is restricted from unauthorized access and behind the local firewall.*

- *(TMF.0006: CAT II) The TMR Administrator will restrict access to the Tivoli Desktop by Policy Region for the Policy Region Administrator(s).*

B.2.2.1.2 Command Line Interface (CLI)

The command line interface CLI is an alternate way of performing many Tivoli functions in a TMR. The CLI enables an administrator to use Tivoli commands instead of navigating through the different panels of the Tivoli Desktop. The command line interface also provides administrators the ability to develop scripts that can be executed to perform management or administrative tasks in a single step. The Tivoli commands can also be used in shell scripts and with system utilities such as the UNIX cron utility.

- *(TMF.0007: CAT II) The IAM will ensure that the SA will restrict CLI access to the TMR Administrator and the Policy Region Administrator(s).*

B.2.2.2 Tivoli Management Framework Components

The TMF software is broken down into a group of components that not only support the functionality of the TMF software but also provide support for the other Tivoli products. The following subsections describe some of the components and services of the TMF software.

B.2.2.2.1 Administrators

Tivoli Administrators are responsible for the management, administration and operational support for the Tivoli Management Environment TME. In order for Tivoli administrators to perform these functions, the Tivoli administrators must have the authority necessary under Tivoli. In order to satisfy this requirement, roles have been created by Tivoli to categorize the job responsibilities of administrators. As a result, depending on the types of tasks an administrator must perform, the administrator is assigned one or more roles under Tivoli, which permits the administrator to perform those functions. The TMF when it is installed provides basic roles for Tivoli administration. As additional Tivoli products are implemented on top of the TMF additional roles and functions may be added. The following subsections describe the roles and functions provided by the TMF software.

- *(TMF.0008: CAT II) The IAM will ensure that Tivoli Administrators are provided with minimal authorization roles necessary to perform their functions.*

B.2.2.2.1.1 Authorization Roles

As mentioned above, authorization roles provide administrator with the privileges necessary to manage Tivoli resources. They are mutually exclusive and not hierarchical. Each role has its own set of associated privileges and as a result an administrator may be given multiple roles. In addition an administrator may be given different roles for different resources within a TMR. Authorization roles can be granted at the global TMR level or at a resource level within a local Policy Region. When authorization roles are granted at the global TMR level, they apply to all resources across the entire TMR. When authorization roles are granted at the resource level they apply to all resources in a Policy Region. Tivoli authorization roles have predefined names and specific functions assigned.

The following table describes the predefined default roles that exist under the Tivoli Management Framework:

Role	Description
super	An administrator with the super role is able to connect and disconnect TMRs; perform maintenance operations on the TMR; install clients, products, and patches. The super role should not be given to many administrators.
senior	An administrator with the senior role is able to create and define all Tivoli resources. It is required for configuration and policy tasks such as creating an administrator, setting policy, or expiring notices.
admin	An administrator with the admin role is able to perform day-to-day operational tasks and system configurations. It is required for general system management tasks such as adding a user item to a profile, distributing file packages, or changing the message of the day.
user	An administrator with the user role has read-only browsing capability. This role is required for bringing up a Tivoli Desktop. In addition, the user role allows limited operations that do not affect configuration information.
backup	An administrator with the backup role in a TMR is able to create backups of Tivoli databases. The administrator must be assigned the backup role in the TMR that contains the TMR server and clients being backed up.
restore	An administrator with the restore role in a TMR is able to restore Tivoli databases. The administrator must be assigned the restore role in the TMR that contains the TMR server and clients being restored.
install-product	An administrator with the install-product role is able to install new applications and products in the local TMR.
install-client	An administrator with the install-client role is able to install new managed nodes within policy regions that allow the ManagedNode resource type.
policy	An administrator with both the policy and senior role can create policy regions. In addition, the administrator is able to add managed resources to policy regions and set up the policies that govern the policy region.
Dist_control	An administrator with the Dist_control role is able to control multiplexed distribution 2 (MDist 2) distributions, such as canceling, pausing, and resuming a distribution. Also, the administrator is able to delete distributions from the database.

NOTE: Administrators with the super or senior role can not only create administrators, but can also change an administrator's roles. Administrators with the admin or user role can assign only the user role. Administrators with only one or more of the install-client, install-product, backup, or restore roles are not able to assign any roles. In order to ensure that senior System Administrators can perform operations at their authorization level and below, assign them all authorization roles below their current level.

- *(TMF.0009: CAT II) The TMR Administrator will have the authorization roles necessary to create and distribute all policies.*

- *(TMF.0010: CAT II) The TMR Administrator will have the roles necessary to create all Policy Regions.*

- *(TMF.0011: CAT II) The TMR Administrator will have the roles necessary for the creation of all sub Policy Regions.*

Initially, when the TMF software is installed on the TMR server, a root administrator is automatically created by Tivoli. (On UNIX operating systems, a root administrator has root authority and on Windows operating systems a root administrator has Administrator privileges.) The Tivoli root administrator is assigned with the Tivoli authorization roles of "super" and "senior" by default. Because the administrator has these authorization roles the administrator is able to access Tivoli and create other administrators as needed. When the root administrator creates a new administrator, the new administrator has no authorization roles assigned. The new administrator must be assigned authorization role(s) either at the global TMR level or at the local Policy Region resource level.

As other Tivoli application products are installed onto the TMF software, additional roles may be added by the Tivoli application. These new roles apply to specific functions that will need to be performed by the other Tivoli product. As resources or roles are added to the Tivoli environment, an administrator's roles may be modified in order to perform management operations against new resources. Resources controlled by an administrator are displayed as icons on the administrator's Tivoli Desktop.

It should be noted that in a TMR, a platform System Administrator may or may not be a Tivoli Administrator. Platform System Administrators (SAs) are normally responsible for the software on a platform and the permissions or group authorizations associated with the platform files. On UNIX operating systems, SAs have root authority and on Windows operating systems have Administrator privileges. As a result, in a Tivoli environment there may be a SA assigned to manage the operating system software on the TMR server, or on a gateway or a managed node.

- *(TMF.0012: CAT I) The IAM will ensure that the Tivoli root administrator is limited to the SA and the TMR Administrator.*

B.2.2.2.1.2 Authorization Roles across TMR Boundaries

When an administrator is assigned a TMR role, the administrator is able to perform operations that may affect resources anywhere in the local TMR. If an administrator is assigned any TMR role other than super, the role maps across the boundaries of all two-way connected TMRs. Administrators with the super role can perform tasks that require the super role only within the TMR in which the administrator was created. This restriction stops an administrator with the "super" role from creating administrators in other TMRs.

- *(TMF.0013: CAT II) The IAO will ensure that Authorization Roles across TMR Boundaries are limited to the TMR Administrator(s).*

B.2.2.2.2 Policy and Policy Regions

B.2.2.2.2.1 Policy

A policy is a set of rules that that is used by the TMF to manage resources. Policies can be used to customize Tivoli products in a Tivoli environment. A policy has two parts: a default policy and a validation policy. Default policies are used to describe the default properties assigned to a resource when it is created. Validation policies define the valid values for attributes that are checked when a new instance is created or modified. Validation policies run when a dialog is set and closed. Tivoli will check what values an administrator is attempting to assign against those that the administrator is permitted to assign and are also used to ensure that all resources in a policy region comply with region's established policy.

The following table contains the task library default policy methods and their purpose.

Method	Description
tl_def_dist_mode	Provides the default mode for distributing task binaries throughout a Tivoli management region
tl_def_man_nodes	Provides a default list of managed nodes for a task library
tl_def_prof_mgrs	Provides a default list of profile managers for a task library
tl_def_set_gid	Provides the default group ID assigned to a task
tl_def_set_uid	Provides the default user ID assigned to a task

The following table contains the task library validation policy methods and their purpose.

Method	Description
tl_val_man_nodes	Validates the list of target managed nodes on which a task or job will execute
tl_val_prof_mgrs	Validates the list of target profile managers associated with a task or job
tl_val_set_gid	Validates the effective group ID assigned to a task or job
tl_val_set_uid	Validates the effective user ID assigned to a task or job

- *(TMF.0014: CAT II) The TMR Administrator will be responsible for the creation and distribution and implementation of all policies.*

B.2.2.2.2.2 Policy Regions

A policy region is a collection or container of resources that share one or more common policies. Policy regions contain managed resources. The set of managed resources in a policy region depends on the applications and the products installed in the TMR. The Tivoli managed resources can be managed nodes, task libraries, and profile managers. Initially, managed resource belongs to only one policy region, but as changes occur in an organization, they can be moved to other regions.

Policy regions are similar to directories in a file system in that they enable an administrator to construct a model of the organization through which system management tasks and operations are performed. They can also be arbitrarily nested and can contain any desired set of managed resources. Different administrators can have different roles in different regions, thus giving them the ability to manage appropriate sets of resources. It should be noted that although endpoints are not created by default in a policy region, they can be added to a policy region and can be moved between policy regions the same way as other managed resources.

- *(TMF.0015: CAT II) The TMR Administrator will be responsible for the creation of all Policy Regions.*

B.2.2.2.2.3 Policy Subregions

A policy subregion is a policy region that is part of another policy region. Initially, when a policy subregion is created, it uses the resource and policy properties of the parent policy region. At a later time they can be changed to reflect specific requirements of the subregion.

- *(TMF.0016: CAT II) The TMR Administrator will be responsible for the creation of all sub Policy Regions.*

B.2.2.2.2.4 Managed Resources

Managed resources represent system and or network resources that are subject to certain sets of rules within Tivoli. They provide a model of the physical resources to be managed by Tivoli applications. In order to create a managed resource an administrator must use the Tivoli Application Development Environment. Managed resources are created in a policy region. Managed resources must have a default policy defined in order to be created. Tivoli resources can only be created in or moved into a policy region in which the type is a managed resource. It should be noted that even thought a managed resource type can be valid in several policy regions, each specific instance of the managed resource type belongs to only one policy region. A managed resource is a member of the policy region in which it is currently resides, regardless of where the resource was created.

Resource types are listed in a policy region as either a directly or indirectly managed resources. Directly managed resources are created and managed within the policy region. Indirectly managed resources are created and managed within a profile manager. Tivoli resources can only be created in or moved into a policy region in which the type is specified as a managed resource. In order to control the values or attributes of a resource, the policy objects associate must be changed.

It should be noted that Tivoli draws a distinction between removing and deleting collections and objects. When an object is removed from the Desktop, the object is removed from that Desktop collection. When an object is deleted from the Desktop, that object is deleted.

- *(TMF.0017: CAT II) The IAO will ensure that the Tivoli TMR Administrator and Policy Region Administrator(s) maintains documentation on all authorized managed resources in the Tivoli Management Region.*

- *(TMF.0018: CAT III) The IAM will ensure that only the TMR Administrator has the roles necessary to assign access to Notice Groups in the TMR.*

- *(TMF.0019: CAT II) The IAM will ensure that the TMR Administrator has the roles necessary to create and delete managed resources in the TMR.*

B.2.2.2.2.5 Interconnected TMR Resource Exchange

TMRs can be interconnected in different ways. The first example is referred to as a one-way connection. This is generally used for hierarchical purposes. In a one-way connection the managed TMR is aware of the managing TMR. In addition, information is passed only from the managed TMR to the managing TMR. The reason for such resource exchanges is that a Tivoli application product may need access to all the known resources in all connected TMRs. This information is obtained solely from information contained within the TNR of the local TMR.

The second example of interconnected TMRs is referred to as a two-way connection. In a two-way connection, both TMRs are totally aware of each other. When two TMRs are connected to each other in a two-way connection, the TNR in each region can exchange information with the other about registered resources.

- *(TMF.0019: CAT III) The IAM will ensure that the TMR Administrator has implemented Tivoli one-way connections.*

- *(TMF.0020: CAT III) The TMR Administrator will be responsible for the assignment of TMR resources that may be exchanged between TMRs.*

B.2.2.2.3 Schedule

In a TME, jobs are run to perform various types of administration and management. In order to control job execution the TMF provides a scheduler component. The scheduler is responsible for performing the following functions:

- Schedule jobs to be executed at specific times.
- Schedule a job to be executed repeatedly or a pre-specified number of times.
- Schedule a job to be executed at pre-specified time intervals.

The scheduler is not responsible for defining jobs but allows previously created jobs to be executed. Prior to the scheduler submitting a job for execution, the role required to run the job is compared to information about the administrator who initially scheduled the job. This ensures that the administrator who initially scheduled the job still exists and that the scheduling administrator’s authorization has not been lowered. The scheduler notifies the administrator of the job’s status. If the administrator no longer exists or no longer has the authorization necessary to run the job, the job execution fails when it is run. It should be noted that the administrator who schedules the job, is referred to as the job owner until the job is deleted from the scheduler. Which means, that each time the job is run, it runs under the identity of the administrator who owns the job.

In order for an administrator to view the jobs scheduled for execution, the administrator must have a user role. In order for an administrator to edit, disable, enable, or remove a job the administrator must have at least the admin role and in some cases the senior role.

- *(TMF.0021: CAT III) The TMR Administrator will control the use and execution of the Scheduler.*

B.2.2.2.4 Notification

The TMF provides a notification facility that is used to track system administration activity. It informs Tivoli Administrators of system management operations and reports which administrator performed a particular operation. The notification facility is helpful because it can be used as an audit trail. The following are the notification mechanisms used by Tivoli.

B.2.2.2.4.1 Notices

A notice is a message that contains information about an operation or change in the Tivoli environment. Notices are generated by Tivoli resource operations. Only one copy of a notice is generated for a particular system management operation. Message text is stored in Message catalogs, thus allowing administrators to view messages in different languages, based on the Tivoli Desktop.

- *(TMF.0022: CAT III) The IAO will ensure that all notices are backed up on a regularly scheduled basis.*

B.2.2.2.4.2 Notice Groups

The TMF provides the following default set of notice groups:

TME Scheduler	Contains notices related to the operation of the scheduler.
TME Administration	Contains notices related to general Tivoli functions, such as the installation of applications and managed nodes, the addition of new administrators, and the creation and removal of policy regions and resources.
TME Authorization	Contains notices related to authorization errors, changes in administrator roles, and the creation of new administrators with authorization roles.
TME Diagnostics	Contains notices generated from running the diagnostic command wchkdb or other maintenance operations.

NOTE: In a TME each TMR has the same set of default notification groups. In order for an individual to obtain notices from a remote TMR the user must subscribe to the Notice Group. Whenever two or more TMRs are connected, the local TMR is not automatically updated with the available notice groups in the remote regions. A user can subscribe to a notice group in a remote region without subscribing to the same notice group in the local TMR. The deletion of a user’s subscription in a local notice group does not impact the subscription on any remote TMR’s notice groups.

- *(TMF.0023: CAT III) The IAM will ensure that only the TMR Administrator assigns access to Notice Groups in the TMR.*

B.2.2.2.5 Task Library

Task libraries are used to store tasks and job. Multiple task libraries can exist in a TMR and organized by policy regions. This enables tasks and jobs to be restricted to a specific group of resources. Task libraries can also store executable files that are used by Tivoli applications. These executable files can only be invoked by the Tivoli application. In this case, the task library provides the application with a known location for binaries, scripts, or programs that the application uses. The task library also provides information about options required to run each task it contains.

A task library does not have to be exchanged in a resource exchange before its tasks can be performed across region boundaries. A task can be run on any endpoint or profile manager subscriber in any connected region.

Because tasks and jobs are subject to policy like other managed resources, enforcement of policy can be performed by the policy region that contains the task library. The task library policy options contain default and validation policy for controlling such aspects as the allowable user and group IDs, the set of available task endpoints and profile managers on which a set of tasks may be run, and the task distribution mode.

- *(TMF.0024: CAT II) The IAM will ensure that access to the Tivoli Task libraries is restricted to the TMR Administrator(s) and Policy Region Administrator(s).*

- *(TMF.0025: CAT III) The IAM will ensure that only the TMR Administrator has the roles necessary to assigns access to Notice Groups in the TMR.*

B.2.2.2.5.1 Tasks

A task defines an operation to be performed on a routine basis. It identifies the executable files to be run, the role required to execute the task, and the user ID or group ID under which the task will run. Tasks do not contain such information as where a task will execute, the output type, or execution parameters. This information is specified when the task is executed. In order for a task to be created, the executables must exist on the platform that the task is to be executed. Tasks are controlled by policy restrictions and are enforced by the policy region that contains the task library. Only an administrator with the required role in the policy region of the task endpoint can run the task.

- *(TMF.0026: CAT II) The TMR Administrator will ensure that all tasks are restricted by Policy Region.*

B.2.2.2.5.2 Jobs

A job is defined by an application or component of the Tivoli Management Framework. Jobs are tasks that are executed on specific managed resources in a TMR. When a job is created, the associated task and execution information are defined. The execution information specified identifies a list of managed resources where the on which the task will run and output will be displayed. Jobs can run serially, parallel or in staged groups. It is the responsibility of the scheduler to control the execution of jobs. When the execution mode is serial or staged, the task is executed on the managed resources in alphabetical order.

- *(TMF.0027: CAT II) The TMR Administrator will ensure that all jobs are restricted to Policy Region.*

B.2.2.2.6 RDBMS Interface Module (RIM)

The RDBMS Interface Module (RIM) is responsible for providing a common interface between Tivoli application products and relational databases. RIM enables Tivoli application products to store and retrieve information in a database-independent manner. RIM also enables applications developed within the Tivoli Application Development Environment (ADE). In addition to Tivoli applications, the MDist 2 service also uses RIM to hold distribution data.

RIM is installed as part of the TMF installation, and then each application that uses it creates the appropriate RIM objects. The application installation generally consists of: creating the user ID and user tables in the RDBMS server and creating the RIM component for the application to connect to the RDBMS server. The managed node where the RIM object is created is called RIM host. It has two requirements. They must be a local manage node in the TMR and must be preconfigured with the TDBMS client or server software.

RIM objects are created on managed nodes in a TMR. RIM configuration options are specified during the installation. This information is used to create the RIM object and register it in the Tivoli object database. The password for each RIM object must be the same as the password of the RDBMS database that it accesses. In other words, the RIM password and the RDBMS database password need to be the same. Therefore, the password of each RIM object must be changed to match that of its repository. When the RDBMS database password changed, each RIM object's password must be changed. Not only must the password be the same but the user name for the RIM object and the user name for the RDBMS must also match.

Applications that use RIM will gather and store data in the RDBMS, without interaction by the user. Several components work together to make communication through the RIM possible. The client application uses RIM APIs to make a request to gather and retrieve data. The RDBMS_Interface translation layer receives the request from the client, looks up the RIM host, and sends the request to the RIM host. The third component is the vendor adaptor layer, which sends vendor-specific requests to the database.

Each RIM-supported application that is installed will require the administrator to apply configuration changes. The roles needed to perform this function are described in the later sections of this document. The types of databases supported are: Oracle, DB2, Informix, Sybase and Microsoft SQL Server.

- *(TMF.0028: CAT II) The DBA will ensure that access to a RIM database is restricted in accordance with the Database STIG.*

- *(TMF.0029: CAT II) The SA will ensure that the RIM APIs are protected from unauthorized update and access.*

- *(TMF.0030: CAT II) The TMR Administrator will be responsible for controlling Tivoli access and update to RIM databases.*

B.2.2.3 Tivoli Management Framework Services

In addition to the above components of the TMF, the following services are provided to support both the TMF software and the other Tivoli Products.

B.2.2.3.1 Object Dispatcher oserv

The Object Dispatcher oserv daemon is responsible for coordinating communication between systems within the Tivoli environment. The oserv daemon is an implementation of a CORBA compliant Object Request Broker (ORB). All Managed Nodes run the oserv daemon. The oserv daemon may or may not be started when the TMR server or the gateways are started. It should be noted that when the object dispatcher runs on a Windows Managed Node it starts as the Windows built-in system account because it is a service. As the object dispatcher starts, it attempts to validate the Tivoli remote access account, tmersrvd. Oserv uses a log file to record information about activity. This log file is called oservlog.

- *(TMF.0031: CAT II) The Object Dispatcher oserv will be restricted from unauthorized update, access and execution.*

- *(TMF.0032: CAT III) The IAM will ensure that the starting and stopping of the oserv daemon is restricted to the TMR Administrator or SA.*

B.2.2.3.2 TMF Management Database

The TMF uses a logical management database to store information about the objects in a TMR. The database is distributed between the TMR server and all of the managed nodes that are configured in a TMR. When the management database is installed on a Windows NT system the database file must be installed on the local NT File System (NTFS).

- *(TMF.0033: CAT II) The SA will restrict the TMR database in accordance with platform and database STIG security guidelines.*

B.2.2.3.3 Application Services

Application services are the primary Tivoli capabilities and services that are used by the other Tivoli products. These services include task libraries (for remote or local command execution), schedulers, a notification mechanism, and file distribution capabilities.

In order to support file distributions, the TMF supports Multiplex Distribution (MDist). It provides a fan-out mechanism that enables a more efficient method of file distributions to many systems. Repeaters are used to perform the fan-out mechanism. They receive a copy of a distribution and pass the files onto one or more targets systems, which saves bandwidth.

Finally, in order for the other Tivoli Application product services to provide support they must utilize profiles and profile managers.

- *(TMF.0034: CAT II) The IAO will ensure that TMR Administrator and Policy Region Administrator(s) are the only ones authorized to use MDist.*

B.2.2.3.4 Installation Services

The TMF is able to install other TMF components and Tivoli application products. These services use the TMF’s ability to transfer files and execute commands on client systems. Installation services are a core part of the TMF. They can be used directly or through the Tivoli Software Installation Services (SIS), which provides a Java-based front end to them.

- *(TMF.0035: CAT II) The IAO will ensure that use of the SIS will be restricted to the TMR Administrator and Policy Region Administrators.*

B.2.2.3.5 Tivoli Management Agent Support

A Tivoli Management Agent (TMA) is a managed system that runs the TMA endpoint software. The TMA software works in conjunction with the other products to manage the endpoint resources. Because it is designed to provide full management of systems that do not have the oserv daemon and the Tivoli distributed database loaded, the TMA software is able to use less resources.

In order to track TMA endpoints, the TMR server uses the endpoint manager, which is a service (or daemon) on the TMR. The endpoint manager is responsible for controlling and configuring gateways and endpoints, assigning endpoints to gateways, and maintaining the endpoint list, which contains information about each endpoint in a TMR.

- *(TMF.0036: CAT II) The TMR Administrator will be responsible for the creation of a TMA.*

- *(TMF.0037: CAT II) The SA will be responsible for installing the TMA endpoint software.*

B.2.2.3.6 Tivoli Web Interface

The TMF provides access to Web-enabled Tivoli Enterprise applications through a browser. When an HTTP request is sent to a TMR server, the request is redirected to the Web server that processes the HTTP request. The TMF supports Web access by using a collection of servlets and support files that are installed on the Web server. The servlets establish a secure connection between the Web server and the Tivoli server. Some Tivoli Applications products use this service to present a Web-based interface for administrators and end-users. The TMF uses this service to enable administrators to manage TMA endpoints. Any Web server that supports the Servlet 2.2 specification can be used in conjunction with the Tivoli Web Interface.

The TMF allows an administrator to view endpoint status and configuration through an HTTP browser. Each endpoint has a unique user ID and password. The user ID and password are required when changing the endpoint configuration. No special authorization is needed to view this existing information. When an endpoint is installed, a default user ID and password are created. (The default user ID is tivoli and a default password is boss.) When the endpoint successfully logs in to its assigned gateway, a new password is randomly generated.

- *(TMF.0038: CAT II) The Web Administrator will restrict Tivoli Web access in accordance with the Web STIG.*

- *(TMF.0039: CAT II) The TMR Administrator will be responsible for restricting Web access to Tivoli.*

- *(TMF.0040: CAT II) The TMR Administrator will ensure that the default user ID and password will be changed in accordance with the Web STIG, the UNIX STIG, the Windows STIG, and DODI 8500.2.*

B.2.2.3.7 Name Registry

The Tivoli Name Registry (TNR) is responsible for registering object names and for providing a quick lookup table of object labels and object IDs. When a TMR is initially installed, the TNR is created. The TNR serves as the overseer of names for an entire TMR. It is the primary focal point for exchanging information about known resources when multiple management regions are connected.

A registered name is the name by which a resource instance is registered on the TNR when the entry is created. Every resource has a name and is classified by some particular type. The name registry does not allow two resources of the same type to have the same name within a single TMR. It is possible for resource names to be duplicated within two or more connected TMRs. If an attempt is made to perform an action on a resource with a duplicated name, an error message is returned, and the action is not performed. This can be avoided by either renaming one of the resources or differentiating between the resources by appending a region name to the resource name.

When additional resource types, such as policy regions, Tivoli administrators, and profile managers are created, they are registered in the TNR. When a Tivoli application product is installed, new resource type (class) names are also registered in the TNR. When an instance of a resource is deleted, the entry in the TNR is removed.

Within Tivoli, most objects in the TNR can be divided into two categories:

- Distinguished objects associated with a single TMR, and resource objects that are typically associated with an application that models some system resource.

- A distinguished object is an object unique to a TMR with a specific name.

- Distinguished objects are used as service providers. Distinguished objects relate specifically to a TMR and are not exchanged during an update of resources between interconnected TMRs.

Resource objects are instances of Tivoli classes that are used to represent a system resource. There can be more than one instance of a resource type. The name of a resource object is usually specified by the administrator when the object is created. Resource objects are exchanged between TMRs during an interconnected TMR update of resources.

When a lookup for a resource occurs, it looks for an instance of the resource by the name and type. Normally, instances do not contain other instances, with the exception of managed nodes. To avoid the management and resource overhead needed to maintain these resources separately from managed node instances, the nested instances are contained in the managed node instance.

All resources in a TMR should be registered in the name registry when created, unregistered when deleted, and updated when their label is changed. For most resources in a TMF, these actions are handled automatically by the TMF.

- *(TMF.0041: CAT II) The TMR Administrator will be responsible for all TMR updates.*

B.2.2.3.8 Profiles

A profile is a collection of Tivoli application-specific information. Each item in a profile contains system configuration information. The information in a profile is specific to the particular profile type. Profile records are stored in a platform-independent format that allows the same profile records to be distributed across an environment that contains multiple platforms.

- *(TMF.0042: CAT III) The IAO will ensure that the Software Support personnel and the TMR Administrator are responsible for creating and installing profiles.*

B.2.2.3.9 Profile Managers

A profile manager is a container where individual profiles can be created and organized into groups of profiles. In addition subscribers can be linked to them. Profile managers are created within a policy region. A profile manager can logically be viewed as having two sections: a profile section and a subscriber section. Profile managers control the distribution of profiles to subscribers. Subscribers to a profile manager can be in the same policy region as their profile manager or in other policy regions.

- *(TMF.0043: CAT II) The TMR Administrator will be responsible for the management and administration of all Profile Managers.*

B.2.2.4 Tivoli Management Framework Related Products

In addition to the above services provided by the TMF, the following sub sections describe TMF related products. These products can be used to enhance the capabilities of the TMF and the other Tivoli application products.

B.2.2.4.1 Tivoli Application Extension Facility (AEF)

The Tivoli AEF is used to dynamically customize the other Tivoli products by adding site-specific options or values. The Tivoli AEF can be used to add fields to a dialog, create custom attributes and methods for application resources, and create custom icons and bitmaps. It should be noted that even though the AEF is able to extend the capabilities of the other products, it does not change the primary functions of the other Tivoli products. In order to perform many of the functions of the Tivoli AEF a user/administrator must have a role of super.

- *(TMF.0044: CAT II) Use of the Tivoli AEF will be restricted to the software support personnel and TMR Administrator.*

- *(TMF.0045: CAT II) The TMR Administrator will be responsible for all installations and updates to the Tivoli AEF.*

B.2.2.4.2 Tivoli Event Integration Facility (EIF)

The Tivoli Event Integration Facility EIF is a toolkit that can be used to map events from a product resource, or component into a format compatible with Tivoli Enterprise Console and develop additional adapters that are tailored to a specific network environment. Event adapters are used by Tivoli to monitor managed resources and send events to the Tivoli Enterprise Console or other products such as Managed Objects. The Tivoli EIF can be used to create event listeners that receive events. As a result, event listeners can be placed in a TMR where there is a need to distribute events to other management applications. Additionally, the Tivoli EIF can be used to filter events reducing event traffic on the network and the event server.

- *(TMF.0046: CAT II) The software support personnel will be responsible for all Event monitor creation.*

- *(TMF.0047: CAT II) The TMR Administrator will be responsible for ensuring that all monitors are installed in the TMR.*

- *(TMF.0048: CAT III) The Policy Region Administrators will be responsible for ensuring that all monitors are run at the sites.*

B.2.2.4.3 Tivoli Application Development Environment (ADE)

The Tivoli Applications Development Environment ADE is used to design and develop applications for distributed object systems. The Tivoli ADE contains programming tools that can be used for creating new custom management applications on top of TMF.

- *(TMF.0049: CAT II) The TMR Administrator will be responsible for controlling all access to the ADE.*

B.2.2.5 Supported Platforms

The TMF software is designed to run on open system architecture platforms. The following table describes the platforms and the minimum software release levels:

Platform	Software Releases
AIX	AIX 4.3, 4.3.1, or 4.3.2.
HP-UX	HP-UX 10.20 or 11.*x*
Solaris	Solaris 2.6.*x*, Solaris 2.7
SunOS	OS/2 Warp 4.0 or Version 4.5 as a Gateway
Windows NT	NT 4.0

B.2.2.6 Files

The TMF installation process installs the following TMF files on the Tivoli server or managed nodes: Libraries, Binaries, Command reference pages (manual pages), X11 resource files, Message catalogs, Databases (server and client), and Environment setup files. These directories can be installed in the root directory (/) or in an installation directory that is created by the TMR Administrator.

As part of the installation process, certain script system variables are set. The variables are used to make the installation flexible when dealing with different platforms. Some of the variables established are BINDIR, DBDIR, INTERP, LIBDIR, and MANPATH. Appendix B.8 will define these and others and cite their usage when applicable. For UNIX platforms the use of a $ will precede the variable and for Windows platforms the % sign will be used before and after the symbol.

The TMF software is installed on a TMR server or managed node into a series of Tivoli directories. The following table describes the TMF directory structure that is created upon initial installation.

File	Description
/cdrom/cdrom0	Identifies the path to the CD-ROM image.
/Tivoli/bin	Identifies where the binaries are located.
/Tivoli/lib	Describes where the libraries are installed.
/Tivoli/database	Identifies where the database is installed.
/Tivoli/man	Describes where the manual pages are installed.
/Tivoli/X11	Identifies where the X11 application defaults are installed.
/Tivoli/cat	Identifies where the message catalogs are installed.

All directories containing Tivoli binaries will be executable by authorized personnel. Write permission is required on the file server during the installation of TMF and Tivoli applications and to the task subdirectories during task creation and modification.

NOTE: According to Tivoli, the TMF software does not contain any setuid executable files, which would cause a security compromise.

- *(TMF.0050: CAT II) The SA will ensure that the Tivoli directories are protected in accordance with the UNIX and Windows STIGs, Appendix B.8, of this document, and DODI 8500.2.*

- *(TMF.0051: CAT II) The SA will ensure that all directories containing Tivoli binaries will be executable by only authorized personnel.*

- *(TMF.0052: CAT II) The SA will ensure that write permissions that are required on the file server during the installation of TMF, Tivoli applications and task subdirectories are restricted to software support and TMR Administrator personnel.*

- *(TMF.0053: CAT II) The SA will ensure that write permissions that are required on the file server during task creation and modification are restricted to software support and TMR Administrator personnel.*

- *(TMF.0054: CAT II) The SA will ensure that the Tivoli script programs are protected in accordance with the UNIX and Windows STIGs.*

It should be noted that the Tivoli environment setup files are created in the **/etc/Tivoli** directory. This directory contains the following subdirectories:

Directory	Contents
bin	Contains the Perl binaries required for many Tivoli operations.
lib	Contains the Perl language library required for many Tivoli operations.
tll.conf	Contains configuration information for the Task Library Language.

- *(TMF.0055: CAT II) The SA will ensure that the Tivoli environment setup files are protected in accordance with the UNIX and Windows STIGs, Appendix B.8 of this document, and DODI 8500.2.*

The following files are created during installation in the **/etc/Tivoli** files:

oserv.rc	Contains scripts to start or stop the object dispatcher.
setup_env.sh	Sets up the Tivoli system variables if you are using the Bourne shell. Run this script before you start the Tivoli Desktop.
setup_env.csh	Sets up the Tivoli system variables if you are using C shell. Run this script before you start the Tivoli Desktop.

B.2.3 General Security Considerations

There are many areas, which must be addressed when securing an Enterprise. They range from platform security, to network security, to application security, to data security to even personnel security. Since the Tivoli Enterprise Management software affects many areas of an Enterprise, each Tivoli product has been designed to provide their own security controls which work in conjunction with already existing security controls in the Enterprise. Additionally, the TMF security controls not only are used by many of the Tivoli application products but work in conjunction with the security controls provided by the other Tivoli application products.

During the installation of the TMF software the following security areas must be considered:

- Protect the TMF files and directories in accordance with the platform STIG requirements.

- Ensure that the TMR server is located in a restricted area and restrict the TMR server from unauthorized access (internally and externally).

- Ensure that all Firewall and Tivoli Port restrictions follow the guidance of the *Network STIG*.

- Ensure that SSL communications is enabled for each managed node.

- Ensure that Login Names are assigned in accordance with Platform and ESM STIG requirements.

- Ensure that Desktop interfaces are restricted behind the site firewall.

- Limit authorization role assignment based on specific job function or need.

- Restrict authorization roles at the policy region level.

- Restrict the root installation userid for the TMF software to the Central Tivoli Administrator.

- Restrict port usage to authorized ports as specified in the *Network STIG*.

- Encrypt file and message transfers using DES encryption.

- Use one-way connections wherever possible.

- Because some Tivoli applications do not always use TMF services for network communication, and do not honor port ranges that have been set, ensure that no communication channels can be run between resources on either side of a firewall.

- *(TMF.0056: CAT I) The IAO will maintain written procedures for the handling of the introduction of classified information into the Enterprise/TMR.*

- *(TMF.0057: CAT II) The IAO will ensure that contingency processing plans exist for the Enterprise, TMR, and components.*

- *(TMF.0058: CAT II) The IAO will maintain on file documentation describing the results of the performance of annual contingency testing on the Tivoli Enterprise and TMR.*

- *(TMF.0059: CAT I) The IAO will ensure that the SA has applied all IAVAs to the components of the Tivoli Enterprise.*

- *(TMF.0060: CAT III) The IAO will ensure that the TMR Administrator maintains an architecture configuration of the Tivoli Enterprise.*

- *(TMF.0061: CAT II) The IAO will ensure that network and platform SRRs are conducted on the Tivoli Enterprise.*

- *(TMF.0062: CAT II) The IAO will ensure that the SA has restricted the tnsnames.ora file from unauthorized access and update.*

- *(TMF.0063: CAT II) The SA will ensure that unauthorized software does not exist on the TMR server, TMR gateways, and managed nodes.*

B.2.3.1 Authorization Roles

The following tables are categorized by generalized function. Each table describes a specific activity to be performed, the context (global TMR or local Policy Region level), and the Tivoli role(s) needed to perform the function. These tables are included in the general security section to assist in determining the role authorization needed to perform a specific function.

B.2.3.1.1 Installation Function Table

This table is designed to assist in determining the role needed and the context level for installing of servers, managed nodes, products and patches.

Activity	Context	Role
Install a TMR server on a specific UNIX or Windows server.	Tivoli server	N/A (root access or Administrator privileges)
Install a managed node in a TMR. Remove a managed node from a TMR.	Policy region	install_client
Install a product or an application in the TMR. Install a patch or upgrade to a product or application in the TMR.	Tivoli region	install_product

B.2.3.1.2 Repeater Configuration Function Table

This table is designed to assist in determining the role needed and the context level for creating, deleting, tuning, and performing distributions.

Activity	Context	Role
Create a repeater. Delete a repeater. Tune a repeater for network load, maximum number of connections, maximum amount of memory, and maximum amount of disk paging space. Query an MDist repeater for its current configuration and active distributions.	Tivoli region	senior
Cancel, pause, and resume a distribution. Delete distributions from the database.	Tivoli region	senior or both Dist_control and RIM_view

B.2.3.1.3 Administrator Management Function Table

This table is designed to assist in determining the role needed and the context level for performing administrator and resource management.

Activity	Context	Role
Create an administrator. Delete an administrator. Add or remove resource roles. Add or remove login names. Add or remove notice group subscriptions. Edit the properties of a Tivoli administrator.	Administrators collection	senior

Activity	Context	Role
Add or remove TMR roles.	Tivoli region	senior
Create a collection and add resources to it. Drag and drop resources onto an administrator Desktop.	Tivoli Desktop	user

B.2.3.1.4 Managed Node Management Function Table

This table is designed to assist in determining the role needed and the context level for installation and maintenance of managed nodes.

Activity	Context	Role
Install a managed node. Remove a managed node	Policy region	install_client
Add an entry to the IP interface list of a managed node. Remove an entry from the IP interface list of a managed node.	Policy region containing a managed node	admin
Display the properties window of a managed node.	Policy region containing a managed node	user
Open a managed node window. Open an X terminal on a UNIX managed node.	Policy region containing a managed node	user and a system account on the managed node

B.2.3.1.5 Tivoli Region Connection Support Function Table

This table is designed to assist in determining the role needed and the context level for managing the interfacing of TMRs.

Activity	Context	Role
Disconnect Tivoli regions. Make a remote connection between two Tivoli regions. Make a secure connection between two Tivoli regions.	Tivoli region	super
Exchange resource information between Tivoli regions. Schedule an exchange of resource information between Tivoli regions.	Tivoli region	senior

B.2.3.1.6 Policy Region Management Function Table

This table is designed to assist in determining the role needed and the context level for the managing of Policy Regions.

Activity	Context	Role
Create a top-level policy region.*	Tivoli region	senior and policy
Create a subregion.*	Parent policy region	senior and policy
Add or remove a managed resource type to or from a policy region. Assign or change the policy for a managed resource type.	Policy region	senior and policy
Delete a subregion.	Parent policy region	senior
Change the name of a policy region. Delete a top-level policy region.	Policy region	senior
Move resources from a policy region to another policy region.	Policy region	senior
Check policy in a policy region.	Policy region	admin
Open a policy region window.	Policy region	user

*NOTE: The assignment of senior and/or policy roles depends on the options specified on the command.

B.2.3.1.7 Administration of Profiles and Profile Managers Function Table

This table is designed to assist in determining the role needed and the context level for managing profiles, profile managers, profile distribution.

Activity	Context	Role
Change policy on the records of a profile in a profile manager. Create a profile manager in a policy region. Delete a profile manager from a policy region. Edit a profile manager. Clone a profile in a profile manager. Copy a profile in a profile manager. Create a profile in a profile manager. Delete a profile from a profile manager.	Policy region	senior
Move a profile manager from one policy region window to another policy region window. Move a profile from one profile manager window to another profile manager window.	Source and destination policy region	senior
Set policy on the records of a profile in a profile manager.	Policy region	senior

Activity	Context	Role
Synchronize a profile in a profile endpoint.		
Distribute one or more profiles from a profile manager. Subscribe a managed node or another profile manager. Remove a subscription for a managed node or another profile manager.	Policy region	admin

B.2.3.1.8 Management of Task Libraries Function Table

This table is designed to assist in determining the role needed and the context level for managing tasks, task libraries, and jobs.

Activity	Context	Role
Change the policy for a task library in a policy region. Create a task library in a policy region. Delete a task library from a policy region. Set the policy for a task library in a policy region.	Policy region	senior
Move a task library from a policy region window.	Source and destination policy region	senior
Create a job or task in a task library. Delete a job or task from a task library. Edit a job or task in a task library.	Policy region	admin
Schedule a job for future execution.	Job	admin
Execute a job.	Job	The role defined in the task or job specification
Execute a task through either drag-and-drop or double-click. Save the output from task execution. Save the output from job execution.	Task	The role defined in the task or job specification

B.2.3.1.9 Notice Group(s) Management Function Table

This table is designed to assist in determining the role needed and the context level for the management of notices and notice groups.

Activity	Context	Role
Expire notices in a notice group. Set notice expiration length for a notice group.	Tivoli region	senior
Combine multiple related notices in a notice group into a single listing. Display old notices in a notice group. Filter notices in a notice group. Forward one or more notices through e-mail. Mark notices in a notice group as read or unread.	Tivoli Desktop	user

Activity	Context	Role
Read notices in a notice group. Save notices in a notice group to a file. Sort notices in a notice group.		

B.2.3.1.10 Scheduler Management Function Table

This table is designed to assist in determining the role needed and the context level for managing the operation of the scheduler and scheduled jobs.

Activity	Context	Role
Start the scheduler.	Tivoli region	super
Disable or enable a scheduled job. Edit a previously scheduled job. Remove a previously scheduled job. Schedule a job.	Tivoli region	admin
Browse the list of scheduled jobs. Change the information displayed in the list of scheduled jobs. Find one or more jobs displayed in the list of scheduled jobs. Sort the jobs displayed in the list of scheduled jobs.	Scheduler	user

B.2.3.1.11 Maintenance Support Function Table

This table is designed to assist in determining the role needed and the context level for managing and administering the TMR database.

Activity	Context	Role
Check the integrity of the TMR database. Enable or disable diagnostic tracing. Run the odadmin and odstat commands. Place the Tivoli region in single-user mode.	**Tivoli region**	**super**

- *(TMF.0064: CAT II) The IAO will ensure that the Tivoli roles are restricted by job responsibilities.*

- *(TMF.0065: CAT II) The TMR Administrator will be responsible for the assignment of Policy Region Administrator roles in a TMR.*

- *(TMF.0066: CAT II) The Policy Region Administrators will be responsible for the assignment of user roles in a Policy Region.*

B.2.3.2 Tivoli Management Facility Commands

The Command Line Interface CLI enables administrators to perform functions by entering commands in place of using pre-specified panels. In order for an administrator to perform management functions using commands, the administrator must be authorized through the local platform security and through Tivoli. The following tables describe actions that can be performed using Tivoli commands. The tables are included to assist in quickly determining by action the command needed to perform the activity. It should be noted that a user must be authorized at the platform level and through Tivoli authorization roles.

B.2.3.2.1 Administrator Type Commands

This table is designed to assist in determining the command needed to perform administrative activities.

Activity	Command
Add, remove, or display root authority of Tivoli Administrators in a Tivoli region.	wauthadmin
Create a new Tivoli administrator.	wcrtadmin
List information about a Tivoli administrator.	wgetadmin
List and modifies user login mapping.	widmap
Change information about a Tivoli administrator.	wsetadmin
Determine the operating system locale to use for a Tivoli server or managed node.	wsetlang

B.2.3.2.2 Configuration Management Type Commands

This table is designed to assist in determining the command needed to perform configuration management functions.

Activity	Command
Create a new profile or clones an existing profile.	wcrtprf
Create a profile manager.	wcrtprfmgr
Distribute one or more profile copies.	wdistrib
Retrieve subscription copies of one or more profiles.	wgetprf
List the subscribers of a profile manager.	wgetsub
List the profile managers to which a host NIS domain or profile manager subscribes.	wlssub
Populate a profile from system files.	wpopulate
Enable or disable a profile manager to operate in dataless mode.	wsetpm
Subscribe Tivoli resources to a profile manager.	wsub
Uninstall Tivoli applications from a specified node or from the entire Tivoli region.	wuninst
Remove Tivoli resources from the subscription list of a profile manager.	wunsub
Validate a profile against its validation policy.	wvalidate

B.2.3.2.3 Endpoint and Gateway Type Commands

This table is designed to assist in determining the command needed to perform gateway and endpoint management.

Activity	Command
Start the endpoint daemon (lcfd) on an endpoint and install or remove the daemon as a service on a Windows NT, Windows 2000 or Windows XP operating system.	lcfd
Start or stop the endpoint daemon (lcfd) on UNIX endpoints.	lcfd.sh
Install an endpoint on an OS/400® system.	w4inslcf.pl
Add an entry to the path statement in the registry hive of the current control set (Windows only).	waddpath
Perform automatic upgrade of an endpoint client.	wadminep
Remove a block of statements from a file.	wclrblk
Remove a single line from a file.	wclrline
Enable an .NCF configuration program to copy a file (NetWare only).	wcpyfile
Create an endpoint Gateway.	wcrtgate
Delete an endpoint.	wdelep
Delete an endpoint Gateway.	wdelgate
Verify the amount of disk space available (DOS Windows and NetWare only).	wdskspc
Modify the groups, variables, and values in an .INI file.	weditini
Perform actions on endpoint information contained in the endpoint list.	wep
Provide control and configuration for the endpoint manager.	wepmgr
Upgrade an endpoint to the newest software. NOTE: Replaces wadminep upgrade.	wepupgd
Start, stop, or list the properties of an endpoint gateway.	wgateway
Retrieve the subkey listing in a registry hive (Windows only).	wgetkey
Retrieve a registry subkey (Windows only).	wgetval
Insert a block of statements into a file.	winsblk
Insert a single line into a file.	winsline
Install an endpoint on UNIX, Windows XP, Windows NT, and Windows 2000 workstations.	winstlcf
List all the endpoints subscribed to a profile manager.	wlsendpts
Merge groups and variables from one .INI file into another.	wmrgini
Initiate a system restart and optional restart (Windows only).	wrestart
Replace a block of statements in a file.	wrplblk
Replace a single line in a file.	wrplline
Set the return code from a batch file for a configuration program.	wseterr
Set a registry key value (Windows only).	wsetval

B.2.3.2.4 httpd Type Commands

This table is designed to assist in determining the command needed to perform Tivoli web server administration.

Activity	Command
Register an HTTP 1.0 authentication realm with the HTTP daemon.	waddrealm
Delete a registered HTTP 1.0 authentication realm. The HTTP daemon must be restarted before the realm is actually deleted.	wdelrealm
Uninstall the Tivoli HTTP service or forwards HTTP request to a third-party HTTP server.	whttpd
List the currently registered HTTP 1.0 authentication realms.	wlsrealms
Start the Tivoli HTTP daemon.	wstarthttpd
Stop the Tivoli HTTP daemon.	wstophttpd

B.2.3.2.5 Installation Type Commands

This table is designed to assist in determining the command needed to perform installation activities.

Activity	Command
Install, update, or remove the Tivoli object dispatcher service in the Windows Service Manager.	oinstall
Install Tivoli clients.	wclient
Copy installation images from a CD to a system directory.	wcpcdrom
Install a Tivoli product.	winstall
Installs an endpoint on a UNIX, Windows NT, Windows XP, or Windows 2000 workstation.	winstlcf
Specify the mail server used by TMF on Windows operating systems.	wmailhost
Install a Tivoli patch.	wpatch
Install the Tivoli management region server.	wserver
Set the properties of the Tivoli Authentication Package (Windows only).	wsettap

B.2.3.2.6 Interregion Type Commands

This table is designed to assist in determining the command needed to perform inter-region management.

Activity	Command
Connect two Tivoli regions.	wconnect
Disconnect two Tivoli regions.	wdisconn
Search for the object reference of a resource.	wlookup
List the current Tivoli region connections or information about a single connection.	wlsconn
Register a resource with the name registry.	wregister
Display or changes the name of the local Tivoli region.	wtmrname
Update resources in the local name registry.	wupdate

B.2.3.2.7 Kerberos Type Commands

This table is designed to assist in determining the command needed to perform Kerberos administration.

Activity	Command
Network utility used for Kerberos database administration.	kadmin
Network daemon used for Kerberos database administration.	kadmind
Use to destroy a Kerberos key distribution center database.	kdb_destroy
Kerberos key distribution center database editing utility.	kdb_edit
Initialize a Kerberos key distribution center database.	kdb_init
Kerberos key distribution center database utility.	kdb_util
Destroy Kerberos tickets.	kdestroy
Introduce the Kerberos authentication service.	kerberos
Kerberos login utility.	kinit
List currently held Kerberos tickets.	klist
Change the Kerberos password for a user.	kpasswd
Fetch and stores Kerberos ticket-granting-ticket using a service key.	ksrvtgt
Stash the Kerberos key distribution center database master key Low-Level Maintenance Commands.	kstash
Extract individual options from an option list returned by the idlinput command.	idlarg
Get or set implementation attributes.	idlattr
Provide a method of invoking Interface Definition Language (IDL) operations from the shell command line.	idlcall
Raise exceptions for a shell method.	idlexception
Get the input or inout options list to a shell method.	idlinput
Format inout or output options or the result (if any) of a shell method.	idlresult
Create a readable version of a transaction log file.	logls
Perform an object call from the shell.	objcall
Manage object dispatchers.	odadmin
List the contents of an object database.	odbls
List the status of current and recent object calls.	odstat
Provide operations to control and configure object dispatchers.	oserv
Force a change of state of a running transaction.	tmcmd
Display the status of current transactions and locks.	tmstat
Set the name of the local host in the Windows registry (Windows NT, Windows XP, and Windows 2000 only).	wlocalhost
Place the current Tivoli region in maintenance mode.	wlocktmr
Specify the mail server used by TMF, Windows NT, Windows XP, and Windows 2000 systems.	wmailhost

B.2.3.2.8 Managed Node Type Commands

This table is designed to assist in determining the command needed to perform managed node administration.

Activity	Command
Create a managed node.	wclient
Print out the current date and time of the managed node.	wdate
Print the number of free kilobytes available in the specified directory (file system) of the specified managed node.	wdiskspace
Print the host ID of the specified managed node.	whostid
Query or changes the IP interfaces on a managed node.	wifconfig
Print out the path of the installation directory of the specified managed node.	winstdir
Print the interpreter type of the specified managed node.	winterp
Return the properties of a managed node.	wmannode
Report the amount of physical memory of a managed node.	wmemsize
Print the time zone value of the specified system.	wtimezone
List operating system information.	wuname
Remove TMF files from a managed node.	wunstmn
Start an Xterminal session on a UNIX managed node.	wxterm

B.2.3.2.9 Multiplexed Distribution Type Commands

This table is designed to assist in determining the command needed to perform MDist and endpoint management and administration.

Activity	Command
Provide configuration information to the endpoint daemon, (lcfd), such as enabling wake-on-LAN functionality on the endpoint	lcfd
Perform administrative operations on an endpoint, such as upgrading the endpoint daemon or generating the endpoint, wake-up packet for wake-on-LAN operations	wadminep
Manage MDist 2 repeater depots	wdepot
Perform actions on endpoint information contained in the endpoint list, such as setting a Windows endpoint to receive and control MDist 2 distributions through the use of the Tivoli Mobile Computing console	wep
Configure MDist 2 repeaters and manages distributions	wmdist
Start the Distribution Status console of the MDist 2 service, from the managed node on which this command is run	wmdistgui
Create a repeater on a managed node (for both MDist and MDist 2), configures MDist repeaters and manages MDist distributions	wrpt

B.2.3.2.10 Notification Type Commands

This table is designed to assist in determining the command needed to perform notice and message management.

Activity	Command
Broadcast a message to all Tivoli Desktops	wbroadcast
Expire notices from a notice group	wexpnotif
List notices on an administrator bulletin board	wlsnotif
Translate standard input into a message structure and sends it to the notification server	wsndnotif
Connect to the notification server and displays new notices as they are posted	wtailnotif

B.2.3.2.11 Policy Type Commands

This table is designed to assist in determining the command needed to perform policy and policy region management and administration.

Activity	Command
Check policy region members against a policy	wchkpol
Create a new policy object for a class	wcrtpol
Create a policy region	wcrtpr
Delete a default policy object	wdelpol
Delete a policy region	wdelpr
List a default policy object	wgetdfpol
List the body and constant values of an endpoint policy script	wgeteppol
List the body or constant value of a default or validation policy method	wgetpolm
List the properties of a policy region	wgetpr
List available policy default and validation objects for a Tivoli resource	wlspol
List policy methods for a Tivoli resource	wlspolm
Replace an endpoint policy script that has been modified	wputeppol
Replace the body of a policy method	wputpolm
Set the default policy for a class	wsetdfpol
Assign the policy used in a policy region enables or disables policy validation and adds or removes a managed resource in a policy region	wsetpr

B.2.3.2.12 Query Type Commands

This table is designed to assist in determining the command needed to perform query type activities.

Activity	Command
Create a query library	wcrtqlib
Create a query	wcrtquery
List information about a query	wgetquery

Activity	Command
Query the database for inventory information and returns a list of object IDs and object labels that match the query criteria	wruninvquery
Run a query and returns the results to either standard output or a file	wrunquery
Edit the properties of a query	wsetquery

B.2.3.2.13 RDBMS Interface Module (RIM) Type Commands

This table is designed to assist in determining the command needed to perform RIM administration.

Activity	Command
Createa RIM object	wcrtrim
List information about a RIM object	wgetrim
Move a RIM object to another managed node	wmvrim
Verify the connectivity and functionality of a RIM object	wrimtest
Enable or disables tracing for RIM objects	wrimtrace
Change the database information for a RIM object	wsetrim
Set the RIM password for a RIM object database	wsetrimpw

B.2.3.2.14 Revision Control System (RCS) Type Commands

This table is designed to assist in determining the command needed to perform RCS administration and management.

Activity	Command
Check in RCS revisions	wci
Check out RCS revisions	wco
Identify files	wident
Change RCS file attributes	wrcs
Compare RCS revisions	wrcsdiff
Merge RCS revisions	wrcsmerge
Print log messages and other information about RCS files	wrlog

B.2.3.2.15 Scheduler Type Commands

This table is designed to assist in determining the command needed to perform scheduled job administration.

Activity	Command
Remove jobs from the scheduler	wdelsched
Edit a job that currently exists in the scheduler	wedsched
Disable or enables scheduled jobs	wenblsched
Retrieve information about jobs currently scheduled	wgetsched
Schedule a job that exists in a task library	wschedjob
Start the Tivoli scheduler	wstartsched

B.2.3.2.16 Task Library Type Commands

This table is designed to assist in determining the command needed to perform task, job, and task library management.

Activity	Command
Create a job in a task library	wcrtjob
Create a task in a task library	wcrttask
Create a task library	wcrttlib
Delete a job from a task library	wdeljob
Delete a task from a task library	wdeltask
Control the distribution of task binaries for a task library	wdisttask
List the properties of a job	wgetjob
List the properties of a task	wgettask
List the properties of a task library	wlstlib
Run a job in a task library	wrunjob
Run a task in a task library	wruntask
Set the properties of a job	wsetjob
Set the properties of a task	wsettask
Abort a task transaction and rolls back any uncommitted changes	wtaskabort
Import and exports task library definitions	wtll

B.2.3.2.17 Miscellaneous Type Commands

This table is designed to assist in determining the command needed to perform Desktop administration.

Activity	Command
Start the Tivoli graphical user interface	tivoli
Check status of all managed nodes	vdisp
Add an icon to a Windows Program Manager group (Windows 95- Windows NT, Windows XP, and Windows 2000 only)	waddicon

Activity	Command
Retrieve a translated string from a local message catalog and binds any variables	wbindmsg
Back up and restores Tivoli databases	wbkupdb
Save custom dialogs in TMF or a Tivoli application before an upgrade to a new version of TMF or the application	wcatcher
Change the current working collection	wcd
Associate a dependency set with a method header	wchdep
Verify and repairs the Tivoli database	wchkdb
Verify and updates references to a specific dispatcher number from parts of the Tivoli database	wchknode
Delete objects from the Tivoli database	wdel
Set the message that is displayed when the Tivoli Desktop is started	wdtmsg
Specify dependencies that a method needs to run	wdepset
Display all instances of a resource type	wgetallinst
Convert the characters or sequences of characters in a file from one code set to another code set	wiconv
Install behavior for an endpoint resource type	winstendpt
Set the properties of the Tivoli Authentication Package on a Windows client	wlcftap
Link an object into a collection	wln
Return the path for the localized file or directory	wlocpath
List the member objects of a collection	wls
List the products and patches installed in a Tivoli management region	wlsinst
Perform a three-way file merge	wmerge
Merge custom dialogs into TMF or a Tivoli application after upgrading	wmrgaef
Move objects between collections	wmv
Attempt to contact the object dispatcher on a host	wping
Print the current working collection	wpwd
Refresh a Tivoli collection window	wrefresh
Remove objects from a collection	wrm
Remove a managed node from a Tivoli environment	wrmnode
Retrieve passwords and launches commands	wrunas
Encrypt and stores passwords	wsetpkey
Collect problem information from users to send to a customer support representative	wsupport
Display the name of the directory in which Tivoli products create temporary files	wtemp
Provide information to debug methods	wtrace

- *(TMF.0067: CAT II) The SA will ensure that the Tivoli commands are restricted from unauthorized access and usage at the platform level.*

- *(TMF.0068: CAT II) The TMR Administrator will ensure that the Tivoli commands are restricted at the Policy Region level.*

B.2.3.3 TMF Encryption and Encryption Levels

The TMF provides a facility for encrypting both Tivoli security credentials and sensitive data. Security credentials include such sensitive data as inter-region passwords, Tivoli administrator logins and roles, and authentication requests and data. An encryption service is available for use when applications are designed to allow for the protection of sensitive application data. Tivoli User Administration uses this service when handling passwords for user accounts. The TMF also provides data encryption through the use of the TMF SSL-A run-time package, that is an implementation of the Secure Sockets Layer (SSL) Protocol, Version 3.0. The three levels of data encryption that are provided by the TMF are as follows:

- None - If none is specified as the encryption level, the TMF does not encrypt any of the Tivoli security data from casual viewing.

- Simple- If simple is specified as the encryption level, the TMF provides an XOR-based encryption scheme that protects Tivoli security credentials from casual viewing.

- DES- If DES is specified as the encryption level, the TMF protects Tivoli security credentials from unauthorized network intrusions.

- *(TMF.0069: CAT II) The IAO will ensure that encryption is in use by Tivoli.*

- *(TMF.0070: CAT II) The TMR Administrator will ensure that the DES encryption option has been turned on in Tivoli on all Managed Nodes.*

B.2.3.3.1 Intraregion and Interregion Encryption

In order to further secure data and credentials, the TMF software is able to support the creation of different encryption levels for activity within a TMR or between connected TMRs. The TMF software supports different encryption passwords for intra and inter region operations, even if they both types use the same encryption level. Any mix of encryption levels and passwords can be used for Tivoli intra-region, inter-region, and intra-installation operations.

It should be noted that by changing the encryption password during the installation of the TMF software, only authorized users with the new password can install the TMF software and other Tivoli application software.

- *(TMF.0071: CAT II) The TMR Administrators will ensure that the DES encryption is in use between TMRs.*

B.2.3.3.2 Secure Sockets Layer Data Encryption

In order to protect the privacy of network traffic in the Tivoli environment, the TMF supports SSL encryption on oserv-to-oserv communication channels. SSL can be enabled on a per-node basis, which results in the possibility of a mix of SSL and non-SSL nodes communicating within a TMR and/or across TMR boundaries. In order for SSL to run on a managed node the TMR server must first be set to be SSL-capable before any managed nodes can use SSL connections.

Then, the network security level on each managed node must be enabled. (This setting determines how the managed node logs in to the server.) And finally, the cipher list on each managed node must be created to dictate the strength of the encryption used by SSL.

The Tivoli *odadmin set_network_security* command and the *odadmin set_crypt_level* commands are used to set the network security level and encryption list.

- *(TMF.0072: CAT II) The TMR Administrator will ensure that security level is set to FORCE_SSL for each managed node.*

- *(TMF.0073: CAT II) The TMR Administrator will ensure that encryption level is set to DES on each managed node.*

Ciphers dictate the SSL encryption strength. During SSL negotiation, both the initiator and receiver of the SSL connection share their cipher list. The SSL session is established through a handshake sequence between the managed nodes. Normally, the managed node assigned the role of SSL server determines the cipher to use. The SSL server does this by checking its cipher list and selecting the first cipher that is also supported by the client. The server then uses the session keys and begins encrypted communications.

A user-defined cipher list can be changed providing the managed nodes are SSL-capable (have the SSL-A package installed). Ciphers specs for specific managed nodes, all clients, or all managed nodes in a Tivoli management region can be established using the *odadmin set_ssl_ciphers* command. The *odadmin set_ssl_ciphers* command sets ciphers on managed nodes, which protect the channel. By entering the oserv –E command on the Tivoli server itself, causes the new cipher values to become the new settings for the server. It should be noted that any managed node can be set to the default regardless of its SSL capabilities.

- *(TMF.0074: CAT II) The TMR Administrator will ensure that security level is set to FORCE_SSL for each managed node.*

- *(TMF.0075: CAT II) The TMR Administrator will ensure that encryption level is set to DES on each managed node.*

- *(TMF.0076: CAT II) The TMR Administrator will ensure that cipher list (0A,04,05) is in use.**

**NOTE: CJCSM 6510.01 dated 25 March 2003, Appendix H, Enclosure C specifies the NSA-NIST-certified cryptographic algorithms that are authorized for Department of Defense use.*

It should be noted that the TMF uses SSL to encrypt data only. The TMF comes with default keys and certificates. In order to improve authentication, the TMF allows for the replacement of the default keys and certificates with DOD approved ones.

- *(TMF.0077: CAT II) The IAO will ensure that SSL is in use during Managed Node communication.*

- *(TMF.0078: CAT I) The IAO will ensure that DOD approved certificates are in use by Tivoli.*

- *(TMF.0079: CAT I) The TMR Administrator will be responsible for ensuring that Tivoli default keys and certificates are removed from the Managed Nodes in the TMR.*

- *(TMF.0080: CAT I) The SA will be responsible for ensuring that Tivoli default keys and certificates are protected using platform security.*

B.2.3.3.3 TMF PKI Certificates

The TMF software comes with default PKI certificates and keys. In order to implement peer-to-peer authentication, the default Tivoli certificates and keys must be removed and replaced with DOD approved ones. The keystore on each managed node can be modified by using the iKeyman (gsk4ikm) key management utility, which is installed with the SSL-A package. The following table describes the locations of the private keys and certificates.

File	Description
$DBDIR/Tivoli.kdb	Contains the private key and trusted certificates of a managed node. Because it contains the private key, it must be protected from unauthorized access and use.
$DBDIR/TivoliCert.kdb	Contains only the trusted certificates. Both files contain the certificate of the Tivoli certificate authority as a trusted signer. This means that the keystores trust all TMF installations throughout the world. In addition, Tivoli.kdb contains a Tivoli global signing key, which is signed by the Tivoli certificate authority. When a managed node starts for the first time, the Tivoli global signing key automatically generates and signs a unique private key that is trusted by all Tivoli product installations.

Because these files permit access by TMF services that have root permissions and user applications that might not have root privileges, they must be restricted.

- *(TMF.0081: CAT II) The IAM will ensure that the Tivoli.kdb default file is replaced with a DOD authorized private key file.*

- *(TMF.0082: CAT II) The IAM will ensure that the TivoliCert.kdb default file is replaced with a DOD authorized trusted certificated file.*

B.2.3.3.4 Intraregion and Interregion Encryption

In order to further secure data and credentials, the TMF software is able to support the creation of different encryption levels for activity within a TMR or between connected TMRs. The TMF software supports different encryption passwords for intra and inter region operations, even if they both types use the same encryption level. Any mix of encryption levels and passwords can be used for Tivoli intra-region, inter-region, and intra-installation operations.

It should be noted that by changing the encryption password during the installation of the TMF software, only authorized users with the new password can install the TMF software and other Tivoli application software.

- *(TMF.0083: CAT II) The TMR Administrator will ensure that the DES encryption is in use between TMRs.*

B.2.4 Specific Security Considerations

A TMR server can be either a UNIX or Windows machine. Each machine has its own set of requirements that must be addressed. The following subsections describe areas that apply to both machines and where necessary address either the UNIX or Windows specifics.

Department of Defense (DOD) Instruction 8500.2, Information Assurance (IA) Implementation, documents IA controls that apply to DOD information systems. The following integrity and confidentiality controls from *DODI 8500.2* are the basis for the requirements in this section: Access for Need-to-Know; Audit Trail, Monitoring, Analysis, and Reporting; Changes to Data; Individual Identification and Authentication; Least Privilege; and Privileged Account Control.

The following subsections address the considerations and requirements as they pertain to the TMF software. When reviewing these requirements, the following must be noted:

- The specific directory names in the requirements reflect defaults indicated in vendor documentation. If a site or organization deploying Tivoli chooses other names, the requirements apply to the site-specific names.

- The structure of UNIX permissions can make it difficult to restrict file access to multiple groups of users. The approach taken in this section is to allow UNIX “world” access in cases where non-update (e.g., read or execute) access permission is required for users who may be members of multiple groups and there is no specific need for the files to be inaccessible to other users.

B.2.4.1 Tivoli Login Names

The creation of Tivoli Administrators requires specific considerations regarding login names. The following two subsections describe the login names that must be created for Tivoli Administrators and specific security considerations.

B.2.4.1.1 User Login Name

When an administrator is created, two login names are required to be entered. The first is the User Login Name. It must match a valid UNIX or Windows NT account name. The Tivoli Administrators will have an account that matches this login name on every system that they are going to manage. The *User Login Name* is the login name used when certain TME operations are performed. The following are examples of these operations: saving task output to a file, running an xterm from a managed node, and performing TME backups.

- *(TMF.0084: CAT II) The SA will ensure that each User Login Name meets both UNIX and Windows STIG guidelines.*

- *(TMF.0085: CAT I) The IAO will ensure that all Tivoli Administrators have SF41s on file.*

B.2.4.1.2 Set Login Name

The second login name required is used when an Administrator launches the Tivoli Desktop or wants to execute Tivoli commands from the CLI. In order to launch the Desktop from a system, the administrator must have a system account on that system that matches one of the *Set Login Names.* The only allowed login names will be in the form **username@ManagedNode**. This enhances security by forcing administrators to launch the Tivoli Desktop only from specific managed nodes.

- *(TMF.0086: CAT II) The TMR Administrator will restrict Policy Region Administrators login to the Policy Region Level.*

- *(TMF.0087: CAT III) The TMR Administrators will ensure that the Set Login Name is created for Policy Region Administrators in the above format.*

B.2.4.2 Authorization Roles

In a Tivoli Management Environment TME there are many different administrators/users. Each administrator/user may perform one or more functions. As a result, each administrator/user may have multiple roles assigned and at either the TMR (global) or Policy Region (local) level. The assignment of roles depends on the management function being performed by the individual.

NOTE: Under DISA the following types of individuals exist in a TMR: SSO Tivoli Development Team Personnel, System Administrators (SAs) and Database Administrators (DBAs) for Endpoints, System Administrators (SAs) for gateways, Managed Nodes and Tivoli servers including the Tivoli Event Consoles (TEC), Central Tivoli Administrators (TMR Administrators) and Site Tivoli Administrators (STAs), which are Policy Region Administrators.

The Central Tivoli Administrator (TMR Administrator) is responsible for the management of the entire Tivoli Production Environment. The Site Tivoli Administrator (Policy Region Administrator) is responsible for managing the use of Tivoli by users/customers at the local site. The Software Development Personnel are responsible for the Installation, Maintenance and

Testing of the Tivoli Application Software Products, to include the development of monitors, tasks and scripts. In addition the Software Development Personnel are responsible for resolving Tivoli software application product problems.

The following table describes the TMR administrators/users and the role(s) that will be assigned.

Authorized Personnel	Role	Context
Software Support Personnel	admin backup Dist_control install_client install_product restore super user	TMR, Policy Region TMR TMR TMR, Policy Region TMR TMR TMR TMR, Policy Region, Tivoli Desktop
TMR Administrator	admin backup Dist_control install_client install_product restore super user	TMR, Policy Region TMR TMR TMR, Policy Region TMR TMR TMR TMR, Policy Region, Tivoli Desktop
Policy Region Administrator	admin senior policy	Policy Region Policy Region Policy Region

B.2.4.3 TMF PKI Certificates

As stated earlier the TMF software comes with default PKI certificates and keys. The default PKI certificates, keys and passwords must be removed from the system and replace with DOD approved ones. In order to replace the default Tivoli certificates and keys the following must occur:

1. Remove the Tivoli certificate, Tivoli global signing key, and any generated TMF key from the keystores.

2. Add DOD approved certificates to both files. The trust hierarchy can be any granularity, ranging from a deployment-wide certificate encompassing all Tivoli management regions to trusting only certain managed nodes.

3. Add a DOD approved private key to Tivoli.kdb, signed by the certificate authority. The only requirement is that the label for the private key is Framework.

4. Change the default passwords for the Tivoli.kdb and TivoliCert.kdb files. TMF uses the password encrypted in the stash file to access either keystore. The names of the stashed password files are Tivoli.sth and TivoliCert.sth, respectively. The permission of each file must be equivalent to the keystore; that is, root, read, and write access only for Tivoli.sth; read permission for all users on TivoliCert.sth.

- *(TMF.0088: CAT II) The TMR Administrator will ensure that the default password for the Tivoli.kdb and the TivoliCert.kdb files is changed.*

- *(TMF.0089: CAT II) The TMR Administrator will ensure that the key database password in the stash file is protected.*

B.2.4.4 Tivoli Firewall Ports

Tivoli is configured to use ports 1023-5999 for bulk data transfers. These are known as unnamed ports or ports not assigned to any specific service such as FTP. Tivoli recommends the use of these ports. Ports outside this range should not be reserved for Tivoli unless documented with the IAO and IAM.

All ports must be open from source IP address to destination IP address (bi-directional) for creating managed nodes. The Ports Tivoli use will be restricted based on the list of ports in table below and the Network STIG. The configuration of the ports listed in the table must be bi-directional. The firewall and network administrators must coordinate opening the ports during the Tivoli implementation.

TCP Server Listening Port	TCP Client Ephemeral Port	Type of Connection	Protocol	Duration
94	Chosen from the assigned port.	Inter-ORB	TCP	Sustained
94	Chosen from the assigned port.	Inter-TMR	UDP	Sustained
162	Chosen from the assigned port.	SMNP	UDP	Sustained on host receiving SNMP trap.
512	Chosen from the assigned port.	Remote Execution Service	TCP	Sustained only at the beginning of managed node installation.
514	Chosen from selected range or assigned.	Shell (set installation password)	TCP	Sustained only at the beginning of managed node installation.
1414	Chosen from selected range or assigned.	MQ Series Server to MQ Series Server	TCP	Sustained

TCP Server Listening Port	TCP Client Ephemeral Port	Type of Connection	Protocol	Duration
[40,000 + 3(n)]. Chosen ports from selected range or assigned (3 ports per managed clients beginning at port 40,000)	Chosen from selected range or assigned.	Inter-Object Messaging (IOM)	TCP	Sustained
2501	Chosen from the assigned port.	Remote Control Target	TCP	Sustained
2600	Chosen from the assigned port.	ARM server and clients subagents	TCP	Sustained
4000	Chosen from the assigned port.	GEM console	TCP	Sustained at sites have GEM consoles.
4010	Chosen from the assigned port.	GEM to TEC server	TCP	Sustained at sites have GEM server.
4020	Chosen from the assigned port.	MQ Series	TCP	Sustained
5529	Chosen from the assigned port.	TEC_recv_agent - TEC listening port on Windows NT	TCP	Sustained
6543	Chosen from the assigned port.	User link - listening Port & Managed Node Listening Port	TCP	Sustained for DECC's Tivoli Administrator. Otherwise, access port can be closed.
8180	Chosen from the assigned port.	SSL daemon - Tivoli user administration "One Password" utility.	TCP	Port can be closed. One Password web-based utility is not implemented.
8890	Chosen from the assigned port.	TACF - Tivoli Security Management	TCP	This port can be closed until Security Management module is implemented.
8891	Chosen from the assigned port.	TACF-Tivoli Security Management	TCP	This port can be closed until Security Management module is implemented.

The *Network STIG* states that all Port 512 and 514 traffic will be denied. The *Network STIG* also states that if use of these ports is required, the using organization must request a waiver and control the traffic from source to destination. The Enterprise System Management Export Package requires ports 512 and 514 to be open between the Tivoli Managed Region (TMR) and the managed node for two hours during installation.

- *(TMF.0090: CAT II) The IAO will ensure if port 512 (rexec) or trusted relationships (.rhosts, hosts.equiv) are used for the installation, they are restored to the original state immediately after the installation.*

B.2.4.4.1 Tivoli Desktop Port Usage

The Tivoli Desktop for Windows is the interface running on an NT platform. It provides the same functionality as running the tivoli command on a UNIX managed node. For it to work in a firewall environment certain ports must be open. These ports are in addition to the ports listed in above section.

It should be noted that since each window opened on the Desktop requires an Inter-object Communication channel each window uses a port until it is closed. As more endpoints and managed nodes are added to the ESM environment, the number of opened ports may need to be increased.

Source		**Destination**		**Service/ Protocol**	**Description**
Component	Port	Component	Port		
Gateway	94	TMR	94	TCP	Initial opening of Desktop
Desktop	Random	TMR Server	94	TCP	Opening of windows within the Desktop
Desktop	Random	Gateway	94	TCP	Normal operation

Instead of opening all ports from the Tivoli Desktops, x-terminal can be used between the Desktop and the endpoint gateway.

B.2.4.5 TMF Commands and Authorized Users

The TMF provides commands to enable administrators to perform management functions by using the CLI. In order for the administrator to use commands the administrator must have the correct authorization role assigned. The following table describes the authorized user, the Tivoli Roles assigned to the individual and the Tivoli commands the individual is authorized to execute.

Individual Type	Tivoli Roles	Commands
Software Support Personnel	admin backup Dist_control install_client install_product restore super user	idlarg, idlattr, idlcall, idlexception, idlinput, idlresult, kadmin, kadmind, kdb_destroy, kdb_edit, kdb_init, kdb_util, kdestroy, kerberos, kinit, klist, kpasswd, ksrvtgt, kstash, lcfd, lcfd.sh, logls, objcall, odadmin, odbls, odstat, oinstall, oserv, tivoli, tmcmd, tmstat, vdisp, w4inslcf.pl, waddicon, waddpath, waddrealm, wadminep, wauthadmin, wbindmsg, wbkupdb, wbroadcast, wcatcher, wcd, wchdep, wchkdb, wchknode, wchkpol, wci, wclient, wclrblk, wclrline, wco, wconnect, wcpcdrom, wcpyfile, wcrtadmin, wcrtgate, wcrtjob, wcrtpol, wcrtpr, wcrtprf, wcrtprfmgr, wcrtqlib, wcrtquery, wcrtrim, wcrttask, wcrttlib, wdate, wdel, wdelep, wdelgate, wdeljob, wdelpol, wdelpr, wdelrealm, wdelsched, wdeltask, wdepot, wdepset, wdisconn, wdiskspace, wdistrib, wdisttask, wdskspc, wdtmsg, weditini, wedsched, wenblsched, wep, wepmgr, wepupgd, wexpnotif, wgateway, wgetadmin, wgetallinst, wgetdfpol, wgeteppol, wgetjob, wgetkey, wgetpolm, wgetpr, wgetprf, wgetquery, wgetrim, wgetsched, wgetsub, wgettask, wgetval, whostid, whttpd, wiconv, wident, widmap, wifconfig, winsblk, winsline, winstall, winstdir, winstendpt, winstlcf, winterp, wlcftap, wln, wlocalhost, wlocktmr, wlocpath, wlookup, wls, wlsconn, wlsendpts, wlsinst, wlsnotif, wlspol, wlspolm, wlsrealms, wlssub, wlstlib, wmailhost, wmannode, wmdist, wmdistgui, wmemsize, wmerge, wmrgaef, wmrgini, wmv, wmvrim, wpatch, wping, wpopulate, wputeppol, wputpolm, wpwd, wrcs, wrcsdiff, wrcsmerge, wrefresh, wregister, wrestart, wrimtest, wrimtrace, wrlog, wrm, wrmnode, wrplblk, wrplline, wrpt, wrunas, wruninvquery, wrunjob, wrunquery, wruntask, wschedjob, wserver, wsetadmin, wsetdfpol, wseterr, wsetjob, wsetlang, wsetpkey, wsetpm, wsetpr, wsetquery, wsetrim, wsetrimpw, wsettap, wsettask, wsetval, wsndnotif, wstarthttpd, wstartsched, wstophttpd, wsub, wsupport, wtailnotif, wtaskabort, wtemp, wtimezone, wtll, wtmrname, wtrace, wuname, wuninst, wunstmn, wunsub, wupdate, wvalidate, wxterm
TMR Administrator	admin backup	idlarg, idlattr, idlcall, idlexception, idlinput, idlresult, kadmin, kadmind, kdb_destroy, kdb_edit, kdb_init, kdb_util, kdestroy,

Individual Type	**Tivoli Roles**	**Commands**
	Dist_control install_client install_product restore super user	kerberos, kinit, klist, kpasswd, ksrvtgt, kstash, lcfd, lcfd.sh, logls, objcall, odadmin, odbls, odstat, oinstall, oserv, tivoli, tmcmd, tmstat, vdisp, w4inslcf.pl, waddicon, waddpath, waddrealm, wadminep, wauthadmin, wbindmsg, wbkupdb, wbroadcast, wcatcher, wcd, wchdep, wchkdb, wchknode, wchkpol, wci, wclient, wclrblk, wclrline, wco, wconnect, wcpcdrom, wcpyfile, wcrtadmin, wcrtgate, wcrtjob, wcrtpol, wcrtpr, wcrtprf, wcrtprfmgr, wcrtqlib, wcrtquery, wcrtrim, wcrttask, wcrttlib, wdate, wdel, wdelep, wdelgate, wdeljob, wdelpol, wdelpr, wdelrealm, wdelsched, wdeltask, wdepot, wdepset, wdisconn, wdiskspace, wdistrib, wdisttask, wdskspc, wdtmsg, weditini, wedsched, wenblsched, wep, wepmgr, wepupgd, wexpnotif, wgateway, wgetadmin, wgetallinst, wgetdfpol, wgeteppol, wgetjob, wgetkey, wgetpolm, wgetpr, wgetprf, wgetquery, wgetrim, wgetsched, wgetsub, wgettask, wgetval, whostid, whttpd, wiconv, wident, widmap, wifconfig, winsblk, winsline, winstall, winstdir, winstendpt, winstlcf, winterp, wlcftap, wln, wlocalhost, wlocktmr, wlocpath, wlookup, wls, wlsconn, wlsendpts, wlsinst, wlsnotif, wlspol, wlspolm, wlsrealms, wlssub, wlstlib, wmailhost, wmannode, wmdist, wmdistgui, wmemsize, wmerge, wmrgaef, wmrgini, wmv, wmvrim, wpatch, wping, wpopulate, wputeppol, wputpolm, wpwd, wrcs, wrcsdiff, wrcsmerge, wrefresh, wregister, wrestart, wrimtest, wrimtrace, wrlog, wrm, wrmnode, wrplblk, wrplline, wrpt, wrunas, wruninvquery, wrunjob, wrunquery, wruntask, wschedjob, wserver, wsetadmin, wsetdfpol, wseterr, wsetjob, wsetlang, wsetpkey, wsetpm, wsetpr, wsetquery, wsetrim, wsetrimpw, wsettap, wsettask, wsetval, wsndnotif, wstarthttpd, wstartsched, wstophttpd, wsub, wsupport, wtailnotif, wtaskabort, wtemp, wtimezone, wtll, wtmrname, wtrace, wuname, wuninst, wunstmn, wunsub, wupdate, wvalidate, wxterm
Policy Region Administrator	admin senior policy	kpasswd, lcfd, lcfd.sh, logls, odbls, odstat, tivoli, tmstat, wadminep, wbroadcast, wchkpol, wclient, wdate, wdskspc, weditini, wepmgr, wgetdfpol, wgeteppol, wgetjob, wgetpolm, wgetpr, wgetprf, wgetquery, wgetrim, wgetsched, wgetsub, wgettask, wlocalhost, wlocpath, wls, wlsconn, wlsendpts, wlsnotif, wlspol, wlspolm, wlssub, wrestart, wrlog, wruninvquery, wrunquery, wsndnotif, wsupport, wtailnotif, wtemp, wtimezone, wtrace

- *(TMF.0091: CAT II) The IAO will ensure that all commands are restricted to the above authorization roles.*

B.2.4.6 UNIX TMR Servers and Managed Nodes

This subsection is intended to provide guidance on security measures that should be observed when installing the TMF software on UNIX platforms. STIG issues and compliance will be the responsibility of the local SA and Security office.

In order to secure the TMF software and ensure the security of the TMF software the following must occur:

- Assign UNIX file permissions to all TMF files and directories.
- Restrict access to Tivoli to authorized users.
- Limit the assignment of authorization roles to a minimum.
- Create privileged system accounts for administrative functions and ensure they are allowed full access.
- Ensure that non-privileged accounts are allowed execute access to program files and read access to other objects.

B.2.4.6.1 UNIX TMR Servers and Managed Nodes Files and Permissions

When the TMF is installed on a UNIX platform, new files and directories are created, some existing files are modified, and several system variables are defined.

The assignment of directory and file access permissions for the TMF software is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts will be allowed execute and read access. The following table describes the UNIX file and directory permission requirements.

- *(TMF.0092: CAT II) The SA will ensure that access to the following directories, subdirectories, and all files in those subdirectories on UNIX TMR servers and managed node systems are restricted as follows:*

Object	Tivoli Account	Access Allowed	Appendix Reference
/var/spool/Tivoli	tmersrvd TMR Administrator Software Support Policy Region Administrator SA	Search \ Execute Update Update Execute Update	B.8

Object	Tivoli Account	Access Allowed	Appendix Reference
/var/spool/Tivoli/*host*.db	tmersrvd TMR Administrator Software Support Policy Region Administrator SA	Search \ Execute Update Update Execute Update	C.8
/usr/lib/X11/app-defaults	tmersrvd TMR Administrator Software Support Policy Region Administrator SA	Search \ Execute Update Update Execute Update	C.8
/usr/local/Tivoli/ **and subdirectories**	tmersrvd TMR Administrator Software Support Policy Region Administrator SA	Search \ Execute Update Update Execute Update	C.8
/usr/local/Tivoli/bin	tmersrvd TMR Administrator Software Support Policy Region Administrator SA	Search \ Execute Update Update Execute Update	C.8

B.2.4.7 Windows TMR Servers and Managed Nodes

This subsection is intended to provide guidance on security measures that should be observed when installing the TMF software on UNIX platforms. STIG issues and compliance will be the responsibility of the local SA and Security office.

In order to secure the TMF software and ensure the security of the TMF software the following must occur:

- Assign Group permissions to all TMF files and directories.
- Restrict access to Tivoli to authorized users.
- Limit the assignment of authorization roles to a minimum.
- Create privileged system accounts for administrative functions and ensure they are allowed full access.
- Ensure that non-privileged accounts are allowed execute access to program files and read access to other objects.

B.2.4.7.1 Windows TMR Servers, Managed Nodes Files and Permissions

When the TMF is installed on a Windows platform, new files and directories are created, some existing files are modified, and several system variables are defined.

The assignment of directory and file access permissions for the TMF software is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts will be allowed execute and read access. The following table describes the UNIX file and directory permission requirements.

- *(TMF.0093: CAT II) The SA will ensure that access to the following directories and subdirectories, and all files in those subdirectories on Windows TMR server and managed nodes are restricted as follows:*

Object	Tivoli Account	Access Allowed	Appendix Reference
\Tivoli\bin	tmersrvd TMR Administrator Software Support Policy Region Administrator SA	Search \ Execute Update Update Execute Update	C.8
\Tivoli\db	tmersrvd TMR Administrator Software Support Policy Region Administrator SA	Search \ Execute Update Update Execute Update	C.8
\Tivoli\lib	tmersrvd TMR Administrator Software Support Policy Region Administrator SA	Search \ Execute Update Update Execute Update	C.8
\Tivoli\include	tmersrvd TMR Administrator Software Support Policy Region Administrator SA	Search \ Execute Update Update Execute Update	C.8

Object	Tivoli Account	Access Allowed	Appendix Reference
\Tivoli\msg_cat	tmersrvd TMR Administrator Software Support Policy Region Adminstrator SA	Search \ Execute Update Update Execute Update	B.8

B.3 Tivoli Enterprise Console

B.3.1 General Overview

Enterprise management requires the ability to monitor resource activity and to resolve potential problems as they occur. In an enterprise, resources can be: networks, platforms, operating systems, databases and applications. In an enterprise where Tivoli products are used to manage the enterprise, the Tivoli Enterprise Console application TEC is used to provide performance monitoring and automated problem management. TEC enables Tivoli Enterprise Console Administrators to view, manage and administer enterprise resources and monitors. In addition, TEC enables TEC administrators to create, distribute and implement new monitors (adapters), as new resources are added to the enterprise.

In order to provide this functionality, TEC must obtain current information about the resources in the enterprise. TEC uses event data, which describes state changes of resources. The type of information, the level of detail and the format of the information that is provided by a resource depends on the event. Events can begin as error messages, traps, or similar information that is displayed or written to an error or log file.

When an event occurs on a managed resource, the TEC adapter captures information about the event, reformats pertinent event information and sends it to the event server. The event server receives the event information and processes the event. Events may be discarded, or acted upon automatically based on rules that have been preprogrammed into TEC. Whether or not an automated response is required, TEC displays the event information to the TEC console so that a TEC administrator can monitor activity and respond to an event manually.

TEC Administrator(s) are not only able to view and manage events, but also able to perform adapter management and administration. In addition TEC enables administrators to create, distribute and implement and tailor adapters for resources that may be added to the enterprise. These functions may be performed at a central location or remotely using the Desktop capability of TEC.

- *(TEC.0001: CAT II) The IAO will ensure that the TEC product is installed in the Tivoli Enterprise either using the TMF or authorized scripts.*

- *(TEC.0002: CAT II) The IAO will ensure that unauthorized copies of the TEC product, to include old release levels, are not installed in the Tivoli Enterprise.*

B.3.1.1 Components

The Tivoli Enterprise Console application is composed of nine components. Each component is designed to perform specific functions and work in conjunction with the TMF. The following subsections describe the components, their functionality and security considerations.

B.3.1.1.1 Event Adapters

An event adapter is a process (program) that resides on a host. It is used to monitor and collect information about events. Once an adapter collects event information, the information is reformatted into a format that is understandable by the TEC, and forwarded to the event server for further processing. Some types of resources that can be monitored by event adapters are: operating systems, databases, applications and networks.

Event adapters can be referred to as TME event adapters or non-TME event adapters. The difference is the way the adapters send information to the event server. TME adapters send event information to the event server using TMF services. There are two types of TME adapters: endpoint adapters and managed node adapters. At times it may be necessary to create non-TME adapters. The reason why non-TME adapters are used is because the adapters need to run on a system that is not supported by TMF. Non-TME event adapters use standard interprocess communication mechanisms to send event information to the event server. Non-TME adapters and managed node adapters send event information directly to the event server. Endpoint adapters send event information to the TEC gateway. The TEC gateway then forwards the event information to the event server.

Event adapters can be configured to discard selected events instead of forwarding the events to the event server. Thus, network traffic can be reduced and event server workload can be minimized. Both the TME and non-TME versions are developed the same way. Different libraries are linked to when creating the two different versions. The Tivoli Event Integration Facility EIF is used to create them and specialized event adapters as needed.

Some additional functions that can be performed by event adapters are: checking files at configurable intervals and polling system resources or system condition at predetermined intervals.

- *(TEC.0003: CAT II) The IAO will ensure that the creation, distribution and implementation of TME endpoint adapters, TME managed node adapters, and non-TME adapters are restricted to TMR administrators and authorized personnel.*

- *(TEC.0004: CAT II) The IAO will ensure that the adapter files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

B.3.1.1.2 Event Server

The TEC event server is responsible for the centralized processing of all events collected by the adapters in an enterprise. Only one instance of an event server may exist in a TMR. The event server is responsible for creating an entry in an RDBMS for each incoming event and after it creates an entry it evaluates the events against a set of rules to determine if it should respond to or modify the event automatically.

The event server consists of five daemons that run on the event server host. The following table describes the five daemons, their process name and functional description.

Daemon	Process Name	Function
Master process	tec_server process	Controls the activities of the other four processes.
Reception engine process	tec_reception process	Receives events from adapters, stores them in memory and logging them in the reception log of the database.
Rule engine process	tec_rule process	Defines the actions to be taken by the tec_dispatch process.
Dispatch engine	process tec_dispatch process	Performs the action(s) specified by the tec_rule process and writes information to the event repository portion of the database.
Task engine process	tec_task process	Executes tasks or programs as identified by the tec_dispatch process

The TEC event server must be installed on a managed node and only one TEC event server may exist in each TMR in an enterprise. If more that one TEC event server is needed in an enterprise, they can be set up in a hierarchical mode (this is accomplished by establishing the communications and the exchanging of data between each TMR). In an enterprise where each TMR supports its own TEC event server, one TEC event server may act as a backup for the other in case of network or TEC event server failures. The event adapters are able to automatically reroute their events to an available secondary TEC event server based on the adapter’s configuration file.

- *(TEC.0005: CAT II) The IAO will ensure that the Tivoli Enterprise Console event server is located in a secure location.*

- *(TEC.0006: CAT II) The IAO will ensure that the TEC event server daemons are restricted from unauthorized update and access.*

- *(TEC.0007: CAT II) The IAO will ensure that the TEC event server will use listening ports that are compliant with the Network STIG and DODI 8500.2.*

- *(TEC.0008: CAT II) The IAO will ensure that connection-oriented connections are used for all connections.*

B.3.1.1.3 Event Console

An event console is a graphical user interface (GUI) used by administrators and operators to monitor and respond to events in a Tivoli Enterprise. Many event consoles may exist in a TMR and the information displayed on each may vary. In order to control the events displayed event consoles, Administrators initially classify events into event groups and assign event groups to event consoles. It is possible that multiple event consoles can display the same event groups. Because event views can be shared, the UI server is responsible for preventing multiple event consoles from updating the same event and update event status on all event consoles. Thus, only one operator can respond to and resolve a problem.

There are two versions of event consoles: the Java version and the Web version. The Java version is used by administrators to perform configuration tasks, to start Tivoli NetView functions, to run local automated tasks and to monitor events. Operators use the Java version to start Tivoli NetView functions, to run local automated tasks, and to manage events. The Java version of the event console is the only version that can be used to perform administrative functions.

The Java version of the event console provides the following views:

Display	Purpose
Configuration view	Used to configure both the Java version and the Web version of the event console. Only administrators are given access to this view.
Summary Chart view	Used to show a high-level overview of the health of resources represented by an event group.
Priority view	Used to access priority event groups.

- *(TEC.0009: CAT II) The IAO will ensure that the Java event console is located in a restricted location in accordance with DODI 8500.2.*

The Web version is used by both administrators and operators to perform event monitoring. The Web console requires WebSphere Application Server, Version 5.0 Base Edition, and must be run on a dedicated WebSphere Application Server environment. By default, the Web console is not configured to operate in SSL mode. In order to use SSL for the Web console, it must be activated in the WebSphere Application Server configuration.

- *(TEC.0010: CAT II) The IAO will ensure that SSL is implemented under WebSphere to support TEC Web consoles.*

- *(TEC.0011: CAT II) The IAO will ensure that Web consoles will be located in secure locations.*

- *(TEC.0012: CAT II) The IAO will ensure that use of the Web consoles will be restricted to authorized users and appropriate roles are assigned in accordance with DODI 8500.2.*

The Web version of the event console organizes executable tasks into a portfolio, My Work. The following table describes the tasks contained in the MyWork portfolio.

Task	Description
Select an Event Group	Lists the event groups that have been assigned to the event console.
View Summary of Events	Shows a high level overview of the health of resources represented by an event group, indicating the number of events for each event severity in each event group and the total number of events for each event severity. You can also display the percentage of events for each event severity.
Run Tasks	Runs predefined tasks from task libraries.
Change User Preferences	Controls the display of events in the event viewer.

TEC is distributed initially with two pre-defined event consoles. They are the administrative console and the E Business Events Console. The administrative console, referred to as the AdministrativeConsole, is the default event console. It is assigned to the root administrator and enables the root administrator to immediately manage events in the predefined event groups without having to perform any additional configuration.

The E Business Events Console, referred to as the EbusinessEventsConsole, is used by operators to manage e-business events. The EbusinessEventsConsole has the following event groups assigned to it: EbusinessEvents, UnMonitoredEBusiness, and ProbableEventAssn).

- *(TEC.0013: CAT II) The IAO will ensure that access to the AdministrativeConsole is restricted to the TMR administrator and authorized personnel as specified in DODI 8500.2.*

- *(TEC.0014: CAT II) The IAO will ensure that access to the EbusinessEventsConsole is restricted to authorized users as specified in DODI 8500.2.*

- *(TEC.0015: CAT II) The IAO will ensure that local commands are restricted by role requirements.*

The event console also provides the ability to run automated tasks from the event console. Automated tasks are configured ahead of time and run when a particular event is received.

- *(TEC.0016: CAT II*) *The IAO will ensure that running tasks in the event console is restricted to the TMR administrator and authorized personnel as specified in DODI 8500.2.*

B.3.1.1.4 Tivoli Event Database

The Tivoli Event database is an external RDBMS used by TEC to store information about events. There are two main areas where event information is stored. They are the reception log, tec_t_evt_rec_log, and event repository, tec_t_task_rep. The reception log is used to track information about events that are received by the reception process. The event repository is used to store the results of the execution of tasks that are run as a result of receiving an event.

Early versions of TEC were limited to the types of databases supported, (DB2 and Oracle). Later releases of TEC, additional database types are supported. The following table describes the database types that are supported in the most current release.

Database	Version
DB2	Universal Database 7.2 (fix pack 7) WE, EE, EEE; 8.1 WSE, ESE
Oracle	8.1.7, 9i, 9i V2, 9i for Linux
Microsoft SQL Server	7.0 SP2, SP3; SQL Server 2000
Informix	9.3 Dynamic Server 2000
Sybase	11.9.2, 12.0, 12.5 (Adaptive Server Enterprise for Linux)

NOTE: TEC does not support a database that is on the native z/OS partition. The TEC database can be used on a z/OS system only if the supported database is running on a Linux z/OS partition.

The TEC event database is accessed through the RDBMS Interface Module (RIM) component of the TMF. Whenever TEC needs to communicate with the event database, it requests RIM to start a RIM agent process, which runs on the RIM host. The RIM host uses a RIM object to obtain such information as the database name and database user ID. Then the RIM agent is able to use the RDBMS client libraries to communicate with the database and issue SQL statements. The RIM interface is located, by default, on the TEC server. When the RIM host does not reside on the TEC server, it can be installed on any Managed Node in a Tivoli Management Region.

RIM not only provides TEC with a common set of APIs but the set is also usable by other applications getting and storing data. Whenever data is submitted through the RIM APIs, it is converted to the format of the different database types. Because RIM provides the database calls, there is no need for specific SQL for each database.

RIM is controlled by the TMF roles: *rim_view* and *rim_ update*. Anyone with the appropriate *rim_** access role can access the database without having to be defined to the database and granted access to database resources.

- *(TEC.0017: CAT II) The IAO will ensure that the RIM access roles are restricted to authorized users and processes.*

- *(TEC.0018: CAT II) The IAO will ensure that access to the TEC database is restricted in accordance with the Database STIG.*

B.3.1.1.5 Tivoli Enterprise Console Gateway

The TEC gateway was designed to receive events from TME adapters and non-TME adapters, filter them and pass them to the event server for processing. When the TEC gateway is started it establishes a connection-oriented service connection to the event server and continues to use it for passing all events to the event server.

The TEC gateway is comprised of two programs, tec_gateway and tec_gwr. The tec_gateway program processes events from TME adapters. The tec_gwr program, which is an endpoint adapter, receives events from non-TME adapters and sends them to the tec_gateway program. In order for the TEC gateway to receive events from non-TME adapters, the TEC gateway configuration file must be turned on. Once it is turned on, tec_gwr uses the LCF transport type to send events to the tec_gateway program. The Tivoli Enterprise Console gateway forwards events to the event server and communicates with the event server using the Tivoli Event Integration Facility.

The tec_gateway program and its adapter files for each endpoint operating system are installed as part of the Adapter Configuration Facility installation process on a managed node. The Adapter Configuration Facility must be installed on any managed node that is configured as a Tivoli Enterprise Console gateway. The Tivoli Enterprise Console gateway forwards events to the event server and communicates with the event server using the Tivoli Event Integration Facility.

The configuration file for the TEC gateway is optional and does not exist on the managed node until an adapter configuration profile containing the gateway configuration information is distributed to the endpoint on that managed node. The TEC gateway uses default values unless modified and an adapter configuration profile containing the changed gateway configuration information is distributed. Whenever a value is not specified in the configuration file, the TEC gateway assumes the default specification.

- *(TEC.0019: CAT II) The IAO will ensure that TEC gateway is located in a restricted location.*

- *(TEC.0020: CAT II) The IAO will ensure that communication between the TEC gateway and the event server is encrypted.*

- *(TEC.0021: CAT II) The IAO will ensure that that communication between the TEC gateway and event adapters is encrypted.*

- *(TEC.0022: CAT II) The IAO will ensure that the configuration file for the TEC gateway is restricted from unauthorized updates and access.*

B.3.1.1.6 User Interface Server (UI)

The user interface (UI) server, tec_ui_server, is responsible for providing communication services between the event consoles and the event server and the event database. The UI server is not part of the event server and as a result can be installed on any managed node in a TMR. Only one instance of a UI server may exist in a TMR. Whenever a connection to the event server is made, the UI server uses the dispatch engine process of the event server.

A major function of the UI server is to provide transaction locking for event console status updates. This function prevents multiple event consoles from responding to the same event. In addition, the UI server automatically updates the status of events on all event consoles by forwarding the event changes from the event consoles to the dispatch engine, which sends the changes to the event database.

The UI server provides a set of commands that enable operators to change any event attribute, list events in a specific event group, and display a message on the operator's Desktop. If the UI server encounters a failure, all error messages are written to the UI server log file.

- *(TEC.0023: CAT II) The IAO will ensure that UI server is located on a managed node, which is located in a restricted location.*

- *(TEC.0024: CAT II) The IAO will ensure that access and updates to the UI server files are restricted to authorized personnel.*

- *(TEC.0025: CAT II) The IAO will ensure that all connections between the UI server and the event server are encrypted.*

- *(TEC.0026: CAT II) The IAO will ensure that authorized roles are restricted in accordance with DODI 8500.2.*

B.3.1.1.7 Adapter Configuration Facility

Adapters are generic processes that collect event information about the resources. They use configuration files and configuration profiles to provide specific details about resources, event types, methods for collecting data, the format of the event information and the final disposition of the event information once it has been collected.

The Adapter Configuration Facility is a profile-based application that enables administrators to configure and distribute TME adapters using a GUI. Administrators can use the Adapter Configuration Facility to create and manipulate profiles for adapters and set configuration and distribution options. The adapters can then be distributed to the profile's subscribers using the menu options of the gui or by dragging and dropping the profiles to the appropriate profile manager.

- *(TEC.0027: CAT II) The IAO will ensure that access to the ACF is restricted to the TMR Administrator and authorized personnel in accordance with DODI 8500.2.*

B.3.1.1.8 Tivoli Event Integration Facility EIF

The Tivoli Event Integration Facility is a toolkit that enables administrators to develop new adapters and tailor existing adapters in order to meet specific needs of a network. It can run on a TMA endpoint. EIF can be used to create event listeners, which are applications that receive events. Once created, they can be placed in the managed environment where a need exists to distribute events to other management applications. EIF supports the Java and the C development environments for creating adapters. The APIs for both are functionally the same and support communication with the event server.

Additionally, EIF can be used to filter events near their source. This reduces event traffic on the network and the event server. It uses filtering with configuration files and state correlation to analyze, summarize, and distribute the incoming event information. When properly defined, configuration files and state correlation optimize event management by minimizing the number of events that each operator must monitor.

- *(TEC.0028: CAT II) The IAO will ensure that access to the EIF is restricted to TMR administrators or authorized personnel in accordance with DODI 8500.2.*

- *(TEC.0029: CAT II) The IAO will ensure that creation and tailoring of adapters using the EIF is restricted to TMR administrators or authorized personnel in accordance with DODI 8500.2.*

B.3.1.1.9 Tivoli NetView

The Tivoli NetView product is designed to provide the network management support in a Tivoli Enterprise. It is used to control network configurations, identify and resolve problems, and execute performance management functions for network resources. In addition, Tivoli NetView is capable of monitoring the status of network devices and automatically filtering and forwarding network-related events to TEC for display and potential action.

Tivoli NetView uses foreground and background processes to perform this type of activity. The foreground processes or applications can only be invoked while the Tivoli NetView graphical interface is running. Whereas the background processes or daemons run continuously regardless of whether the graphical interface is running or not. These processes can be started only by the root user or the root shell and stopped only by the root user. Normally, the daemons provide services that must be available at all times.

The Tivoli NetView product provides a default set of significant events that are forwarded to the Tivoli Enterprise Console including status events, selected SNMP data collection threshold events, and Router Fault Isolation events. In order for TEC to correlate the NetView event information, it uses the netview.rls rule set.

Tivoli NetView is able to communicate with user-configured IBM Tivoli Monitoring servers for determining lists of endpoints that are being monitored and the applications that are running on each endpoint. NetView TEC adapters are able to forward service-related events for ITM nodes to TEC for correlation and potential response. Information obtained from these adapters is also stored in the object database for use in determining the service impacts to network faults.

- *(TEC.0030: CAT II) The IAO will ensure that access to Tivoli NetView is restricted to authorized personnel in accordance with DODI 8500.2.*

- *(TEC.0031: CAT II) The IAO will ensure that Tivoli NetView is installed and maintained by authorized system programmer personnel.*

- *(TEC.0032: CAT II) The IAO will ensure that Tivoli NetView is installed either using TMF facilities or authorized scripts and processes in accordance with DODI 8500.2.*

B.3.1.1.9.1 NetView Server

The Tivoli NetView server program is responsible for performing network management functions. The NetView server uses SNMP to discover, monitor, and configure TCP/IP networks. In a Tivoli environment, the NetView server can only be installed on AIX and Solaris managed nodes. In a non-Tivoli environment the NetView server program can be installed and run on Solaris, Linux, zLinux, and AIX and Windows operating systems.

The NetView server is able to communicate with the event server using either Tivoli-based communication or non-Tivoli-based (socket-based) communication. For Tivoli-based communication to occur, an endpoint must be installed on the computer where the NetView server is installed and a TEC gateway must exist between the NetView server and the event server. Even though there can be multiple endpoints on a single computer that are connected to different gateways, the NetView server forwards events to only one event server at a time. In addition, the NetView server requires that an SNMP agent be installed. The NetView server is manually configured in a Tivoli environment.

- *(TEC.0033: CAT II) The IAO will ensure that access to Tivoli NetView Server is restricted to authorized personnel in accordance with DODI 8500.2.*

- *(TEC.0034: CAT II) The IAO will ensure that access to Tivoli NetView Server is located on a platform with restricted physical access in accordance with DODI 8500.2.*

- *(TEC.0035: CAT II) The IAO will ensure that communication between Tivoli NetView Server and the TEC event server is encrypted.*

B.3.1.1.9.2 NetView Consoles

Tivoli NetView provides for two types of consoles. They are the NetView Web console and the NetView Native Console. The Tivoli NetView Web console is a Java-based graphical user interface (GUI) that enables operators to view network topology, and obtain diagnostic informational for performing network troubleshooting and problem resolution. The NetView Web console is installed automatically when the NetView server is installed. For the NetView Web console to work, it must be installed on the same computer as the event console (Java version), which can only be a non-Tivoli environment.

The Tivoli NetView native console is an X/Motif-based (UNIX) or MFC-based (Windows) graphical user interface (GUI) that enables administrators to configure the NetView server. It also provides the same operator functionality as the Tivoli NetView Web console. It is also installed automatically when the Tivoli NetView server is installed. The native NetView console supports customization of the menu structure using Application Registration Files (ARF). The ARF files enable the addition of menu items to the menu bar.

- *(TEC.0036: CAT II) The IAO will ensure that access to Tivoli NetView consoles are restricted to authorized personnel in accordance with DODI 8500.2.*

- *(TEC.0037: CAT II) The IAO will ensure that locations of the Tivoli NetView consoles are secured in accordance with DODI 8500.2.*

- *(TEC.0038: CAT II) The IAO will ensure that communication between the Tivoli NetView consoles and the servers are encrypted.*

B.3.1.2 General Considerations

The following subsections address areas, which impact the performance of the TEC product.

B.3.1.2.1 Event Classes

Event classes are designed to serve as an agreement between adapters and the event server. They provide the guidelines as to the type of information adapters send to the event server for a given event. Event class definitions define each possible event type that can be received at the event server. Each definition must have a unique name.

Event classes are organized in a hierarchy with the top of the hierarchy being the base event class named EVENT. Event classes can be further sub-divided into subclasses that provide for a more detailed set of rules for breaking down event information.

Event classes are defined in event class definition files. Each definition must have a unique name. The base event class definition file *root.baroc* is located in the *$BINDIR/TME/TEC/default_rb/TEC_CLASSES* directory. Other event classes are subclasses of the base event class.

When a new adapter is created, the types of events that the adapter can send to the event server is defined in a BAROC file and loaded on the event server. Multiple BAROC files can exist on an event server.

- *(TEC.0039: CAT II) The IAO will ensure that creation of event classes is restricted to TMR Administrator or authorized personnel as specified in DODI 8500.2.*

- *(TEC.0040: CAT II) The IAO will ensure that the distribution of adapters and event class files are restricted to the TMR Administrator, and authorized personnel.*

- *(TEC.0041: CAT II) The IAO will ensure that the root.baroc access permissions are restricted in accordance with Appendix B.8, the UNIX and Windows STIGs, and DODI 8500.2.*

B.3.1.2.2 Rules

A Tivoli Enterprise Console rule is a construct that specifies the type of action to be performed when a certain event is received. Rules are written in a high-level language called the rule language. The rule language provides a simplified interface to the Prolog programming language, which is the language actually used internally by the rule engine. Rules, which are in the rule language, are precompiled into Prolog source code, and then compiled into Prolog executable files.

As part of the rule language, a set of predefined predicates is provided by Tivoli. These predicates are frequently used actions in rules. A rule executes when the event under analysis has satisfied all of the conditions specified in the rule’s event filter. An event filter can contain tests for an event class name and event attribute conditions.

The following table describes the five types of rules that may exist.

Rule	Description
Plain rule	Used with incoming new events, or with previously received events to be re analyzed. Plain rules allow the flexibility to use any predicate or Prolog feature in its actions.
Change rule	Used with previously received events that have a request to change their information. A request to change an event's information is called a change request. For change requests, the change rules are checked before the change is actually made. This timing allows for the creation of rules to take action depending on the old value of an attribute, the new value of the attribute, and the origin of the change request. Change rules provide the flexibility to use any predicate or Prolog feature in its actions. Change rules can only specify plain actions. Redo actions and reception actions are considered errors when they are specified in change rules.
Timer rule	Used when a previously set timer on an event expires. Timers can be set on an event with the set_timer predicate in a rule. Timer rules, provide the flexibility to use any predicate or Prolog feature in its actions. Redo actions and reception actions are considered errors when they are specified in timer rules.
Simple rule	Used with incoming new events, or with a redo request. A simple rule is not as flexible as a plain rule, in that it contains predefined conditions and actions, and cannot have or use a predicate or any Prolog feature in its actions. A simple rule does not do any correlation with other events in the event cache, except for dropping duplicate events.
Correlation rule	Used with incoming new events, or with a redo request. A correlation rule enables a causal relationship between two event classes to be established. One event either causes the other to be generated, or one event causes the other to be closed. With a correlation rule, the value of the status attribute from a cause event to an effect event can be propagated. Correlation rules are called compound rules in the rule builder dialogs.

- *(TEC.0042: CAT II) The IAO will ensure that rule creation will be restricted to the TMR administrator or authorized personnel in accordance with DODI 8500.2.*

B.3.1.2.2.1 Rule Base Targets

The TEC event server is the master container from which rule base targets are created. Rule base targets are the actual rule bases used by rule engines to process events. When there is more than one rule engine managing events in the environment, the rule bases used by the rule engines in that environment are referred to as distributed rule bases.

In a distributed rule base environment, event classes and rules must be synchronized among all the rule engines. In order to keep these synchronized, all rule base development must be done with the TEC event server, which is the centralized point of control for managing a distributed rule base environment.

Rule base targets, after compilation of the rule base on the TEC event server, are located in the rule_base_directory/.rbtargets/target_name directories (note the leading period in the .rbtargets subdirectory name). The name for the rule base target used by the rule engine on TEC event server is EventServer. The EventServer rule base target is automatically created in every rule base. Distributed event servers only need the rule base target directory structure starting from the target_name subdirectory for their use.

The event classes and predicates in a rule base target are the same throughout a distributed environment, and are replicated from the rule base on the TEC event server to each rule base target during compilation. The rule sets in rule base targets can differ, depending on design and implementation of distributed rule bases in your environment. Because of this, the rule sets which are to be included with each rule base target must be specified.

Rule bases needed by other event servers in the distributed environment (such as Tivoli Availability Intermediate Manager event servers) are obtained as rule base targets created by the TEC event server. Because all rule base targets for a rule base use the same set of classes, all rule builder and wrb commands manipulate BAROC files at the rule base level on the IBM Tivoli Enterprise Console event server. When the rule base is compiled, the event classes are replicated to the rule base targets defined in the rule base.

- *(TEC.0043: CAT II) The IAO will ensure that rule base creation will be restricted to the TMR administrator or authorized personnel in accordance with DODI 8500.2.*

- *(TEC.0044: CAT II) The IAO will ensure that rule base distribution is conducted by the TMR administrator or authorized personnel in accordance with DODI 8500.2.*

- *(TEC.0045: CAT II) The IAO will ensure that rule base targets are restricted from unauthorized access and updates in accordance with DODI 8500.2.*

- *(TEC.0046: CAT II) The IAO will ensure that rule base target directory permissions are set in accordance with Appendix B.8, the Windows and UNIX STIGs, and DODI 8500.2.*

B.3.1.2.2.2 Rule Sets and Rule Packs

Rule sets are the files that contain rules. Related rules are generally contained within a rule set. When a rule base is compiled, rule sets are replicated to those rule base targets that have specified which rule sets to import. When a rule engine is using a rule base, generally the rules are processed in the order defined within a rule set and within the order of how the rule sets were imported into the rule base target. The regular rule processing order can be altered with the use of certain predicates called from within rules.

It should be noted that the order of rule sets defined for a rule base target is important, because it affects rule engine performance.

Placement of rule sets determines evaluation order by the rule engine. A default set of rule sets is provided by Tivoli with the Default rule base. A default rule set for the Tivoli Availability Intermediate Manager is also included with the default rule base. Another way to import rule sets into a rule base target are with rule packs. Rule packs are a convenient way to package a group of rule sets so they can be imported into a rule base target in a single operation. Rule packs are used to combine a group of rule sets that are used in multiple rule base targets. When a rule base is compiled, those rule base targets that are defined to receive rule packs receive their rule pack contents, which are rule sets. Before rule sets and rule packs can be imported into rule base targets, they must first be imported into the rule base on the TEC event server.

- *(TEC.0047: CAT II) The IAO will ensure that rule sets will be restricted from unauthorized access and updates in accordance with DODI 8500.2.*

- *(TEC.0048: CAT II) The IAO will ensure that rule packs will be restricted from unauthorized access and updates in accordance with DODI 8500.2.*

- *(TEC.0049: CAT II) The IAO will ensure that rule sets and rule targets are restricted in accordance with Appendix B.8, the Windows and UNIX STIGs, and DODI 8500.2.*

B.3.1.2.3 WebSphere Application Server

The TEC console Version 3.9 Web console component requires WebSphere Application Server WAS Version 5.0 Base Edition. This product is required for the Web console and can be installed during the installation of the Web console. Other editions of the WebSphere Application Server Version 5.0 are not supported. Securing the WebSphere Application Server have be specified in the platform STIG that the WAS server is to be loaded on.

- *(TEC.0050: CAT II) The IAO will ensure that the WAS server is secured with respect to the platform STIG requirements and DODI 8500.2.*

B.3.1.2.4 Secondary Event Servers

One or more secondary event servers can be specified for an event adapter. A secondary event server is a backup event server that receives events when the TEC gateway cannot contact the adapter-specified event server. Secondary event servers are specified in the TEC gateway configuration file.

- *(TEC.0051: CAT II) The IAO will ensure that the secondary event servers are secured in accordance with the above event server section, the platform STIG requirements, and DODI 8500.2.*

B.3.1.2.5 Platforms Supported

The TEC server and console products must be installed on managed nodes. The following table describes the supported operating systems, release levels and Tivoli platforms.

Operating System	Version	Event Server	NetView Server	Adapter Configuration Facility	UI Server	Event console (Java version)	Endpoint
IBM AIX	4.3.3, 5.1, 5L, 5.2	X	X	X	X	X	X
Sun Solaris Operating Environment (Solaris)	8, 9	X	X	X	X	X	X
HP-UX (PA-RISC)	11.0 spl, 11i	X		X	X	X	X
Windows	XP Pro, 2000 Pro					X	X
Windows	Windows 2000 Server	X	X	X	X	X	X
Windows	2000 Datacenter						X
Windows	2003 Server	X	X	X	X	X	X
Windows 2003	2003 Advanced Server	X	X	X	X	X	X
IBM zOS	V1R2, V1R3, V1R4						X
SuSE Linux Enterprise Server for IA32	7 (7.2 base)	X	X	X	X	X	X
SuSE Linux Enterprise Server for iSeries	7 (7.2 base)						X
SuSE Linux Enterprise Server for zSeries	7 (7.2 base)	X	X	X	X		X
SuSE Linux	7						X

Operating System	Version	Event Server	NetView Server	Adapter Configuration Facility	UI Server	Event console (Java version)	Endpoint
Enterprise Server for pSeries	(7.2 base)						
United Linux for IA32 (SLES 8)	1.0	X	X	X	X	X	X
United Linux for iSeries (SLES 8)	1.0						X
United Linux for PSeries	1.0						X
United Linux for zSeries (SLES 8)	1.0	X	X	X	X		X
Redhat Linux for zSeries	7.2	X	X	X	X		X
Redhat Linux for iSeries	7.2						X
Redhat Linux for PSeries	7.2						X
Redhat Linux Advanced Server 2.1 for IA32	2.1	X	X	X	X	X	X
NetWare	5.1, 6.0						X
OS/400®	V5R1, V5R2						X
OS/2 Warp	4.5						X
OS/2 Server for eBus	4.5.1						X

Operating System	Version	Event Server	NetView Server	Adapter Configuration Facility	UI Server	Event console (Java version)	Endpoint
Compaq Tru64	5.1, 6.0						X
Reliant UNIX	5.4.5						X
SCO Unixware	7.1.1, Open, UNIX 8						X
Irix SGI	6.5.*x*						X
Solaris ix86	7,8						X
Sequent Dynix/PTX	4.6.1						X

- *(TEC.0052: CAT II) The IAO will ensure that the TEC software is loaded on platforms supporting the appropriate release levels.*

B.3.1.2.6 Databases

The TEC software supports the multiple database types for the event database. The following table describes the databases and versions supported.

Database	Version
DB2 Universal Database	7.2 (fix pack 7) WE, EE, EEE; 8.1 WSE, ESE
Oracle	8.1.7, 9i, 9i V2, 9i for Linux
Microsoft SQL Server	7.0 SP2, SP3; SQL Server 2000
Informix	® 9.3 Dynamic Server 2000
Sybase	11.9.2, 12.0, 12.5 (Adaptive Server Enterprise for Linux)

- *(TEC.0053: CAT II) The IAO will ensure that unsupported releases of the above databases are not used as the event server.*

B.3.1.2.7 Files

There are many files that may be installed to support TEC and its components. The following subsections identify the files and provide a description of usage.

B.3.1.2.7.1 TEC Base Product Files

The TEC server software is installed in a directory structure that is subordinate to a logical directory known as the BINDIR directory. The physical implementation of the BINDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. Some of the TEC server software files are installed in directories that are shared with the Management Framework. These directories are protected through the specific Management Framework requirements elsewhere in this document. Requirements for the TEC-specific directories are addressed here. TEC may be installed using the TMF or by using a script. Tivoli files for Windows NT are, by default, stored under the \Tivoli directory on the root of the selected drive. Tivoli will also write install and other log files to %DBDIR%\tmp.

- *(TEC.0054: CAT II) The IAO will ensure that the secondary event servers are secured in accordance with the above event server section, the platform STIG requirements, and DODI 8500.2.*

B.3.1.2.7.2 Adapter Files

The following table describes the files that are related to adapters. These files are in addition to the header file or Java package files.

File	Extension	Description
Configuration	.conf	The configuration file controls filtering and buffering of events, and also communications. This file is located with the adapter and TEC gateway. This file is optional.
Basic recorder of objects in C (BAROC)	.baroc	The Basic Recorder of Objects in C (BAROC) file validates if events are in a valid format based on their event class definitions. This file is located on the event server. Each adapter comes with a BAROC file describing the classes of events the adapter supports. This file is not used by the adapter itself, but serves as a mandatory link between the adapter and the event server. The event server must load this file before it is able to understand events received from the adapter.
XML	.xml	The XML file defines the state machines for state correlation. This file is located on the adapter, but also can be present on the TEC gateway. This file is optional.
Class definition statement (CDS)	.cds	A CDS file is used by an adapter to map incoming raw events to a particular class and to define event attributes before forwarding the events to the event server. If any event definition is changed in a CDS file, the corresponding event class definition in the BAROC file might need changing as well.

File	Extension	Description
Rules	.rls	The rule file applies custom rules to events for filtering, tasks, and other actions. Some rule files are installed on the event server by default. Other rules can be optionally specified. Some adapters come with a rule file describing the classes of events the adapter supports. This file is not used by the adapter itself, but serves as a mandatory link between the adapter and the event server. The event server must load this file before it is able to understand events received from the adapter.
Format	.fmt	This file defines the format of messages and matches them to event classes for the UNIX logfile, NetWare logfile, OS/2, and Windows event log adapters. No alterations to this file are necessary to use an adapter.
Error	.err	Log file used to log adapter errors.

- *(TEC.0055: CAT II) The IAO will ensure that the adapter files permissions are set in accordance with Appendix B.8, the Windows and UNIX STIGs, and DODI 8500.2.*

B.3.1.2.7.3 ACF Files

The TEC ACF may be installed using the TMF or by using a script. When the TEC ACF files are installed, they should be installed under the TEC directory.

- *(TEC.0056: CAT II) The IAO will ensure that the ACF directory file permissions are set in accordance with Appendix B.8, the Windows and UNIX STIGs, and DODI 8500.2.*

B.3.1.2.7.4 EIF Files

The TEC EIF may be installed using the TMF or by using a script. When the TEC EIF is installed, the following directory structure is created to support TEC:

- *(TEC.0057: CAT II) The IAO will ensure that the EIF directory file permissions are set in accordance with Appendix B.8, the Windows and UNIX STIGs, and DODI 8500.2.*

The directory structures for EIF are as follows:

File	Contents
bin	Contains the CLI executables postemsg, wpostemsg, postzmsg, and wpostzmsg for each interp type.
contrib.	Contains compiled sample programs.
default_sm	Contains samples for using State Based Correlation.

File	Contents
Include	Contains header files (.h) for building adapters. These are common across all interps.
Jars	Contains all the jar files necessary to use the Java based Event Integration Facility API and State Based Correlation.
Javadoc	Contains the javadoc for the Java based Event Integration Facility API.
Lib	Contains, for each interp type, the libraries needed for linking an adapter. Includes libraries for endpoint, managed node and non-TME adapters.
Samples	Contains sample adapter source code.

- *(TEC.0058: CAT II) The IAO will ensure that the EIF file permissions are set in accordance with Appendix B.8, the Windows and UNIX STIGs, and DODI 8500.2.*

B.3.1.2.7.5 Class Definition Statement (CDS) Files

A class definition statement file is used by adapters to determine what to do with events. The CDS file consists of a list of SELECT, FETCH, and MAP statements for all event classes supported by adapters that utilize it. The CDS file is required for most adapters and follows the same format for all adapters that use it. A CDS file has an extension of .cds.

The class definition statements that are contained in a CDS file are evaluated in the order they appear in it. An incoming event is mapped to the class specified by the first class definition statement whose SELECT statement is evaluated successfully. When more than one class definition statement is provided for a particular class of event, the class definition statement with the most restrictive SELECT statement is placed before the less restrictive statements in the CDS file. Locating the most restrictive class definition statement first for a same-named class helps provide for better performance of adapters.

- *(TEC.0059: CAT II) The IAO will ensure that the CDS file permissions are set in accordance with Appendix B.8, the Windows and UNIX STIGs, and DODI 8500.2.*

B.3.1.2.7.6 TEC Gateway Configuration File

The following table describes the location of the configuration file by platform.

Platform	Path
UNIX	/etc/Tivoli/tec/tec_gateway.conf
Windows	%SystemRoot%\drivers\etc\Tivoli\tec\tec_gateway.conf

- *(TEC.0060: CAT II) The IAO will ensure that the TEC Gateway Configuration file permissions are set in accordance with Appendix B.8, the Windows and UNIX STIGs, and DODI 8500.2.*

B.3.2 General Security Considerations

B.3.2.1 Overview

The unique considerations for TEC include:

- Access to TEC functions is controlled through the assignment of Tivoli authorization roles to Tivoli users. Without proper assignment of these roles, the confidentiality and integrity of the TEC data could be compromised.

- Because TEC requires the use of an RDBMS for data storage, the secure configuration of the client access software and the RDBMS are significant.

- As with most software products, TEC is composed of program and data files for which proper access controls are essential. Although most of the TEC files will reside in directories subject to access controls required for the Framework, there are some TEC directories and files that require specific access controls.

- If the TEC web interface features are used, administrative or user transactions are processed through HTML pages. The integrity of the HTML files must be assured to maintain transaction security.

- There are two considerations relative to the security of the TEC event server:

    - The host OS must be configured securely.
    - The integrity of the TEC event server software must be maintained.

  There are three considerations relative to the security of the RIM host:

    - The host OS must be configured securely.
    - If required on the platform, the OS user account utilized by TEC functions must be defined properly.
    - The integrity of the RDBMS client software must be maintained.

  There are four considerations relative to the security of the RDBMS server:

    - The host OS must be configured securely.
    - The RDBMS software must be configured securely.
    - If required on the platform, the OS user account utilized by TEC functions must be defined properly.
    - The RDBMS user account utilized by TEC functions must be defined properly.

- There are two considerations relative to the security of the TEC Gateway:

  - The host OS must be configured securely.
  - The integrity of the TEC Gateway software must be maintained.

- There are four considerations relative to the security of the TEC-specific web interfaces:

  - The integrity of the WebSphere server-based web components used in the Software Signatures Editor, the User Data Template update, and the WebSphere facility must be maintained.
  - The proper assignment of the Tivoli authorization role for the link functions must be managed.
  - The confidentiality of the user authentication data that passes over the network connection between the endpoint or managed node client and the TMR server must be maintained.
  - The integrity of the user machine-based WebSphere web page must be maintained.

The host OS security consideration applies generically to the host. It is addressed through the platform security requirements described in the applicable platform STIG. Specifically, the RIM host must be compliant with the STIG that covers the OS used on the RIM host.

The security consideration for the confidentiality of the user authentication data applies to the transmission of the Tivoli user password from the web browser on the endpoint or managed node client to the TMR server. The HTTP protocol between the client and server does not provide encryption. This makes the password vulnerable to capture during transmission over the network.

Password data is encrypted in accordance with the *DODI 8500.2, IA Controls for Individual Identification and Authentication*. Passwords must be encrypted both for storage and for transmission. The actions that are required to address these considerations are addressed in the following section of this document.

B.3.3 Specific Security Considerations

This section describes the specific considerations and required actions to help ensure that the TEC components are implemented in a secure fashion. As noted in the previous section, compliance with the requirements in the Framework section of this document is assumed as the prerequisite for the information here.

Department of Defense (DOD) Instruction 8500.2, Information Assurance (IA) Implementation, documents IA controls that apply to DOD information systems. The following integrity and confidentiality controls from *DODI 8500.2* are the basis for the requirements in this section: Access for Need-to-Know; Audit Trail, Monitoring, Analysis, and Reporting; Changes to Data; Individual Identification and Authentication; Least Privilege; and Privileged Account Control.

The following subsections describe the considerations and requirements of TEC components. When reviewing these requirements, the following must be noted:

- The specific directory names in the requirements reflect defaults indicated in vendor documentation. If a site or organization deploying Tivoli chooses other names, the requirements apply to the site-specific names.

- The structure of UNIX permissions can make it difficult to restrict file access to multiple groups of users. The approach taken in this section is to allow UNIX "world" access in cases where non-update (e.g., read or execute) access permission is required for users who may be members of multiple groups and there is no specific need for the files to be inaccessible to other users.

B.3.3.1 Authorization Roles

Authorization roles defined within the Management Framework provide role based access control over functions in the Tivoli products. A Tivoli administrator must be assigned the associated roles to be permitted to perform specific product functions.

Authorization roles such as senior or super enable a broad range of capabilities in multiple products. The following tables describe the authorization roles required for TEC and used under the different components.

Required Tivoli authorization roles:

Activity	Context	Tivoli Authorization Role
Assigning administrative roles for the event server	Event server	senior
Configuring		super
Starting and stopping		senior
Assigning an operator to an event console.	Event console	senior
Configuring your own event viewer preferences		user
Assigning event group roles	Event groups. senior	senior
Creating modifying or deleting event groups		senior
Assigning event groups		senior
Deleting events	Events	senior
Acknowledging and closing events		admin, RIM_view, and RIM_update
Viewing events		user and RIM_view
Sending events		user, if using the wpostemsg command, otherwise none

Activity	Context	Tivoli Authorization Role
Creating, modifying, or loading rule bases	Rule base	Senior
Installing the Tivoli Enterprise	Console product	Tivoli region super

Special Considerations:

- Each operator must be defined as a Tivoli administrator with the appropriate authorization roles to manage events before assigning the operator to an event console.

- If an authorization role is added to a Tivoli administrator after the operator starts the event console, the authorization role does not take affect until the event console is restarted.

The following table describes the Tivoli authorization roles and associated ACF functions:

Role	Function
ACF_glopol	Used by an administrator to set the global adapter policy.
ACF_polmod	Used by an administrator to edit profile policy and create new profiles.
ACF_rwdist	Used by an administrator to edit and distribute adapter configuration profiles.
ACF_readonly	Used by an administrator to view adapter configuration profiles, but the administrator cannot create, edit, or distribute the adapter configuration profiles.

NOTE: These TMR and resource roles for an administrator can be created from the TMF Administrators dialog boxes.

The following table describes the roles required to perform adapter configuration file maintenance.

Activity	Context	Authorization Role
Set policy region managed resources for an adapter configuration profile	Policy region	senior
Create an adapter configuration profile	Profile manager	senior

It should also be noted that to install components in a Tivoli environment, requires a Tivoli root Administrator with all available roles, and in order to install the NetView server in a non-Tivoli environment, requires a user with root authority.

The assignment of Tivoli authorization roles is restricted and tracked in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know, Least Privilege, and Privileged Account Control*. Roles must be assigned only to users who require the associated privileges to perform designated job functions.

B.3.3.2 File Permissions

The assignment of directory and file access permissions for the TEC software is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as the accounts used by TEC users, are allowed execute and read access. Appendix B.8 provides the permissions and privileges for the Tivoli Enterprise Console Server files, Tivoli Enterprise Console User Interface Server files, Tivoli Enterprise Console Java Console files, Tivoli Enterprise Console Sample Event Information files, and ACF files.

- *(TEC.0061: CAT II) The IAO will ensure that access to the following subdirectories of the BINDIR directory, their subdirectories, and all files in those subdirectories on UNIX Gateway systems is restricted in accordance with Appendix B.8, the UNIX STIG, and DODI 8500.2.*

- *(TEC.0062: CAT II) The IAO will ensure that access to the subdirectories of the BINDIR directory, their subdirectories, and all files in those subdirectories on Windows Gateway systems is restricted in accordance with Appendix B.8, the Windows STIG, and DODI 8500.2.*

B.3.3.3 TEC Event Server Data Base (Oracle)

The Oracle RDBMS client software is installed on a RIM host in a directory structure that is subordinate to a logical directory known as the ORACLE_HOME directory. The physical implementation of the ORACLE_HOME directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices.

- *(TEC.0063: CAT II) The SA will ensure that access to the ORACLE_HOME directory, its subdirectories, and all files in those directories on UNIX RIM Host systems is restricted in accordance with Appendix B.8, the UNIX and Windows STIGs, and DODI 8500.2.*

- *(TEC.0064: CAT II) The SA will ensure that access to the ORACLE_HOME directory, its subdirectories, and all files in those directories on Windows RIM Host systems is restricted in accordance with Appendix B.8, the Windows STIG, and DODI 8500.2.*

On MS Windows systems, the Windows registry contains parameters used for the execution of the Oracle RDBMS client software. If these parameters were altered incorrectly or with malicious intent, the client software might become unusable or might store or return invalid data. Therefore the registry entries must be protected from unauthorized update.

- *(TEC.0065: CAT II) The SA will ensure that access to the following registry entries for the Oracle client software on Windows RIM Host systems is restricted in accordance with Appendix B.8, the Windows STIG, and DODI 8500.2.*

Object	Appendix
HKLM\SOFTWARE\ORACLE *(include all subkeys)*	*B.8*

- *(TEC.0066: CAT II) The Tivoli administrator will ensure that the password for the TEC account defined to the RDBMS is changed after installation and conforms to the password complexity requirements.*

B.3.3.4 Adapter File Locations

By default, adapters expect their configuration, format, CDS, and error file to be in the following locations. The following table describes the adapter types, nodes and path locations.

Adapter Type	Node Type	Location
TME	Managed node	$BINDIR/TME/TEC/adapters/etc/ or /etc/Tivoli/tecad/etc (which is a link to the TME adapter directory)
TME	Endpoint	$LCFROOT/bin/$INTERP/TME/TEC/adapters/etc or /etc/Tivoli/tecad/etc (which is a link to the TME adapter directory)
non-TME	Not applicable	*path*/etc where the adapter was manually installed or /etc/Tivoli/tecad/etc (which is a link to the TME adapter directory)

- *(TEC.0067: CAT II) The IAO will ensure that the TEC Adapter file permissions are restricted in accordance with Appendix B.8, the UNIX STIG, and DODI 8500.2.*

B.3.3.5 TEC Gateway Configuration File

The following table describes the location of the configuration file by platform.

Platform	Path
UNIX	/etc/Tivoli/tec/tec_gateway.conf
Windows	%SystemRoot%\drivers\etc\Tivoli\tec\tec_gateway.conf

- *(TEC.0068: CAT II) The IAO will ensure that the TEC Gateway Configuration file permissions are restricted in accordance with Appendix B.8, the Windows and UNIX STIGs, and DODI 8500.2.*

B.4 IBM Tivoli Monitoring (Tivoli Monitoring)

B.4.1 General Overview

The IBM Tivoli Monitoring product, also known Tivoli Monitoring, is designed to provide availability management support in a Tivoli Enterprise. The application product detects bottlenecks and other potential problems and provides automatic recovery from critical situations. Thus, eliminating the need for manually scanning performance data. Some of the types of resources that are monitored by Tivoli Monitoring are: system hardware, operating systems, and applications.

Tivoli Monitoring can be integrated with other Tivoli products such as Tivoli Business Systems Manager and Tivoli Enterprise Console for total availability monitoring. Tivoli Monitoring was previously known as Tivoli Distributed Monitoring (Advanced Edition).

- *(TTM.0001: CAT II) The IAO will ensure that the TIVOLI MONITORING product is installed in the Tivoli Enterprise either using the TMF or authorized scripts.*

- *(TTM.0002: CAT I) The IAO will ensure that unsupported TIVOLI MONITORING software is removed or upgraded prior to a vendor dropping support.*

- *(TTM.0003: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading DBMS systems prior to the date the vendor drops security patch support.*

- *(TTM.0004: CAT II) The IAO will ensure that the creation, distribution and implementation of Tivoli Monitoring endpoint monitors or adapters are restricted to TMR administrators and authorized personnel.*

- *(TTM.0005: CAT II) The IAO will ensure that the Tivoli Monitoring files are restricted from unauthorized update and access by unauthorized processes, scripts or users.*

B.4.2. General Considerations

B.4.2.1 Architecture and Components

The implementation of Tivoli Monitoring is based on the structure of the Tivoli Management Regions Environment, which is a three-tier management structure. Components are implemented on the TMR server, the gateways/managed nodes of the endpoints to that are to be monitored and on the endpoints, which perform the monitoring and collection of data.

B.4.2.1.1 Tivoli Monitoring Manager

The highest tier, known as the Tivoli Monitoring Manager, runs on the TMR server. It defines the container used for distributing the resource models to target resources. The Tivoli Monitoring Manager provides the user with a graphical user interface integrated with the Tivoli Desktop and a command line interface to manage the resource models and the Tivoli Monitoring profiles. The Tivoli Desktop is also available for use on the Tivoli management gateway, which can perform similar functions. The server component interacts with the Tivoli MDist 2 functionality to manage the distribution of the Tivoli Monitoring profiles to the subscribers asynchronously. The database, which contains available default resource models, is maintained at the server. Commands issued to manage resource models from gateways are routed to and performed on the server.

- *(TTM.0006: CAT II) The IAO will ensure that the Tivoli Monitoring Manager files are restricted from unauthorized update and access by unauthorized processes, scripts or users.*

- *(TTM.0007: CAT II) The IAO will ensure that the Tivoli Desktop(s) and Tivoli Business Systems Manager workstation(s) are located in a restricted location in accordance with DODI 8500.2.*

- *(TTM.0008: CAT II) The IAO will ensure that use of the Tivoli Desktop(s) and the Tivoli Business Systems Manager workstation(s) will be restricted to authorized users and appropriate roles are assigned in accordance with DODI 8500.2.*

- *(TTM.0009: CAT II) The IAO will ensure that encrypted communication is used between the Tivoli Desktop(s), the Tivoli Business Systems Manager workstation(s) and the supporting servers.*

- *(TTM.0010: CAT II) The IAO will ensure that the Tivoli Monitoring Manager is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.1.2 Tivoli Monitoring Middle Layer

The second layer, known as the Tivoli Monitoring Middle Layer consists of a set of services running at the Tivoli gateways (managed nodes). These services are responsible for mediating and optimizing the interaction between the monitored endpoints, the monitored resources, and the other components internal and external to Tivoli Monitoring.

- *(TTM.0011: CAT II) The IAO will ensure that the Tivoli Monitoring Middle Layer files are restricted from unauthorized update and access by unauthorized processes, scripts or users.*

- *(TTM.0012: CAT II) The IAO will ensure that the Tivoli Desktop(s) connected to gateways are located in a restricted location in accordance with DODI 8500.2.*

- *(TTM.0013: CAT II) The IAO will ensure that use of the Tivoli Desktop(s) connected to gateways will be restricted to authorized users and appropriate roles are assigned in accordance with DODI 8500.2.*

- *(TTM.0014: CAT II) The IAO will ensure that the Tivoli Monitoring Middle Layer is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.1.2.1 Request Manager

The Request Manager is responsible for collecting, storing, and managing all endpoint requests created and used by the Tivoli Monitoring components (such as the Web Health Console and the heartbeat) and other Tivoli applications. The request manager serves as a down-call concentrator: it receives requests from its user components and concentrates them into one request to the endpoint. When the endpoint sends data back, the request manager stores it in a centralized cache on the gateway. The user components can retrieve the data from the gateway cache.

- *(TTM.0015: CAT II) The IAO will ensure that the Request Manager files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0016: CAT II) The IAO will ensure that the Request Manager is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

- *(TTM.0017: CAT II) The IAO will ensure that the Application Proxy files are restricted from unauthorized update and access and in accordance with Appendix B.8.*

B.4.2.1.2.2 Heartbeat Manager

The Heartbeat Manager function is responsible for monitoring the basic signs of life at endpoints. It will only perform this function if it is enabled and will only perform the function on endpoints that are attached to the gateway. The heartbeat manager regularly monitors the endpoints, checking to see that they are running correctly. Events related to endpoints status may be sent to the Tivoli Business Systems Manager (provided that the Tivoli Business Systems Manager Adapter component is installed at the gateway), the Tivoli Enterprise Console (Tivoli events only), or the Tivoli Monitoring Notice Group. The Heartbeat Manager maintains information about the endpoint in a local persistent cache and uses the request manager for handling the status requests that are periodically sent to the endpoints.

- *(TTM.0018: CAT II) The IAO will ensure that the Heartbeat Manager files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0019: CAT II) The IAO will ensure that the Application Proxy is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

- *(TTM.0020: CAT II) The IAO will ensure that the Application Proxy files are restricted from unauthorized update and access and in accordance with Appendix B.8.*

B.4.2.1.2.3 Data Collector

The Data Collector is responsible for collecting the data logged periodically by Tivoli Monitoring agents to the endpoint database. It is also responsible for moving the data into a centralized database accessed through a RDBMS Interface Module (RIM) object. It uses the MDist 2 framework service to move data from the endpoints to the managed node where it is installed.

The Data Collector uses the request manager for handling the data collection requests that are periodically sent to the endpoints. The collected data is aggregated every 24 hours and is used by Tivoli Monitoring to interact with Tivoli Data Warehouse.

- *(TTM.0021: CAT II) The IAO will ensure that the Data Collector files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0022: CAT II) The IAO will ensure that the Data Collector is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

- *(TTM.0023: CAT II) The IAO will ensure that the RIM Database is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

- *(TTM.0024: CAT II) The IAO will ensure that the MDist 2 files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

B.4.2.1.2.4 Tivoli Business Systems Manager Adapter

The Tivoli Business Systems Manager Adapter component is responsible for forwarding discovery and status events to the Tivoli Business Systems Manager.

- *(TTM.0025: CAT II) The IAO will ensure that the Tivoli Business System Manager Adapter files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0026: CAT II) The IAO will ensure that the Tivoli Business System Manager Adapter component is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

- *(TTM.0027: CAT II) The IAO will ensure that the Tivoli Desktop(s) and Tivoli Business Systems Manager workstation(s) are located in a restricted location in accordance with DODI 8500.2.*

- *(TTM.0028: CAT II) The IAO will ensure that use of the Tivoli Desktop(s) and the Tivoli Business Systems Manager workstation(s) is restricted to authorized users and appropriate roles are assigned in accordance with DODI 8500.2.*

- *(TTM.0029: CAT II) The IAO will ensure that encrypted communication is used between the Tivoli Desktop(s), the Tivoli Business Systems Manager workstation(s), and the supporting servers.*

B.4.2.1.2.5 Task Manager

The Task Manager is responsible for invoking Tivoli tasks that are to be performed when a consolidated event is generated by the resource model engine.

- *(TTM.0030: CAT II) The IAO will ensure that the Task Manager files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.031: CAT II) The IAO will ensure that the Task Manager is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.1.2.6 Tivoli Monitoring Gateway (Upcall Collector)

The Tivoli Monitoring Gateway (Upcall Collector) is responsible for dispatching to the appropriate processor the up-calls coming from the endpoints attached to the Tivoli gateway on which it is installed. The processors that are dispatched by the Tivoli Monitoring Gateway are: heartbeat manager, task manager, and the Tivoli Business Systems Manager Adapter.

- *(TTM.0032: CAT II) The IAO will ensure that the Tivoli Monitoring Gateway files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0033: CAT II) The IAO will ensure that the Tivoli Monitoring Gateway is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.1.3 Tivoli Monitoring Agent

The lowest tier, known as the Tivoli Monitoring Agent requires that a Tivoli management agent be installed on the endpoint. It performs resource management through one or more resource models that are distributed to the endpoint with a Tivoli Monitoring profile. The agent is installed automatically when a Tivoli Monitoring profile is distributed to the endpoint, the first time. The agent is comprised of two main parts: a Resource Model Engine (RME), which is responsible for interpreting resource models, and a set of Tivoli down-call methods that the other components invoke to interact with the RME and the resource models.

- *(TTM.0034: CAT II) The IAO will ensure that the Tivoli Monitoring Agent files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0035: CAT II) The IAO will ensure that the Tivoli Monitoring Agent is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.1.3.1 Resource Model Engine (RME)

The Resource Model Engine (RME) is referred to as the heart of the system locally. The RME is driven by resource models. It implements the monitoring of the resources (operating system and applications) to detect indications of specific situations. When an indication for a sufficient period of time or intensity the RME can activate actions or to send notifications to higher-level management applications depending on how the profiles have been configured. The RME enables resource models to interact with Windows Management Instrumentation (WMI) on Windows platforms, Instrumentation Library Type (ILT) providers, and shell scripts to collect data to be analyzed. The collected data may be logged to a local database in either raw or aggregated format.

- *(TTM.0036: CAT II) The IAO will ensure that the Resource Model Engine (RME) files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0037: CAT II) The IAO will ensure that the RME is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.1.3.2 TME Down-call Methods

The TME down-call methods are invoked by the other components of Tivoli Monitoring when they need to interact with the RME for managing the lifecycle of Tivoli Monitoring profiles and resource models, retrieving information about the status of resources and resource models, and retrieving the data that was logged locally.

- *(TTM.0038: CAT II) The IAO will ensure that the TME down-call method files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0039: CAT II) The IAO will ensure that the TME down-call methods are located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.1.4 Web Health Console

The Web Health Console is the web-based graphical user interface for Tivoli Monitoring. It can be accessed from any Netscape 6.2 (or later) and Internet Explorer 6.x browser that runs on any system connected through TCP/IP to the TMR. The Web Health Console can be connected to any Tivoli management region server or managed node and configured to monitor any or all of the endpoints that are found in that region (assuming that all of the gateways are interconnected).

The Web Health Console enables an administrator to view real time information about a specific problem, and check the status (or health) of a set of endpoints. The status reflected by the Web Health Console reflects the state of the endpoint, such as running or stopped, and the health is represented as a numeric value determined by resource model settings. A numeric value between 100 (perfect health) and zero (corresponding event conditions met) is used to represent the health of each resource.

The Web Health Console can also be configured to work with historical data that has previously been logged to a Tivoli Monitoring database. The Web Health Console can be used to perform targeted analysis of problems associated with individual endpoints when an event is sent to the Tivoli Enterprise Console. An administrator can use the online and historical data to follow up on specific problems with single endpoints.

All user management and security is handled through the IBM Tivoli management environment. This includes creating users and passwords as well as assigning authority.

- *(TTM.0040: CAT II) The IAO will ensure that the Web Health Console files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0041: CAT II) The IAO will ensure that the Web Health Console is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

- *(TTM.0042: CAT I) The IAO will ensure that the Web Health Console(s) is/(are) located in a restricted location in accordance with DODI 8500.2.*

- *(TTM.0043: CAT II) The IAO will ensure that use of the Web Health Console(s) will be restricted to authorized users and appropriate roles are assigned in accordance with DODI 8500.2.*

- *(TTM.0044: CAT II) The IAO will ensure that encrypted communication is used between the Web Health Console(s) and the supporting servers.*

The following table lists the software platform supported by the Web Health Console.

Software Platform	Version
AIX	5.1
HP-UX	11.0
Red Hat Linux for Intel	7.1
Solaris	2.8
Windows 2000 Server and Advanced Server	SP1 or SP2
Windows NT Server	Version 4.0 SP6

NOTE: Since the Web Health Console runs on Netscape 6.2 (or later) and Internet Explorer 6.x. The software platforms supported for these browsers are:

- *Windows 2000 Server and Windows 98 for Internet Explorer 6.x*
- *Red Hat Linux and Solaris 2.8 for Netscape 6.2 (or later)*

- *(TTM.0045: CAT II) The IAO will ensure that all security mechanism for Netscape and Explorer have been implemented in accordance with the Desktop STIG and in accordance with DODI 8500.2.*

As part of the Web Health Console installation three software components are installed:

- WebSphere Application Server, Advanced Edition, Single Server, 4.0.2
- IBM HTTP Server
- Web Health Console

- *(TTM.0046: CAT II) The IAO will ensure that WebSphere Application Server has been implemented in accordance with the UNIX and Windows STIGs, and in accordance with DODI 8500.2.*

B.4.2.1.5 Gathering Historical Data Component

The Gathering Historical Data component enables Tivoli Monitoring to use Tivoli Decision Support for Server Performance Prediction (Advanced Edition) and Tivoli Data Warehouse. It uses data collected by specific Tivoli Monitoring resource models to populate a database on the Tivoli server where it is installed. The collected data is aggregated every 24 hours and added to the Tivoli Monitoring database, from where it can be used in analyses that helps in the planning for network growth.

- *(TTM.0047: CAT II) The IAO will ensure that the Gathering Historical Data Component is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

- *(TTM.0048: CAT II) The IAO will ensure that the Gathering Historical Data files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

B.4.2.1.6 TME Data Warehouse Support Component

The Tivoli Enterprise Data Warehouse Support component enables the integration of Tivoli Monitoring with Tivoli Data Warehouse. As opposed to the Gathering Historical Data component, it interacts with the IBM Tivoli Monitoring for other products. To interact with IBM Tivoli Monitoring, the IBM Tivoli Monitoring Tivoli Enterprise Data Warehouse Support Component, also known as the data collector, is installed on the TMR server and on all gateways that support monitored endpoints. The data collector component collects the monitored data and stores it in a RIM database.

Tivoli Enterprise Data Warehouse provides the capability to store historical data and generate reports and graphs. The Tivoli Enterprise Data Warehouse controller is installed on a Windows server, which can be on the TMR or not. Other warehouse components can be installed on other servers, which can be Windows servers or not.

It should be noted that the integration with Tivoli Data Warehouse is provided through: an ETL1 script responsible for extracting data from the centralized Tivoli Monitoring database and loading it to the Central Data Warehouse, and an ETL2 script responsible for creating a data-mart supporting a set of sample reports.

- *(TTM.0049: CAT II) The IAO will ensure that the TME Data Warehouse Support Component files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0050: CAT II) The IAO will ensure that the TME Data Warehouse Support Component is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.1.7 Tivoli Monitoring Workbench

Tivoli Monitoring Workbench is a stand-alone programming product that provides an environment specifically for creating, modifying, debugging, and packaging resource models.

- *(TTM.0051: CAT II) The IAO will ensure that the Tivoli Monitoring Workbench files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

- *(TTM.0052: CAT II) The IAO will ensure that the Tivoli Monitoring Workbench is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.2 Files

The Tivoli Monitoring software can be installed using the SIS or by script. All files for Tivoli Monitoring will be located on managed nodes under the directory structure that is subordinate to a logical directory known as the BINDIR directory. For the endpoints the Tivoli Monitoring files are located under the LCF_DATDIR/LCFNEW.

- *(TTM.0053: CAT II) The IAO will ensure that the Tivoli Monitoring directories and files are secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

B.4.2.3 Platforms

System	Versions	Server	Gateway	Endpoint
AIX	4.3.3, 5.1.0.C, 5.2	X	X	X
Solaris	2.6	X	X	X
Windows NT, Version 4.0	Service Packs 6, and 6a	X	X	X
Windows 2000	Server, Advanced Server, Professional, DataCenter Svr sp3+	X	X	X
Windows Server 2003	Standard, Enterprise	X	X	X
Windows XP	Professional			X
Turbo Linux Svr	6.1, 6.5	X	X	X
SuSE	6.4, 7.0, 7.1, 7.2, 8.0, and 8.1	X	X	X
SuSE SLES	7.0	X	X	X
SLES	7.0 for S/390 and z/Series	X	X	X
HP-UX	11, 11i	X	X	X
OS/400	5.1, 5.2			X
RedHat Server (IA32)	7.0, 7.1, 7.2, and 7.3	X	X	X
RedHat Ent Linux (IA32)	2.1	X	X	X
RedHat for OS/390	6.0	X	X	X
RedHat for Intel	7.3, 8.0	X	X	X

System	Versions	Server	Gateway	Endpoint
UL (SLES 8)	1.0 for IA32 sp2+	X	X	X
UL (SLES 8)	1.0 for z/Series sp2+	X	X	X
UL (SLES 8)	1.0 for PowerPC			X

NOTE: Some platforms have additional restrictions and the Tivoli Monitoring documentation should be consulted.

- *(TMQ.0054: CAT II) The IAO will ensure that Tivoli Monitoring is installed on secured platforms in accordance with the appropriate platform STIGs, and DODI 8500.2.*

B.4.3 General Security Considerations

The unique considerations for Tivoli Monitoring include:

- Access to Tivoli Monitoring functions is controlled through the assignment of Tivoli authorization roles to Tivoli users, userids and passwords. Without proper assignment of these roles, userids and passwords, the confidentiality and integrity of the WebSphere object data could be compromised.

- Because Tivoli Monitoring requires the use of an RDBMS for data storage, the secure configuration of the client access software and the RDBMS are significant.

- As with most software products, the Tivoli Monitoring is composed of program and data files for which proper access controls are essential. Although most of the Tivoli Monitoring files will reside in directories subject to access controls required for the Framework, there are some Tivoli Monitoring directories and files that require specific access controls.

- There are two considerations relative to the security of the Tivoli Monitoring server:

  - The host OS must be configured securely.
  - The integrity of the server software must be maintained.

- There are three considerations relative to the security of the RIM host:

  - The host OS must be configured securely.
  - If required on the platform, the OS user account utilized by Tivoli Monitoring functions must be defined properly.
  - The integrity of the RDBMS client software must be maintained.

- There are four considerations relative to the security of the RDBMS server:

  - The host OS must be configured securely.
  - The RDBMS software must be configured securely.
  - If required on the platform, the OS user account utilized by Tivoli Monitoring functions must be defined properly.
  - The RDBMS user account utilized by the Tivoli Monitoring functions must be defined properly.

- There are two considerations relative to the security of the Gateways:

  - The host OS must be configured securely.
  - The integrity of the Tivoli Monitoring Gateway software must be maintained.

- There are four considerations relative to the security of the Tivoli Monitoring and Web Health Console interfaces:

  - The integrity of the Tivoli Monitoring remote access facility software must be maintained.
  - The proper assignment of the userids, passwords and Tivoli authorization role for the link functions must be managed.
  - The confidentiality of the user authentication data that passes over the network connection between the endpoint or managed node client and the TMR server must be maintained.
  - The integrity of the user machine-based must be maintained.

The host OS security consideration applies generically to the host. It is addressed through the platform security requirements described in the applicable platform STIG. Specifically, the RIM host must be compliant with the STIG that covers the OS used on the RIM host.

The security consideration for the confidentiality of the user authentication data applies to the transmission of the Tivoli user password from the web browser on the endpoint or managed node client to the TMR server. The HTTP protocol between the client and server does not provide encryption. This makes the password vulnerable to capture during transmission over the network.

Password data is encrypted in accordance with the *DODI 8500.2, IA Controls for Individual Identification and Authentication*. Passwords must be encrypted both for storage and for transmission. The actions that are required to address these considerations are addressed in the following section of this document.

Tivoli Monitoring exploits the infrastructure provided by TMF to enable the functionality of the product across firewalls. To protect the privacy and data integrity, TMF enables the configuring of either Bulk Data Transfer (BDT) proxy mechanism and Secure Socket Layer 3 (SSL3) encryption support, or install the TMF Firewall Security Toolbox. In either case, the implementation of encryption must be implemented either through Tivoli or other methods.

B.4.4 Specific Security Considerations

This section describes the specific considerations and required actions to help ensure that the Tivoli Monitoring components are implemented in a secure fashion. As noted in the previous section, compliance with the requirements in the Framework section of this document is assumed as the prerequisite for the information here.

Department of Defense (DOD) Instruction 8500.2, Information Assurance (IA) Implementation, documents IA controls that apply to DOD information systems. The following integrity and confidentiality controls from *DODI 8500.2* are the basis for the requirements in this section: Access for Need-to-Know; Audit Trail, Monitoring, Analysis, and Reporting; Changes to Data; Individual Identification and Authentication; Least Privilege; and Privileged Account Control.

The following subsections describe the considerations and requirements of the Tivoli Monitoring. When reviewing these requirements, the following must be noted:

- The specific directory names in the requirements reflect defaults indicated in vendor documentation. If a site or organization deploying Tivoli chooses other names, the requirements apply to the site-specific names.

- The structure of UNIX permissions can make it difficult to restrict file access to multiple groups of users. The approach taken in this section is to allow UNIX “world” access in cases where non-update (e.g., read or execute) access permission is required for users who may be members of multiple groups and there is no specific need for the files to be inaccessible to other users.

B.4.4.1 Authorization Roles and Platform Security

Authorization roles defined within the Management Framework provide role based access control over functions in the Tivoli products. A Tivoli administrator must be assigned the associated roles to be permitted to perform specific product functions. Authorization roles are required for the installation of the product through Software Installation Services of the TMF. The authorization roles associated with the installation of this product are included as part of the TMF section of this document.

The following table describes the installation of Tivoli Monitoring at the platform level requires the following:

Platform	Authorization
Windows	The SA who installs the product must have membership in the Administrators group.
UNIX	The SA who installs the product must have root privileges.

- *(TTM.0055: CAT II) The IAO will ensure that all authorization roles are assigned on a need to know basis and in accordance with DODI 8500.2.*

- *(TTM.0056: CAT II) The IAO will ensure that all userids and passwords are assigned on a need-to-know basis and in accordance with DODI 8500.2.*

B.4.4.2 File Permissions

The Tivoli Monitoring software is installed in a directory structure that is subordinate to a logical directory known as the BINDIR directory. The physical implementation of the BINDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. Some of the Tivoli Monitoring software files are installed in directories that are shared with the Management Framework. These directories are protected through the specific Management Framework requirements elsewhere in this document. Requirements for the Tivoli Monitoring -specific directories are also addressed in *Appendix B.8*. Tivoli Monitoring may be installed using the TMF or by using a script.

Tivoli files for Windows NT are, by default, stored under the *\Tivoli* directory on the root of the selected drive. Tivoli will also write install and other log files to *%DBDIR%\tmp*.

- *(TTM.0057: CAT II) The IAO will ensure that all file permissions and authorizations are secured in accordance with Appendix B.8, the appropriate platform STIG, and DODI 8500.2.*

B.5 Tivoli Inventory

B.5.1 General Overview

TME 10 Inventory is one of the systems management applications based on the TME 10 Framework (known as the Management Framework). TME10 Inventory, hereafter called Inventory, gathers hardware and software inventory information to help manage distributed client/server enterprises. Inventory provides the following major functions:

- Scans for and stores hardware and software information.
- Monitors and records changes in hardware and software configurations.
- Supports pre-defined and ad hoc queries to analyze collected information.

Inventory is part of the systems management solution line that is undergoing active development by IBM. The information in this section is based on published documentation for version 3.6 of Inventory. Please refer to the *TME 10 Inventory User's Guide, Version 3.6,* for specific product information.

Inventory uses the resources and management components of the Framework along with its own individual components. The following diagram illustrates the logical structure.

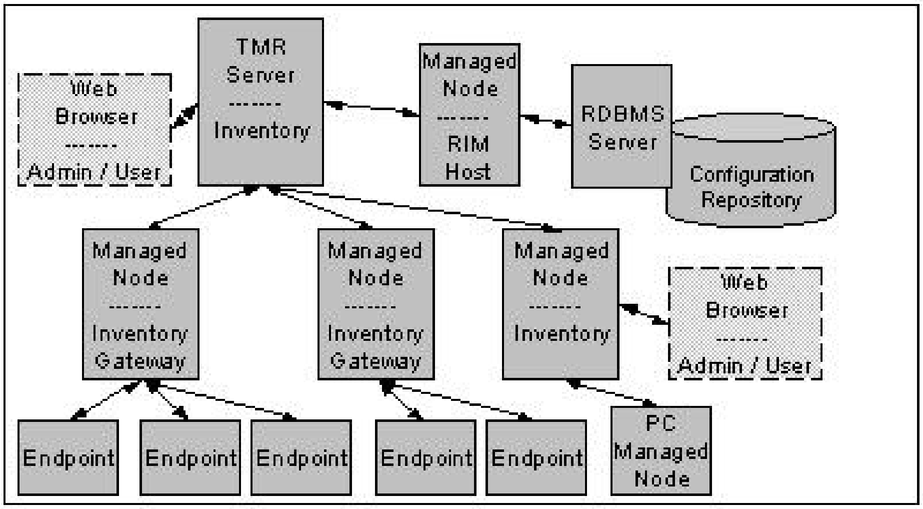
TMR Server ------- Inventory
Web Browser ------- Admin / User
Managed Node ------- RIM Host
RDBMS Server
Configuration Repository
Managed Node ------- Inventory Gateway
Managed Node ------- Inventory Gateway
Managed Node ------- Inventory
Web Browser ------- Admin / User
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint
PC Managed Node

Figure B-1. Tivoli Inventory Logical Structure

It should be noted that the administrative and user functions supported through the web browser could be performed from any platform with the required network access to the Inventory server application.

From a component perspective, Inventory functions are accomplished as follows:

- The Inventory server application is installed on the TMR Server and on managed nodes that might be used to execute Inventory administration commands. This application includes the administrator interface and the scanning software.

- The TMR Server issues scan requests, retrieves scan results from managed nodes, gateways, and endpoints, and sends scan results to the RIM host. The pre-defined queries supplied with Inventory are also installed on the TMR server.

- The Inventory Gateway application is installed on managed nodes that act as gateways for endpoints. The Gateway provides a collection point for communications between the assigned endpoints and the TMR server.

- The RDBMS Interface Module (RIM) host is a managed node that contains client software needed to communicate with the RDBMS server. There can only be one RIM host in each TMR.

- The RDBMS server is the host platform for the database containing the Inventory information. This database is known as the Inventory configuration repository. Multiple TMRs can use a single RDBMS server.

- Inventory scanning applications collect data on individual hosts. For endpoints, the scanning application is downloaded dynamically, as needed, from the Inventory server or Inventory Gateway server. For PC managed nodes (nodes running the Tivoli PC Agent), the PC Scanning Program has to be independently installed.

- The web-based interface includes elements for Tivoli administrators and for other Tivoli users. For administrators there is a Java applet for managing software signatures and a template of the user data web form that can be customized for use in collecting user information. For users there is an Inventory profile web page to initiate certain user-accessible Inventory operations and the user data web form to submit user information. The web-based interface for users is referred to as UserLink.

It is common to consolidate some of these components on a single host in order to reduce hardware requirements. For example, the RIM Host and the RDBMS Server might be combined with the TMR Server on a single UNIX host. The Inventory server application and the Inventory Gateway application can also be combined on a single host. A consideration for consolidation is noted in *Section B.5.2, General Security Considerations*.

Inventory uses the Framework profile and subscriber concepts to accomplish its tasks. The following provides a very high level view of some of the information flow:

- Administrators define profiles that are containers for Inventory-specific information. Profile managers are defined groups of profiles to be distributed to designated target machines known as subscribers.

- Subscribers can be managed nodes, endpoints, or PC managed nodes. Scanning software on each subscriber uses instructions in a distributed profile to scan the machine and create Management Information Format (MIF) files with the desired data.

- Inventory processes the MIF files and sends results to the TMR server where the data is sent through the RIM host to the configuration repository on the RDBMS server.

- After Inventory data is populated in the configuration repository, the Framework query facility can be used to analyze it using pre-defined or ad hoc queries.

B.5.1.1 Server and Client Component Platforms

Inventory components for managed nodes, endpoints, and PC managed nodes are supported on a variety of host operating systems. The following table summarizes this support by component type.

Tivoli Component Type	Supported Platform(s) for Inventory
Managed Node	AIX 4.x, HP-UX, Solaris, Windows NT
Endpoint	AIX 4.x, AS/400, HP-UX, NetWare, OS/2, Solaris, Windows 9x, Windows 2000, Windows NT
PC Managed Node (PC Agent)	NetWare, OS/2, Windows 9x, Windows 2000, Windows NT

Please note that this support information is highly release dependent. The information provided here is based on details from the *Tivoli Inventory Release Notes, Version 3.6.2.*

In order to provide appropriate, focused support for the STIG audience, the discussion for Inventory component platforms in this document is limited to UNIX and Windows (NT –based) platforms.

B.5.1.2 Configuration Repository Platforms

The Inventory configuration repository requires a RIM Host platform with RDBMS client software and an RDBMS Server platform. These platforms may be combined and a variety of RDBMSs are supported. The following table summarizes this support.

Configuration Repository Component	Supported Platform(s) for Inventory
RIM Host	AIX, HP-UX, Solaris, Windows NT
RDBMS Client	DB2 5.x, Informix 2.5 CLI, MS SQL Server, Oracle 7.3.x, Sybase 11.1.1
RDBMS Server	DB2 5.x, DB2 5.x for OS/390, Informix 7.3, MS SQL Server, Oracle, Sybase 11.x

Please note that this support information is highly release dependent. The information provided here is based on details from the *Tivoli Inventory Release Notes, Version 3.6.2.*

In order to provide appropriate, focused support for the STIG audience, the discussion for Inventory configuration repository platforms in this document is limited to Oracle platforms.

B.5.2 General Security Considerations

Because Inventory makes use of the infrastructure provided by the Management Framework, many of the security considerations for Inventory are addressed by controls required for the Framework. Compliance with all of the requirements in the Framework section of this document is assumed as the basis for the considerations described here.

The unique considerations for Inventory include:

- Access to Inventory functions is controlled through the assignment of Tivoli authorization roles to Tivoli users. Without proper assignment of these roles, the confidentiality and integrity of the Inventory data could be compromised.

- Because Inventory requires the use of an RDBMS for data storage, the secure configuration of the client access software and the RDBMS are significant.

- As with most software products, Inventory is composed of program and data files for which proper access controls are essential. Although most of the Inventory files will reside in directories subject to access controls required for the Framework, there are some Inventory directories and files that require specific access controls.

- If the Inventory web interface features are used, administrative or user transactions are processed through HTML pages. The integrity of the HTML files must be assured to maintain transaction security.

The actions that are required to address these considerations are addressed in the following section of this document.

The question of consolidating certain Inventory components on a single platform warrants individual consideration by each organization deploying Tivoli. The particular issue concerns combining the RDBMS server and RIM Host with Inventory on the TMR server. The *Database Security Technical Implementation Guide* recommends that any DBMS be installed on a host system dedicated to its support. By separating the DBMS server, access to that platform can be more finely controlled resulting in limited exposure to any vulnerabilities in the DBMS software. For this reason, sites should install the RDBMS server and RIM Host on a platform that is separate from the TMR server platform when practical.

B.5.3 Specific Security Considerations

This section describes the specific considerations and required actions to help ensure that the Inventory components are implemented in a secure fashion. As noted in the previous section, compliance with the requirements in the Framework section of this document is assumed as the prerequisite for the information here.

Department of Defense (DOD) Instruction 8500.2, Information Assurance (IA) Implementation, documents IA controls that apply to DOD information systems. The following integrity and confidentiality controls from *DODI 8500.2* are the basis for the requirements in this section: Access for Need-to-Know; Audit Trail, Monitoring, Analysis, and Reporting; Changes to Data; Individual Identification and Authentication; Least Privilege; and Privileged Account Control.

The following subsections describe the considerations and requirements organized by major Inventory component. When reviewing these requirements, the following must be noted:

- The specific file and directory names referenced in the requirements and specified in the appendix reflect defaults indicated in vendor documentation. If a site or organization deploying Tivoli chooses other names, the requirements apply to the site-specific names.

- The structure of UNIX permissions can make it difficult to restrict file access to multiple groups of users. The approach taken in this section is to allow UNIX “world” access in cases where non-update (e.g., read or execute) access permission is required for users who may be members of multiple groups and there is no specific need for the files to be inaccessible to other users.

B.5.3.1 Tivoli Authorization Roles

Authorization roles defined within the Management Framework provide role based access control over functions in the Tivoli products. A Tivoli administrator must be assigned the associated roles to be permitted to perform specific product functions.

Authorization roles such as senior or super enable a broad range of capabilities in multiple products. Other roles such as Inventory_scan enable only limited functions in one product. The following table lists the authorization roles and the associated functions within Inventory.

Role	Capability
admin	Configuration Repository data: view Inventory profiles: distribute, edit, view Inventory profile subscribers: set Inventory queries: create, edit, execute, view UserLink web functions - Inventory profile: distribute - User Data Form: submit
backup	TMR object database: backup
Install_client	TMR managed nodes: install
Install_product	Inventory products: install
Inventory_end_user	UserLink web functions - Inventory profile: distribute - User Data Form: submit
Inventory_scan	Inventory profiles: distribute
Inventory_view	Inventory profiles: view Inventory queries: execute

Role	Capability
Query_edit	Inventory queries: edit, execute, view
Query_execute	Inventory queries: execute, view
Query_view	Configuration Repository data: view Inventory queries: view
restore	TMR object database: restore
RIM_update	Inventory software signatures: add, delete, edit -- user must also have RIM_view UserLink web functions - User Data Template: change User Data Form contents -- user must also have RIM_view
RIM_view	Administrator web pages: access Configuration Repository data: view Inventory queries: execute
senior	Configuration Repository data: view Inventory products: install Inventory profiles: clone, create, delete, edit, distribute, view Inventory profile subscribers: set Inventory queries: create, edit, execute, view Inventory software signatures: add, delete, edit UserLink web functions - Inventory profile: distribute - User Data Form: submit
super	Configuration Repository data: view Inventory products: install Inventory profiles: clone, create, delete, edit, distribute, view Inventory profile subscribers: set Inventory queries: create, edit, execute, view Inventory software signatures: add, delete, edit UserLink web functions - Inventory profile: distribute - User Data Form: submit
user	Configuration Repository data: view Inventory profiles: view Inventory queries: view UserLink web functions - Inventory profile: distribute - User Data Form: submit

The assignment of Tivoli authorization roles is restricted and tracked in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know, Least Privilege, and Privileged Account Control*. Roles must be assigned only to users who require the associated privileges to perform designated job functions.

- *(TIN.0001: CAT II) The IAO will ensure that Tivoli authorization roles that control access to functions of Inventory are assigned only to appropriate authorized users, and that the assignment of roles is documented.*

B.5.3.2 RDBMS Interface Module (RIM) Host Considerations

The RDBMS Interface Module (RIM) host is responsible for communications between TMR servers and the RDBMS server that hosts the Inventory configuration repository. In practical terms the RIM host is <u>the</u> client of the RDBMS server used for Inventory.

It is possible to combine the logical functions of the RIM host and the RDBMS server on a single physical platform. Under these circumstances, the security controls defined for the RDBMS server would override those defined in this section. Sites with such a combined configuration should refer to the requirements in *Section B.5.3.3, RDBMS Server Considerations*, in place of those in this section.

The RIM host is the single path through which Inventory data is stored and retrieved. This offers increased integrity for the data, but also represents a potential vulnerability as a single point of failure. If this RDBMS client is compromised, the confidentiality, integrity, or availability of the Inventory data can be lost. Therefore the security of the RIM host is essential to the protection of Inventory.

There are three considerations relative to the security of the RIM host:

- The host OS must be configured securely.
- If required on the platform, the OS user account utilized by Inventory functions must be defined properly.
- The integrity of the RDBMS client software must be maintained.

The host OS security consideration applies generically to the host. It is addressed through the platform security requirements described in the applicable DISA STIG. Specifically, the RIM host must be compliant with the STIG that covers the OS used on the RIM host.

The security consideration for the OS user account used for Inventory applies to the account defined on RIM hosts running Windows NT, Windows 2000, or HP-UX. The tmersrvd account must be defined for these platforms. Because the RIM host is a Tivoli managed node, this account is defined as part of the Management Framework implementation. Please see the TMF coverage in this document for information about the definition of this account.

The security consideration for the integrity of the RDBMS client software applies to the executable and configuration files required to connect to the RDBMS server. If the files are altered or deleted, access to the RDBMS server could be lost or it could be redirected to a counterfeit server.

The assignment of directory and file access permissions for the RDBMS client software on the RIM host is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as the account used by the Inventory process, are allowed execute access to program files and read access to other objects.

B.5.3.2.1 RDBMS Interface Module (RIM) Host Considerations – Oracle

The Oracle RDBMS client software is installed on a RIM host in a directory structure that is subordinate to a logical directory known as the ORACLE_HOME directory. The physical implementation of the ORACLE_HOME directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices.

- *(TIN.0002: CAT II) The SA will ensure that access to the directories and files containing the Oracle client software on RIM Host systems is restricted in accordance with the permissions in Appendix B.8.*

On MS Windows systems, the Windows registry contains parameters used for the execution of the Oracle RDBMS client software. If these parameters were altered incorrectly or with malicious intent, the client software might become unusable or might store or return invalid data. Therefore the registry entries must be protected from unauthorized update.

- *(TIN.0004: CAT II) The SA will ensure that access to the Windows registry entries for the Oracle client software on Windows RIM Host systems is restricted in accordance with the permissions in Appendix B.8.*

B.5.3.3 RDBMS Server Considerations

The RDBMS server used for Inventory is the host of the Inventory configuration repository. This is the store of information collected by the Inventory scanning components. The RDBMS server runs a COTS database product such as Oracle, MS SQL Server, or DB2.

As noted previously, it is possible to combine the logical functions of the RIM host and the RDBMS server on a single physical platform. Under these circumstances, the security controls defined in this section would override those defined in *Section B.5.3.2.1, RDBMS Interface Module (RIM) Host Considerations - Oracle*.

The Inventory configuration repository is the primary data asset of Inventory. If access to the configuration repository is lost or its data is corrupted or deleted, the primary function of Inventory can become degraded or unavailable. If access is not controlled, sensitive configuration information for site systems could be disclosed to unauthorized users.

There are four considerations relative to the security of the RDBMS server:

- The host OS must be configured securely.
- The RDBMS software must be configured securely.
- If required on the platform, the OS user account utilized by Inventory functions must be defined properly.
- The RDBMS user account utilized by Inventory functions must be defined properly.

The host OS security consideration applies generically to the RDBMS server host. It is addressed through the OS security requirements described in the applicable DISA STIG. Specifically, the RDBMS server host must be compliant with the STIG that covers the OS used on the RDBMS server.

The RDBMS software security consideration applies generically to the installed RDBMS. It is addressed through the database security requirements described in the applicable DISA STIG. Specifically, the RDBMS software must be compliant with the *Database Security Technical Implementation Guide*.

The security consideration for the OS user account used for Inventory applies when the RDBMS server is combined on the same physical platform as the RIM host. If this platform runs Windows NT, Windows 2000, or HP-UX, the tmersrvd account must be defined. Because the RIM host is a Tivoli managed node, this account is defined as part of the Management Framework implementation. Please see the TMF coverage in this document for information about the definition of this account.

The security consideration for the RDBMS user account used for Inventory applies to the account that is defined during setup of the Inventory configuration repository on the RDBMS server. The default user name for this account is tivoli (or invtiv in later releases) with a default password of tivoli. Processes on the RIM host use this account to access the configuration repository. If use of this account is not properly restricted, access to Inventory data could be lost or the data itself could be deleted, corrupted, or disclosed to unauthorized users.

The password for the Inventory account in the RDBMS is restricted in accordance with the *DODI 8500.2, IA Control for Individual Identification and Authentication*. The password must be different from the vendor default and must conform to complexity requirements.

- *(TIN.0005: CAT I) The Tivoli administrator will ensure that the password for the Inventory account (e.g., tivoli or invtiv) defined to the RDBMS is changed after installation and conforms to the password complexity requirements.*

Note that, in addition to changing the password in the RDBMS, it is necessary to update the Tivoli RIM object that defines the configuration repository. Refer to the vendor documentation for the use of the wsetrimpw command to perform this task.

B.5.3.4 Inventory Server Considerations

The Inventory server software is installed on the TMR server and on managed nodes from which profiles will be created and distributed or other Inventory functions will be executed. The Inventory server software includes executable and configuration files that are installed on a host.

The Inventory server software provides the programs through which Inventory data is collected, processed, and accessed. If access to the server software is lost or the files are corrupted or deleted, the primary function of Inventory can become degraded or unavailable. If access is not controlled, sensitive configuration information for site systems could be disclosed to unauthorized users.

There are three considerations relative to the security of the Inventory server:

- The host OS must be configured securely.
- The integrity of the Inventory server software must be maintained.
- Information posted to the Inventory notice group must be monitored to ensure proper system function.

The host OS security consideration applies generically to the Inventory server host. It is addressed through the OS security requirements described in the applicable DISA STIG. Specifically, the Inventory server host must be compliant with the STIG that covers the OS used on the Inventory server.

The security consideration for the integrity of the Inventory server software applies to the executable and configuration files that are installed with the server. If the files are altered or deleted, access to the Inventory server could become degraded or unavailable. It might also be possible for Inventory data to be deleted, corrupted, or disclosed to unauthorized users.

The Inventory server software is installed in a directory structure that is subordinate to a logical directory known as the BINDIR directory. The physical implementation of the BINDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. Some of the Inventory server software files are installed in directories that are shared with the Management Framework. These directories are protected through the specific Management Framework requirements elsewhere in this document. Requirements for the Inventory-specific directories are addressed here.

The assignment of directory and file access permissions for the Inventory server software is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as the accounts used by Inventory users, are allowed execute and read access.

- *(TIN.0006: CAT II) The SA will ensure that access to the Inventory server directories and files is restricted in accordance with the permissions in Appendix B.8.*

The security consideration for monitoring notices in the Inventory notice group is related to the need to monitor events that may be related to malicious activity. Notices are posted to the Inventory notice group for events and errors related to Inventory profile distribution.

The Tivoli Notification facility is a tool used to assist with monitoring and audit trail creation in accordance with the *DODI 8500.2, IA Control for Audit Trail, Monitoring, Analysis, and Reporting*. Tivoli administrators need to review Inventory notices to validate that unusual or inappropriate activity is not indicative of a security compromise.

- *(TIN.0008: CAT II) The Tivoli administrator will ensure that, where optional, Inventory is configured to send notices to the Inventory notice group.*

- *(TIN.0009: CAT II) The Tivoli administrator will ensure that at least one Tivoli administrator is subscribed to receive Inventory notices.*

B.5.3.5 Inventory Gateway Considerations

The Inventory Gateway software is installed on managed nodes that are Tivoli gateways that perform Inventory functions on endpoints. The Inventory Gateway software includes executable and configuration files that are installed on a host.

The Inventory Gateway software provides the programs through which Inventory data is collected for endpoints. If access to the Gateway software is lost or the files are corrupted or deleted, the endpoint collection function of Inventory can become degraded or unavailable for the set of assigned endpoints. If access is not controlled, sensitive configuration information for some site systems could be disclosed to unauthorized users.

There are two considerations relative to the security of the Inventory Gateway:

- The host OS must be configured securely.
- The integrity of the Inventory Gateway software must be maintained.

The host OS security consideration applies generically to the Inventory Gateway host. It is addressed through the OS security requirements described in the applicable DISA STIG. Specifically, the Inventory Gateway host must be compliant with the STIG that covers the OS used on the Inventory Gateway.

The security consideration for the integrity of the Inventory Gateway software applies to the executable and configuration files that are installed with the Gateway. If the files are altered or deleted, the collection function of the Inventory Gateway could become degraded or unavailable for some endpoints. It might also be possible for Inventory data for some endpoints to be deleted, corrupted, or disclosed to unauthorized users.

The Inventory Gateway software is installed in a directory structure that is subordinate to a logical directory known as the BINDIR directory. The physical implementation of the BINDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. Another variable, INTERP, is also used in the path name for the directory structure. INTERP represents the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems. Some of the Inventory Gateway software files are installed in directories that are shared with the Management Framework. These directories are protected through the specific Management Framework requirements elsewhere in this document. Requirements for the Inventory Gateway-specific directories are addressed here.

The assignment of directory and file access permissions for the Inventory Gateway software is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as the accounts used by Inventory users, are allowed execute and read access.

- *(TIN.0010: CAT II) The SA will ensure that access to the Inventory Gateway server directories and files is restricted in accordance with the permissions in Appendix B.8.*

B.5.3.6 Scanner Application Considerations

The Inventory scanner applications perform data collection on individual hosts. Except for PC managed nodes, the scanner application is downloaded dynamically from the Inventory or Inventory Gateway server when needed. For PC managed nodes the Tivoli administrator independently installs the scanner application, known as the PC Scanning Program. The scanner applications consist of executable files that perform the collection function.

The scanner applications gather hardware and software information that is ultimately stored in the Inventory configuration repository. If the scanner programs are corrupted or deleted, the Inventory data collection function could be unavailable for a target host. If the scanner programs or data are altered, the Inventory data for a host could be corrupted or it could be disclosed to unauthorized parties. An altered scanner program could also be used to conceal a Trojan horse program that could impact the confidentiality, integrity, or availability of a host.

There are two considerations relative to the security of the Inventory scanner applications:

- The host OS must be configured securely.
- The integrity of the Inventory scanner software and its output data must be maintained.

The host OS security consideration applies generically to the host on which the scanner executes. It is addressed through the OS security requirements described in the applicable DISA STIG. Specifically, the host on which the scanner executes must be compliant with the STIG that covers the OS used on that host.

The security consideration for the integrity of the Inventory scanner software and data applies to the scanner's executable files and to the data that is created by them. If the files are altered or deleted, the collection function of the Inventory scanner could become degraded or unavailable for the endpoint or managed node. It might also be possible for Inventory data for the subject host to be deleted, corrupted, or disclosed to unauthorized users.

The Inventory scanner software and its output are stored in different directories according to the type of Tivoli host.

- On PC managed nodes the files reside at and below a logical directory structure known as the SCANDIR directory. The physical implementation of the SCANDIR directory is dependent on site installation choices.

- On other managed nodes the files reside below a logical directory structure known as the DBDIR directory. The physical implementation of the DBDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. There is also an OID directory, where OID in the Tivoli object ID of the node.

- On endpoints the files reside below a logical directory structure known as the LCF_BASE_DIR directory. The physical implementation of the LCF_BASE_DIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices.

Some of the Inventory scanner software files are installed in directories that are shared with the Management Framework or PC Agent. These directories are protected through the specific Management Framework and PC Agent requirements elsewhere in this document. Requirements for the Inventory scanner-specific directories are addressed here.

The assignment of directory and file access permissions for the Inventory scanner software is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as the accounts used by Inventory users, are allowed write access to the scanner output directory.

- *(TIN.0012: CAT II) The SA will ensure that access to the Inventory scanner application directories and files is restricted in accordance with the permissions in Appendix B.8.*

B.5.3.7 Web Interface \ UserLink Application Considerations

Tivoli provides Inventory-specific web interfaces for administrative and user functions:

- Inventory administrators can access a Java applet to manage software signatures and can update a template of a web form that may be used to collect user-entered, custom information.

- General users can be authorized to access an Inventory profile web page to initiate certain Inventory operations and to use a web form to submit custom information. The web-based interface for users is referred to as UserLink.

Inventory administrators may use the Software Signatures Editor in order to remotely update software signatures in the configuration repository. The Software Signatures Editor is accessed from a web browser and implemented through a Java-based applet and web pages that reside on the Inventory server.

Inventory administrators may update the Inventory User Data Template to collect custom information from users who access the UserLink facility. The User Data Template feature is accessed from a web browser and implemented through web pages and executable files that reside on the Inventory server.

Through the UserLink facility, users may be authorized to access Inventory profiles applicable to their machines and to initiate a distribution or self-scan. This feature is implemented through a web page, executable files, and a Tivoli role authorization.

The UserLink facility also makes it possible for users to enter custom information that can be collected and added to the Inventory configuration repository. This can provide data such as assigned user and location that is not collected through the standard Tivoli scanner application. This feature is implemented through a web page, executable files, and a Tivoli role authorization.

To enable the UserLink features, the UserLink web page is downloaded to the users' machines via the distribution of an Inventory profile. Users have to be defined as Tivoli administrators with the Inventory_end_user authorization role. Users performing self-scans connect to the TMR server and a prompt is displayed for the user name and password for the account assigned the Tivoli role. Each time a user invokes UserLink, a notice is sent to the Inventory Notice Group.

The Inventory-specific web interfaces enable three logical functions: update of software signature files, collection of custom user information, and user self-scans. If the components of the software signature update process are deleted or corrupted, the availability or integrity of a significant portion of Inventory software data might be affected. If the components of the custom user information collection process are deleted or corrupted, the availability or integrity of the custom user data in the configuration repository might be affected or the data might be disclosed to unauthorized parties before it is stored. If the components of the self-scan process are deleted or corrupted, the availability or integrity of user initiated profile distributions might be affected. This could result in invalid or malicious software configurations on the user machine.

There are four considerations relative to the security of the Inventory-specific web interfaces:

- The integrity of the Inventory server-based web components used in the Software Signatures Editor, the User Data Template update, and the UserLink facility must be maintained.

- The proper assignment of the Tivoli authorization role for the UserLink functions must be managed.

- The confidentiality of the user authentication data that passes over the network connection between the endpoint or managed node client and the TMR server must be maintained.

- The integrity of the user machine-based UserLink web page must be maintained.

The security consideration for the integrity of the Inventory server-based web page and executable files applies to files installed on the Inventory server. If the files are altered or deleted, the ability to maintain software signatures or to update the User Data Template could become degraded or unavailable. It might also be possible for Inventory data to be deleted, corrupted, or disclosed to unauthorized users.

The Inventory server-based web page and executable files are installed in directories logically owned by the Inventory Server. Access to these directories is controlled through the requirements documented in *Section B.5.3.4, Inventory Server Considerations*.

The security consideration for the proper assignment of the Tivoli authorization role applies to the Inventory_end_user role. If the role were not assigned when required, the UserLink functions would be unavailable to the authorized user. If the role were assigned to an unauthorized user, that user might gain access to Inventory data or might be able to alter a machine’s configuration at an inappropriate time. Access to this role is addressed through the requirements documented in *Section B.5.3.1, Tivoli Authorization Roles*.

The security consideration for the confidentiality of the user authentication data applies to the transmission of the Tivoli user password from the web browser on the endpoint or managed node client to the TMR server. The HTTP protocol between the client and server does not provide encryption. This makes the password vulnerable to capture during transmission over the network.

Password data is encrypted in accordance with the *DODI 8500.2, IA Control for Individual Identification and Authentication*. Passwords must be encrypted both for storage and for transmission.

- *(TIN.0014: CAT II) The SA will ensure that network connections between endpoints or managed nodes using the UserLink facility and the TMR server are encrypted.*

The security consideration for the integrity of the UserLink web page applies to the HTML file that is downloaded via Inventory profile distribution to managed nodes and endpoints. Deleting or altering this file could cause the web page not to load in the browser, to load improperly or link to a counterfeit server, or to include data that might exploit vulnerabilities on the Inventory server.

The **UserLink** web page file is stored in different directories according to the type of Tivoli host.

- On PC managed nodes the default location of the file is DRIVE:\etc\Tivoli, where DRIVE is the drive on which the PC Agent is installed.
- On UNIX managed nodes the default location of the file is /etc/Tivoli.
- On other managed nodes the default location of the file is C:\etc\Tivoli.
- On endpoints the default location of the file is DRIVE:/etc/Tivoli/LANG, where DRIVE is the drive on which the Management Framework agent is installed and LANG represents the code for the language defined for the endpoint.

The assignment of file access permissions for the UserLink web page is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as the accounts used by Inventory users, are allowed read access.

- *(TIN.0015: CAT II) The SA will ensure that access to the Inventory UserLink web page file is restricted in accordance with the permissions in Appendix B.8.*

B.6 Tivoli Software Distribution

B.6.1 General Overview

TME 10 Software Distribution is one of the systems management applications based on the TME 10 Framework (known as the Management Framework). TME 10 Software Distribution, hereafter called Software Distribution, provides the capability to deploy software and data to multiple types of computing platforms spread across local or wide area networks. Software Distribution provides the following major functions:

- Distributes and deploys software, software configuration updates, and data to Windows and UNIX systems using File Package or AutoPack packaging techniques.

- Optimizes network bandwidth usage through network-sensitive distribution controls and caching options.

- Enables simpler construction of updates through the AutoPack Control Center utility.

- Integrates with the Inventory application to allow determination of appropriate distribution targets and tracking of software distributions and removals.

- Integrates with the Tivoli Enterprise Console application to allow Software Distribution operations to be monitored as events.

Software Distribution is part of the systems management solution line that is undergoing active development by IBM. The information in this section is based on published documentation for Version 3.6 of Software Distribution. Please refer to the *TME 10 Software Distribution User's Guide, Version 3.6,* for specific product information.

Software Distribution uses the resources and management components of the Framework along with its own individual components. There are also interfaces to the Tivoli Inventory and Tivoli Enterprise Console products. The diagram that follows illustrates the logical structure.

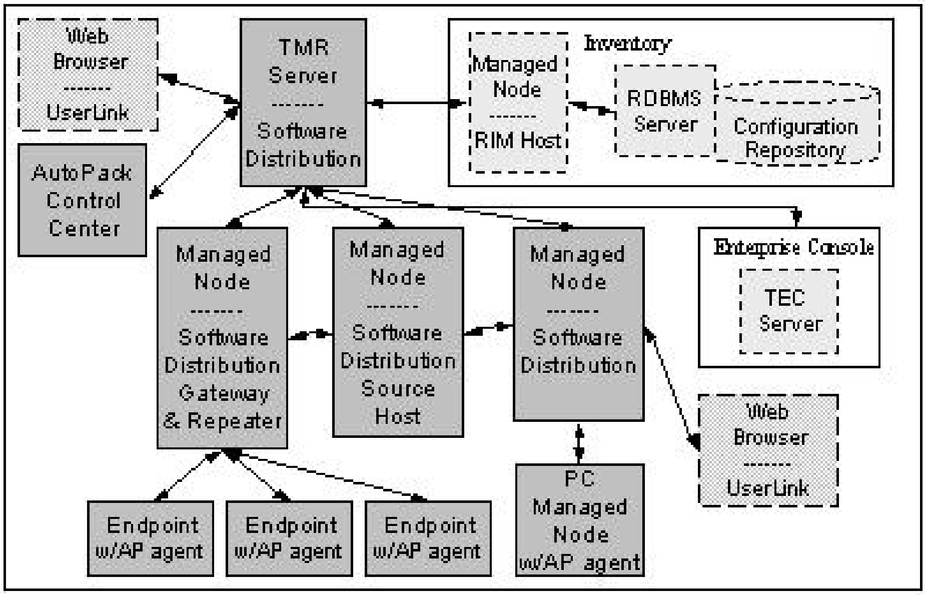
Web Browser ------- UserLink
TMR Server ------- Software Distribution
Inventory
Managed Node ------- RIM Host
RDBMS Server
Configuration Repository
AutoPack Control Center
Managed Node ------- Software Distribution Gateway & Repeater
Managed Node ------- Software Distribution Source Host
Managed Node ------- Software Distribution
Enterprise Console
TEC Server
Web Browser ------- UserLink
PC Managed Node w/AP agent
Endpoint w/AP agent
Endpoint w/AP agent
Endpoint w/AP agent

Figure B-2. Tivoli Software Distribution Logical Structure

From a component perspective, Software Distribution functions are accomplished as follows:

- The Software Distribution server application is installed on the TMR Server and on managed nodes that might be used to execute Software Distribution administration commands. This application includes the administrative commands.

- The TMR Server manages profile distributions to managed nodes and gateways. As part of the Management Framework Multiplexed Distribution (MDist) services, the TMR server also functions as a repeater.

- The Software Distribution Gateway application is installed on managed nodes that act as gateways for endpoints. The gateway provides a collection point for communications between the assigned endpoints and the TMR server. The gateway also functions as an MDist repeater.

- Software Distribution uses the services of Management Framework MDist repeaters to support more efficient distribution of large amounts of data to multiple target machines. In addition to the TMR server(s) and gateways, managed nodes can be explicitly configured as repeaters.

- A Software Distribution Source Host is a machine designated to hold the original copy of files that are part of a distribution.

- The Software Distribution TEC Integration feature allows Software Distribution operations to be reported as events to the Tivoli Enterprise Console application.

- The Software Distribution Historical Database feature provides an interface to the Tivoli Inventory application. This feature can automatically update the Inventory Configuration Repository with information from Software Distribution operations.

- The Software Distribution Extension API feature includes files that can be used with the Tivoli Application Development Environment (ADE) to customize Software Distribution.

- The Software Distribution AutoPack Control Center is a Windows application that can be used to create AutoPack distributions for Windows-based targets. AutoPack Control Center is installed on a “prep” machine that is used for model software installations. Using a process consisting of pre-installation snapshot, install, post-installation snapshot, and build, AutoPack Control Center automates the process of building software distributions.

- The Software Distribution AutoPack Agent is an application used on endpoints and PC managed nodes in conjunction with AutoPack distributions. The AutoPack Agent automates configuration tasks on the targets of distributions.

- The web-based interface provides a way for authorized end users to initiate the deployment of Software Distribution profiles that download software and data to their machine. The web interface is referred to as UserLink.

Software Distribution uses the Framework profile and subscriber concepts to accomplish its tasks. The following provides a very high level view of some of the information flow:

- Administrators define profiles that are containers for Software Distribution-specific information. Profile managers are defined groups of profiles to be distributed to designated target machines known as subscribers.

- Administrators deploy File Package distributions by creating or loading installation files on a Source Host; defining a File Package profile that specifies the Source Host, source file names, destination directory path, and any configuration programs to be run before or after the files are copied; and distributing the File Package profile to subscribers.

- Administrators deploy AutoPack distributions by creating an AutoPack file using the AutoPack Control Center, defining an AutoPack profile that includes the Source Host and path to the AutoPack file, and distributing the AutoPack profile to subscribers.

- Subscribers can be managed nodes, endpoints, or PC managed nodes. When using AutoPacks, the AutoPack Agent must be explicitly installed on managed nodes and PC managed nodes. Endpoints are automatically configured with the AutoPack Agent during endpoint creation.

B.6.2 Server and Client Component Platforms

Software Distribution components for managed nodes, endpoints, and PC managed nodes are supported on a variety of host operating systems. The following table summarizes this support by component type.

Tivoli Component Type	Supported Platforms for Software Distribution
Managed Node	AIX 4.x, Digital UNIX, HP-UX, NCR UNIX, Sequent DYNIX/ptx, SCO UnixWare, SGI IRIX, Solaris, Solaris x86, SunOS, Windows NT
Endpoint	AIX 4.x, AS/400, Digital UNIX, DG/UX, HP-UX, NCR UNIX, NetWare, OS/2, Pyramid MIServer, Sequent DYNIX/ptx, SCO UnixWare, SGI IRIX, Solaris, Solaris x86, SunOS, Windows 3.x, Windows 95, Windows NT
PC Managed Node (PC Agent)	NetWare, OS/2, Windows 3.x, Windows 95, Windows NT
AutoPack Control Center and Agent	Windows 3.x, Windows 9x, Windows NT

Please note that this support information is highly release dependent. The information provided here is based on details from the *TME 10 Software Distribution User's Guide, Version 3.6,* and the *TME 10 Software Distribution Release Notes, Version 3.6.*

In order to provide appropriate, focused support for the STIG audience, the discussion for Software Distribution component platforms in this document is limited to UNIX and Windows (NT-based) platforms.

B.6.3 General Security Considerations

Because Software Distribution makes use of the infrastructure provided by the Management Framework, many security considerations for Software Distribution are addressed by controls required for the Framework. Compliance with all of the requirements in the Framework section of this document is assumed as the basis for the considerations described here.

The unique considerations for Software Distribution include:

- Access to Software Distribution functions is controlled through the assignment of Tivoli authorization roles to Tivoli users. Without proper assignment of these roles, the integrity of Software Distribution operations could be compromised.

- As with most software products, Software Distribution is composed of program and data files for which proper access controls are essential. Although most of the Software Distribution product files will reside in directories subject to access controls required for the Framework, there are some Software Distribution directories and files that require specific access controls.

- As the primary function of Software Distribution involves moving software files and configuration files from source to target locations, those files can be vulnerable to alteration before or during distribution. To ensure the integrity of those files, the source and any intermediate locations require appropriate access controls.

- If the Software Distribution web interface features are used, user transactions are processed through HTML pages. The integrity of the HTML files must be assured to maintain transaction security.

- The Software Distribution Extension API product provides elements that can be used to customize Software Distribution using the TME 10 Application Development Environment (ADE). If implemented incorrectly, customization code could introduce vulnerabilities.

The actions that are required to address these considerations are addressed in the following section of this document.

B.6.4 Specific Security Considerations

This section describes the specific considerations and required actions to help ensure that the Software Distribution components are implemented in a secure fashion. As noted in the previous section, compliance with the requirements in the Framework section of this document is assumed as the prerequisite for the information here.

Department of Defense (DOD) Instruction 8500.2, Information Assurance (IA) Implementation, documents IA controls that apply to DOD information systems. The following integrity and confidentiality controls from *DODI 8500.2* are the basis for the requirements in this section: Access for Need-to-Know; Audit Trail, Monitoring, Analysis, and Reporting; Changes to Data; Individual Identification and Authentication; Least Privilege; and Privileged Account Control.

The following subsections describe the considerations and requirements organized by major Software Distribution component. When reviewing these requirements, the following must be noted:

- The specific file and directory names referenced in the requirements and specified in the appendix reflect defaults indicated in vendor documentation. If a site or organization deploying Tivoli chooses other names, the requirements apply to the site-specific names.

- The structure of UNIX permissions can make it difficult to restrict file access to multiple groups of users. The approach taken in this section is to allow UNIX "world" access in cases where non-update (e.g., read or execute) access permission is required for users who may be members of multiple groups and there is no specific need for the files to be inaccessible to other users.

B.6.4.1 Tivoli Authorization Roles

Authorization roles defined within the Management Framework provide role based access control over functions in the Tivoli products. A Tivoli administrator must be assigned the associated roles to be permitted to perform specific product functions.

Authorization roles such as senior or super enable a broad range of capabilities in multiple products. Other roles such as SWDist_end_user enable only limited functions in one product. The following table lists the authorization roles and the associated functions within Software Distribution.

Role	Capability
admin	AutoPack and File Package: calculate size AutoPack and File Package profiles: distribute, edit, view File Package definitions: edit, export, import Software Distribution profile subscribers: add, remove UserLink web functions - Software Distribution profile: distribute
install_product	Software Distribution products: install, upgrade
senior	AutoPack and File Package: calculate size AutoPack and File Package profiles: clone, create, delete, distribute, edit, view -- including AutoPack agent File Package definitions: create from AMP, edit, export, import Historical database feature: enable, disable Software Distribution profile subscribers: add, remove Software Distribution queries: create, edit, execute **UserLink** web functions - Software Distribution profile: distribute

Role	Capability
super	AutoPack and File Package: calculate size AutoPack and File Package profiles: clone, create, delete, distribute, edit, view -- including AutoPack agent File Package definitions: create from AMP, edit, export, import Historical database feature: enable, disable Software Distribution products: install, remove, upgrade Software Distribution profile subscribers: add, remove Software Distribution queries: create, edit, execute UserLink web functions - Software Distribution profile: distribute
SWDist_end_user	UserLink web functions - Software Distribution profile: distribute
user	AutoPack and File Package profiles: view File Package definitions: export UserLink web functions - Software Distribution profile: distribute

The assignment of Tivoli authorization roles is restricted and tracked in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know, Least Privilege, and Privileged Account Control*. Roles must be assigned only to users who require the associated privileges to perform designated job functions.

- *(TSD.0001: CAT II) The IAO will ensure that Tivoli authorization roles that control access to functions of Software Distribution are assigned only to appropriate, authorized users and that the assignment of roles is documented.*

B.6.4.2 Software Distribution Server Considerations

The Software Distribution server software is installed on the TMR server and on managed nodes from which profiles will be created and distributed or other Software Distribution functions will be executed. The Software Distribution server software includes executable and configuration files that are installed on a host.

The Software Distribution server software provides the programs through which File Packages and AutoPacks are deployed. If access to the server software is lost or the files are corrupted or deleted, the primary function of Software Distribution can become degraded or unavailable. If access is not controlled, files being deployed might be altered during distribution to include a Trojan Horse program or other virus.

There are some basic and some element-specific considerations relative to the security of the Software Distribution server. The basic considerations are described next. Considerations specific to File Packages and AutoPacks are described following that.

B.6.4.2.1 Basic Considerations

Three basic considerations relative to the security of the Software Distribution server are:

- The host OS must be configured securely.
- The integrity of the Software Distribution server software must be maintained.
- Information posted to the Software Distribution notice group must be monitored to ensure proper system operation.

The host OS security consideration applies generically to the Software Distribution server host. It is addressed through the OS security requirements described in the applicable DISA STIG. Specifically, the Software Distribution server host must be compliant with the STIG that covers the OS used on the Software Distribution server.

The security consideration for the integrity of the Software Distribution server software applies to the executable and configuration files that are installed with the server. If the files are altered or deleted, access to the Software Distribution server could become degraded or unavailable. It might also be possible for Software Distribution data to be deleted, corrupted, or disclosed to unauthorized users.

The Software Distribution server software is installed in a directory structure that is subordinate to a logical directory known as the BINDIR directory. The physical implementation of the BINDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. Some of the Software Distribution server software files are installed in directories that are shared with the Management Framework. These directories are protected through the specific Management Framework requirements elsewhere in this document. Requirements for the Software Distribution-specific directories are addressed here.

The assignment of directory and file access permissions for the Software Distribution server software is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as accounts used by Software Distribution users, are allowed execute and read access.

- *(TSD.0002: CAT II) The SA will ensure that access to the Software Distribution server directories and files is restricted in accordance with the permissions in Appendix B.8.*

The security consideration for monitoring notices in the Software Distribution notice group is related to the need to monitor events that may be related to malicious activity. Notices are posted to the Software Distribution notice group for events and errors related to File Package operations and AutoPack profile distributions.

The Tivoli Notification facility is a tool used to assist with monitoring and audit trail creation in accordance with the *DODI 8500.2, IA Controls for Audit Trail, Monitoring, Analysis and Reporting*. Tivoli administrators need to review Software Distribution notices to validate that unusual or inappropriate activity is not indicative of a security compromise.

- *(TSD.0004: CAT II) The Tivoli administrator will ensure that Software Distribution is configured to send notices to the Software Distribution notice group for logging File Package and AutoPack operations.*

- *(TSD.0005: CAT II) The Tivoli administrator will ensure that at least one Tivoli administrator is subscribed to receive Software Distribution notices.*

B.6.4.2.2 File Package Property Considerations

File Packages are defined on the Software Distribution server in terms of properties that specify information about the software or data being distributed. File Package definition files include the following categories of properties:

- Source directories and files – the names of directories and files to be distributed
- Nested file packages – the names of file packages to be included in the one being defined
- General options – various options such as error actions, source directory depth, target directory naming, and data compression
- Log and notification options
- Platform-specific options – various options that apply to certain OS platforms and special configuration actions

An additional File Package property is defined as part of the profile on the Software Distribution server. The source host for the File Package specifies the name of the host holding the data to be distributed.

A very limited number of File Package properties have security considerations. These arise primarily from the potential for unauthorized or improper values to allow the integrity of the distribution process or the target host to be compromised.

In addition to the source host property, the following File Package properties, listed according to the keywords or options used in File Package definition files, have security considerations:

Property \ Property Keyword	Description
[File and directory options: g, G, m, M, u, U]	File and directory specifications in File Package definitions can include options that specify GID, chmod, and UID settings for files and directories
default_dir_mode	The default mode (permission) for all directories distributed in the package
default_file_mode	The default mode (permission) for all files distributed in the package
post_notice	Option to post notices to the Software Distribution notice group for distribution, commit, or removal operations
unix_default_dir_gid	The default GID for all UNIX directories distributed in the package
unix_default_dir_uid	The default UID for all UNIX directories distributed in the package
unix_default_file_gid	The default GID for all UNIX files distributed in the package
unix_default_file_uid	The default UID for all UNIX files distributed in the package

The specific security issues that can result from unauthorized or improper values include:

- Changing the source host to a different, valid host name could cause incorrect or invalid files to be distributed or the distribution to fail.

- Changing the source host to an invalid (i.e., non-existent) host name would cause the distribution to fail.

- Specifying file or directory modes that are too permissive would cause files to be created on target hosts with inadequate access control.

- Failing to specify a logging option would result in the loss of an audit trail that could be significant to detecting an intrusion.

- Specifying UID or GID values that are not assigned to users or groups would cause files to be created on target hosts with incorrect or invalid ownership attributes and could result in inadequate access control.

The values of certain File Package properties are restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. The properties must be set so that access to distributed files is assigned only to users who require the access to perform designated job functions.

These security issues for File Package properties are addressed through the following requirements.

- *(TSD.0006: CAT II) The Tivoli administrator will ensure that each File Package definition specifies one of the documented source hosts for the TMR.*

- *(TSD.0007: CAT II) The Tivoli administrator will ensure that any directory or file mode specifications in File Package definitions do not result in the creation of world writable directories or files.*

- *(TSD.0008: CAT II) The Tivoli administrator will ensure that any UID or GID specifications in File Package definitions do not result in the creation of directories or files that are assigned to undefined users or groups.*

The issue of logging is addressed by the notice group requirements in the basic considerations section.

B.6.4.2.3 AutoPack Property Considerations

AutoPacks are defined on the AutoPack Control Center in terms of properties that specify information about the software or data being distributed. The general categories of AutoPack properties include:

- General options – the destination drive and working directory on the distribution's target machine
- File mode options – access permissions for distributed files
- Log and notification options
- Distribution options - various options such as error actions, data compression, and distribution mode (mandatory \ optional)

An additional AutoPack property is defined as part of the AutoPack profile on the Software Distribution server. The host name and path for the AutoPack file provide the source host specification for an AutoPack.

A very limited number of AutoPack properties have security considerations. These arise primarily from the potential for unauthorized or improper values to allow the integrity of the distribution process or the target host to be compromised.

In addition to the source host property, the following AutoPack properties, listed according to the keywords or options used in AutoPack definition (.DEF) files, have security considerations:

Property \ Property Keyword	**Description**
[File and directory options: m, M]	File and directory specifications in AutoPack definitions can include options that specify chmod settings for files and directories.
default_dir_mode	The default mode (permission) for all directories distributed in the package
default_file_mode	The default mode (permission) for all files distributed in the package
post_notice	Option to post notices to the Software Distribution notice group for distribution, commit, or removal operations

The specific security issues that can result from unauthorized or improper values include:

- Changing the source host to a different, valid host name could cause incorrect or invalid files to be distributed or the distribution to fail.

- Changing the source host to an invalid (i.e., non-existent) host name would cause the distribution to fail.

- Specifying file or directory modes that are too permissive would cause files to be created on target hosts with inadequate access control.

- Failing to specify a logging option would result in the loss of an audit trail that could be significant to detecting an intrusion.

The values of certain AutoPack properties are restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. The properties must be set so that access to distributed files is assigned only to users who require the access to perform designated job functions.

The security issues for AutoPack properties are addressed through the following requirements.

- *(TSD.0009: CAT II) The Tivoli administrator will ensure that each AutoPack definition specifies one of the documented source hosts for the TMR.*

- *(TSD.0010: CAT II) The Tivoli administrator will ensure that any directory or file mode specifications in AutoPack definitions do not result in the creation of directories or files with write access to the Everyone group.*

The issue of logging is addressed by the notice group requirements in the basic considerations section.

B.6.4.3 Software Distribution Source Host and Repeater Considerations

Software Distribution automates the delivery of software and data to a large number of target hosts. This capability is of high importance to an enterprise when used for essential deployments such as the distribution of mission critical software updates, software security patches, or anti-virus definition files. In these and other cases, malicious modification of the source data can have negative impacts ranging from a failure in distribution to the distribution of malicious code that could cause loss or disclosure of data on multiple target hosts. Preserving the integrity of the source data is therefore a key requirement.

There are two logical locations within the Software Distribution infrastructure where the source data for a distribution resides for an extended period. These locations are the source host and any host acting as a repeater. As noted previously, a source host holds the original data to be distributed. Repeater hosts hold a copy of the data and are typically implemented where target hosts are at remote locations connected through a wide area network. Note that although the TMR server and Software Distribution Gateways act as repeaters by default, the discussion in this section applies to hosts that are defined explicitly as repeaters. The same issue of protecting the data to be distributed applies to both source hosts and repeater hosts.

The security considerations for source and repeater hosts include the following:

- The host OS must be configured securely.
- The integrity of the data to be distributed must be maintained.

The host OS security consideration applies generically to source and repeater hosts. It is addressed through the OS security requirements described in the applicable DISA STIG. Specifically, source hosts and repeater hosts must be compliant with the STIG that covers the OS used on their platform.

The security consideration for the integrity of the data to be distributed applies to the executable and data files that are part of the File Package or AutoPack being distributed. If the files are altered or deleted, the distribution could fail. It might also be possible for software to be corrupted such that data on target hosts might be deleted or disclosed to unauthorized users.

Limiting host access is a strategy that can be used to prevent data being distributed from being altered or deleted. In practical terms this means that only System Administrators or Tivoli administrators have logical access to hosts designated as source hosts or repeater hosts.

The assignment of system access permissions for the Software Distribution source hosts and repeater hosts is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Least Privilege*. Access permissions must be assigned only to users who require the associated access to perform designated job functions.

- *(TSD.0011: CAT II) The Tivoli administrator will maintain documentation including a list of authorized source hosts and repeater hosts for each TMR.*

- *(TSD.0012: CAT II) On TMR source hosts and repeater hosts running an MS Windows OS, the SA will ensure that the "Access this computer from the network" and "Log on locally" user rights are granted only to system and Tivoli administrators.*

- *(TSD.0013: CAT II) On TMR source hosts and repeater hosts running a UNIX OS, the SA will limit defined user accounts to system and Tivoli administrators.*

B.6.4.4 Software Distribution Gateway Considerations

The Software Distribution Gateway software is installed on managed nodes that are Tivoli gateways that perform Software Distribution functions on endpoints. The Software Distribution Gateway software includes executable and configuration files that are installed on a host.

The Software Distribution Gateway software provides the programs through which data is distributed to endpoints. The gateway uses the MDist repeater feature to provide a cache for more efficient delivery of data to endpoints. If access to the Gateway software is lost or the files are corrupted or deleted, the distribution function can become degraded or unavailable for the set of assigned endpoints. If access is not controlled, data being distributed could be deleted, corrupted, or disclosed to unauthorized users.

There are two considerations relative to the security of the Software Distribution Gateway:

- The host OS must be configured securely.
- The integrity of the Software Distribution Gateway software must be maintained.

The host OS security consideration applies generically to the Software Distribution Gateway host. It is addressed through the OS security requirements described in the applicable DISA STIG. Specifically, the Software Distribution Gateway host must be compliant with the STIG that covers the OS used on the Software Distribution Gateway.

The security consideration for the integrity of the Software Distribution Gateway software applies to the executable and configuration files that are installed with the Gateway. If the files are altered or deleted, the distribution function of the Software Distribution Gateway could become degraded or unavailable for some endpoints. It might also be possible for data being distributed to some endpoints to be deleted, corrupted, or disclosed to unauthorized users.

The Software Distribution Gateway software is installed in a directory structure that is subordinate to a logical directory known as the BINDIR directory. The physical implementation of the BINDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. Another variable, INTERP, is also used in the path name for the directory structure. INTERP represents the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems. Some of the Software Distribution Gateway software files are installed in directories that are shared with the Management Framework. These directories are protected through the specific Management

Framework requirements elsewhere in this document. Requirements for the Software Distribution Gateway-specific directories are addressed here.

The assignment of directory and file access permissions for the Software Distribution Gateway software is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as accounts used by Software Distribution users, are allowed execute and read access.

- *(TSD.0014: CAT II) The SA will ensure that access to the Software Distribution Gateway server directories and files is restricted in accordance with the permissions in Appendix B.8.*

B.6.4.5 Software Distribution TEC Integration Considerations

The Software Distribution TEC Integration component enables Software Distribution to send events to the Tivoli Enterprise Console (TEC) application. Through TEC Integration, distribution, commit, and removal Software Distribution operations generate events that are sent to the TEC server.

The Software Distribution TEC Integration component provides two configuration files:

- The tecad_sd.conf file is installed on a TMR sever where the Software Distribution server is installed. It defines operating parameters such as the TEC server and the port on which the TEC server listens. It can also specify event filtering instructions.

- The tecad_sd.baroc file is installed on a TEC server host. It provides event class definitions that are compiled into the event server's rule base.

The security consideration relative to the Software Distribution TEC Integration component is for the integrity of the configuration files. If the tecad_sd.conf file were altered or deleted, events for Software Distribution operations might not reach a TEC server. If the tecad_sd.baroc file were altered or deleted, installation or maintenance of an event server's rule base might be affected, potentially leading to missing or improperly classified events.

The tecad_sd.conf file is located on a TMR server in a directory below a logical directory structure known as the DBDIR directory. The physical implementation of the DBDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. The vendor default path to the file is $DBDIR/.sd_tec/tecad_sd.conf.

The tecad_sd.baroc file is located on a TEC server in a directory below a logical directory structure known as the DBDIR directory. The physical implementation of the DBDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. The vendor default path to the file is $DBDIR/TME/COURIER/tecad_sd.baroc.

The assignment of file access permissions for the tecad_sd.conf and tecad_sd.baroc files is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as accounts used by non-administrative Software Distribution users, are not allowed any access.

- *(TSD.0016: CAT II) The SA will ensure that access to the Software Distribution TEC integration configuration files is restricted in accordance with the permissions in Appendix B.8.*

B.6.4.6 Software Distribution Extension API Considerations

The Software Distribution Extension API component provides elements that can be used in conjunction with the Tivoli Application Development Environment (ADE) to customize the dialogs and behavior of Software Distribution.

The elements provided by the Extension API component include header files, libraries, and Interface Definition Language (IDL) files that can be used by administrators performing product customization.

There are two security considerations relative to the Software Distribution Extension API component. The first consideration is for the integrity of the API files. If these files were altered or deleted, a subsequent use of the files to create or alter a Tivoli application could fail or could result in compromised code. The second consideration is for possible disclosure to individuals with malicious intent. Because these files provide explicit documentation of the internal data structure and interfaces of some Tivoli components, a malicious user might be able to use this information to prepare a denial of service or privilege elevation attack on the Tivoli infrastructure. While the success of such an attack is likely to require some additional vulnerability to be present, restricting access to the API files would help to increase the difficulty of constructing the attack.

The Extension API components are installed in a directory chosen during product installation. The vendor default directory is /usr/local/Tivoli/include which is shared with other Tivoli components. That directory is protected through the specific Management Framework requirements elsewhere in this document. For sites that install the Software Distribution Extension API components in a separate directory, the access control requirement is addressed here.

The assignment of directory access permissions for the Software Distribution Extension API components is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as accounts used by non-administrative Software Distribution users, are not allowed any access.

- *(TSD.0018: CAT II) The SA will ensure that, if the Software Distribution Extension API components are installed in a separate directory, access to the directories and files is restricted in accordance with the permissions in Appendix B.8.*

B.6.4.7 Software Distribution Historical Database Considerations

The Software Distribution Historical Database component provides an interface to Tivoli Inventory so that the results of Software Distribution operations can be automatically populated in the Inventory configuration repository. Data about install and remove operations is maintained for File Packages and AutoPacks. Through the Framework query facility, the data can be analyzed using pre-defined or ad hoc queries.

The Historical Database component includes programs that enable the capture and storage of the data and a script (swdist_queries.sh) that creates a query library with a pre-defined query. Data is stored in the INSTALLED_SW_COMPONENT, SOFTWARE_COMPONENT, and SOFTWARE_FILEPACK tables in the database that houses the Inventory configuration repository.

The security consideration relative to the Software Distribution Historical Database component is for the integrity of the programs, query creation script, and configuration repository data. If the programs were altered or deleted, data about distribution operations could be lost, corrupted, or disclosed to unauthorized users. If the query creation script were altered or deleted, the initial or subsequent attempts to build the query library could fail or the query could be built to include malicious code that could delete, corrupt, or disclose data. If the configuration repository data were altered or deleted, information on system configurations could be lost, corrupted, or disclosed to unauthorized users.

The Historical Database programs are installed in a directory that is protected by other requirements in this section of this document. The configuration repository data is protected by requirements in the Tivoli Inventory section of this document. The query creation script is installed in a directory structure that is subordinate to a logical directory known as the BINDIR directory. The physical implementation of the BINDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. Requirements for the directory holding the query creation script are addressed here.

The assignment of directory access permissions for the Historical Database query creation script is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as accounts used by non-administrative Software Distribution users, are not allowed any access.

- *(TSD.0020: CAT II) The SA will ensure that access to the Software Distribution Historical Database directories and files is restricted in accordance with the permissions in Appendix B.8.*

B.6.4.8 AutoPack Control Center Considerations

AutoPack Control Center is a Windows application used to automate the process of building Windows software distributions. AutoPack Control Center uses a process consisting of pre-installation snapshot, application install, post-installation snapshot, and build to create an AutoPack file that is used in distributions. The AutoPack agent, described in the next section, receives and unpacks an AutoPack profile distribution that contains the AutoPack file.

AutoPack Control Center is installed on a Windows-based machine that is called a prep machine. The vendor recommends that this machine have as few applications as possible installed. The vendor further recommends that the prep machine **not** be a Tivoli managed node, endpoint, or PC managed node. These recommendations help to reduce problems with the inclusion of extraneous files and file contention.

To accomplish its snapshot function, AutoPack Control Center scans directories, files, and Registry entries on a Windows host. In order to have the necessary access to do this, the Tivoli administrator using the application must be a member of the Administrators group for that host.

The AutoPack Control Center application consists of program files installed on a prep machine. It creates a number of output files in a user-specified working directory. There are temporary files used during the build process and result files that may be used to alter or rebuild an AutoPack distribution at a later time. The result files include system change (.CHG), definition (.DEF), log (.ERR), system file list (.REP), and AutoPack (.PAK) files. An AutoPack file is the source file for an AutoPack distribution.

There are three considerations relative to the security of the AutoPack Control Center:

- The host OS must be configured securely.
- The integrity of the AutoPack Control Center software must be maintained.
- The integrity of the AutoPack Control Center output files must be maintained.

The host OS security consideration applies generically to the AutoPack Control Center prep machine. It is addressed through the OS security requirements described in the applicable DISA STIG. Specifically, the AutoPack Control Center host must be compliant with the STIG that covers the OS used on the prep machine.

The security consideration for the integrity of the AutoPack Control Center software applies to the executable files that are installed with the AutoPack Control Center application. If the files are altered or deleted, the snapshot or build functions may work improperly or not all. It might also be possible for data in AutoPack files to be deleted or corrupted, with the potential for distribution of malicious code that could cause loss or disclosure of data on multiple target hosts.

The security consideration for the integrity of the AutoPack Control Center output files applies to the temporary and result files that are created by the application. If the files are altered or deleted, the build function may work improperly or not all. Deletion or malicious modification of an AutoPack file can have negative impacts ranging from a failure in distribution to the distribution of malicious code that could cause loss or disclosure of data on multiple target hosts.

Recognizing the vendor's recommendations for the configuration of the AutoPack prep machine, the requirements to maintain the integrity of the AutoPack Control Center software and output files, and the need for the Tivoli administrator to be a member of the Administrators group on the machine, the best strategy to maintain security is a strict restriction of access to the prep machine. In implementation terms this means that only System Administrators or Tivoli administrators have logical access to the prep machine.

The assignment of system access permissions for the AutoPack prep machine is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Least Privilege*. Access permissions must be assigned only to users who require the associated access to perform designated job functions.

- *(TSD.0022: CAT II) The Tivoli administrator will maintain documentation including a list of AutoPack Control Center hosts for each TMR.*

- *(TSD.0023: CAT II) The SA will ensure that on systems with the AutoPack Control Center installed the "Access this computer from the network" and "Log on locally" user rights are granted only to system and Tivoli administrators.*

B.6.4.9 AutoPack Agent Considerations

The AutoPack Agent is responsible for receiving and unpacking an AutoPack profile on a target host. It works in conjunction with the endpoint daemon (lcfd) on endpoints and the PC agent on PC managed nodes to process an AutoPack distribution.

On endpoints the AutoPack Agent is automatically configured during installation. For PC managed nodes the AutoPack Agent must be explicitly installed through a profile distribution. The primary component of the AutoPack Agent is an executable file, wsyschg.exe, which is installed by default in the Windows SystemRoot (e.g., C:\WINNT) directory.

The security consideration relative to the AutoPack Agent is for the integrity of the AutoPack Agent program. If the program were altered or deleted, AutoPack distributions could fail or the data being distributed could be corrupted or disclosed to unauthorized users. In extreme circumstances the target host could be compromised, resulting in loss or disclosure of data.

The assignment of access permissions for the wsyschg.exe file is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Accounts used by the Software Distribution process are allowed read and execute access.

- *(TSD.0024: CAT II) The SA will ensure that access to the Software Distribution AutoPack Agent file is restricted in accordance with the permissions in Appendix B.8.*

B.6.4.10 Web Interface \ UserLink Application Considerations

Software Distribution provides a web interface that allows authorized users to initiate the deployment of Software Distribution profiles that download software and data to their machine. This web-based interface for users is called UserLink.

UserLink for Software Distribution is available for UNIX and Windows endpoints. A Tivoli user employs a Java-enabled web browser on an endpoint to initiate File Package and AutoPack distributions for which that endpoint is subscribed. This facility has particular value for mobile machines that connect sporadically to the TMR environment.

The UserLink facility is implemented through a web page file on the endpoint and web page and executable files on the TMR server where the Software Distribution server is installed. Users are permitted specific access to the UserLink feature through the Tivoli SWDist_end_user role. Users assigned the senior, super, user, and admin roles have implicit access to UserLink in addition to their other privileges. When a user selects the web page link to connect to the TMR server, a prompt is displayed for the user name and password for the account assigned the Tivoli role.

There are four considerations relative to the security of the Software Distribution-specific web interface:

- The integrity of the Software Distribution server-based web components used in the UserLink facility must be maintained.
- The proper assignment of the Tivoli authorization role for the UserLink function must be managed.
- The confidentiality of the user authentication data that passes over the network connection between the endpoint client and the TMR server must be maintained.
- The integrity of the endpoint-based UserLink web page must be maintained.

The security consideration for the integrity of the Software Distribution server-based web page and executable files applies to files installed on the Software Distribution server host. If the files are altered or deleted, the ability to initiate distributions through UserLink could become degraded or unavailable. It might also be possible to corrupt a distribution resulting in data on the target endpoint being deleted, corrupted, or disclosed to unauthorized users.

The Software Distribution server-based web page and executable files are installed in directories logically owned by the Software Distribution Server. Access to these directories is controlled through the requirements documented in *Section B.6.3.2, Software Distribution Server Considerations*.

The security consideration for the proper assignment of the Tivoli authorization role applies to the SWDist_end_user role. If the role were not assigned when required, the UserLink functions would be unavailable to the authorized user. If the role were assigned to an unauthorized user, that user might be able to alter a machine's configuration at an inappropriate time. Access to this role is addressed through the requirements documented in *Section B.6.3.1, Tivoli Authorization Roles*.

The security consideration for the confidentiality of the user authentication data applies to the transmission of the Tivoli user password from the web browser on the endpoint client to the TMR server. The HTTP protocol between the client and server does not provide encryption. This makes the password vulnerable to capture during transmission over the network.

Password data is encrypted in accordance with the *DODI 8500.2, IA Controls for Individual Identification and Authentication*. Passwords must be encrypted both for storage and for transmission.

- *(TSD.0025: CAT II) The SA will ensure that network connections between endpoints using the UserLink facility and the TMR server are encrypted.*

The security consideration for the integrity of the UserLink web page applies to the HTML file that is downloaded during endpoint configuration. Altering this file could cause the web page not to load in the browser, to load improperly or link to a counterfeit server, or to include data that might exploit vulnerabilities on the Software Distribution server.

The UserLink web page file is installed on an endpoint by default in a directory named DRIVE:/etc/Tivoli/LANG, where DRIVE is the drive on which the Management Framework agent is installed and LANG represents the code for the language defined for the endpoint.

The assignment of file access permissions for the UserLink web page must be restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to users who require the associated access to perform designated job functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged user accounts are allowed read access.

- *(TSD.0026: CAT II) The SA will ensure that access to the Software Distribution UserLink web page file is restricted in accordance with the permissions in Appendix B.8.*

B.7 IBM Tivoli Monitoring for Business Integration

B.7.1 General Overview

The IBM Tivoli Monitoring for Business Integration software product, formerly referred to as Tivoli Manager for MQSeries, is designed to provide centralized management, administration and monitoring of WebSphereMQ resources and activity in a Tivoli Enterprise. Because the Tivoli Monitoring for Business Integration is integrated with multiple Tivoli software products, such benefits as authorization roles, consolidated event displays, event threshold monitoring, configuration backup and recovery and software distribution are realized.

In order for the Tivoli Monitoring for Business Integration software to be used to manage WebSphereMQ, it must be installed on the Tivoli Management Region server, the TEC Event server, each endpoint running WebSphereMQ resources (that are to be managed), and all gateways (for endpoint support). Automated tasks are provided by the product, which enable administrators to define, change, start, stop and delete WebSphereMQ queue managers, queues, channels, and other resources.

In addition, the product provides event adapters specifically designed for the monitoring of WebSphereMQ activity, performance and potential problems. These event adapters collect event information from queue manager’s event queues, reformat the information and send the data to the TEC event server for display and potential response. Automated responses may result by the TEC based on rule sets established for the different events.

- *(TMQ.0001: CAT II) The IAO will ensure that the TIVOLI MONITORING FOR BUSINESS INTEGRATION product is installed in the Tivoli Enterprise either using the TMF or authorized scripts.*

- *(TMQ.0002: CAT I) The IAO will ensure that unsupported TIVOLI MONITORING FOR BUSINESS INTEGRATION software is removed or upgraded prior to a vendor dropping support.*

- *(TMQ.0003: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading DBMS systems prior to the date the vendor drops security patch support.*

- *(TMQ.0004: CAT II) The IAO will ensure that the creation, distribution and implementation of TME endpoint adapters, TME managed node adapters and non-TME adapters are restricted to TMR administrators and authorized personnel.*

- *(TMQ.0005: CAT II) The IAO will ensure that the Tivoli Monitoring for Business Integration adapter files are restricted from unauthorized update and access by unauthorized processes, scripts, or users.*

B.7.2 General Considerations

The following subsections address areas, which impact the performance of the Tivoli Monitoring for Business Integration product.

B.7.2.1 WebSphereMQ Management Domain(s)

In a Tivoli Enterprise, where the Tivoli Monitoring for Business Integration software is used, all WebSphereMQ resources that are to be managed are defined and contained in a management domain. At least one management domain must exist in a Tivoli Enterprise. When a management domain is created, all the profile managers, task libraries, and other objects that belong to the management domain are loaded into the domain. The queue managers are not initially recognized. To discover the queue managers for the management domain, a discovery function is run which will create a queue manager object for each queue manager that is discovered. The label for a queue manager icon is set to qmgrname@hostname.

Additional management domains may be created if a larger number of WebSphereMQ resources exist in an enterprise. If additional management domains are created, multiple administrators may be required to monitor and manage the resources. In addition, because each management domain is implemented as a TME policy region, not only will each domain contain its own set of WebSphereMQ resources, profile managers, monitors, task libraries but may also have different authorizations established. Each management domain may be managed from a different Tivoli Desktop.

Before a queue manager or a queue manager icon is deleted from a Desktop, the queue manager must be unsubscribed from all profile managers to which it is subscribed. Doing this ensures that the queue manager is deleted from the Tivoli database. If an endpoint is to be deleted or moved to another gateway, the queue manager icon must be deleted and then the discovery function rerun.

- *(TMQ.0006: CAT II) The IAO will ensure that the WebSphereMQ Management Domains are restricted from unauthorized update and access.*

B.7.2.2 Remote Administration

The Tivoli Monitoring for Business Integration remote administration facility enables WebSphereMQ commands to be issued to queue managers that do not reside on Tivoli managed nodes or endpoints. Commands may be entered either through the Tivoli Desktop or the Tivoli Business Systems Manager workstation.

- *(TMQ.0007: CAT II) The IAO will ensure that the Tivoli Desktop(s) are located in a restricted location in accordance with DODI 8500.2.*

- *(TMQ.0008: CAT II) The IAO will ensure that use of the Tivoli Desktop(s) will be restricted to authorized users and appropriate roles are assigned in accordance with DODI 8500.2.*

- *(TMQ.0009: CAT II) The IAO will ensure that encrypted communication is used between the Tivoli Desktop(s), the Tivoli Business Systems Manager workstation(s), and the supporting servers.*

B.7.2.3 Application Proxy

The Application Proxy is an extension of the Tivoli Framework that provides a common set of services that Tivoli Monitoring for Business Integration uses. The Application Proxy is installed on the Tivoli server, Event server, each managed node running WebSphereMQ resources, and all gateways.

- *(TMQ.0010: CAT II) The IAO will ensure that the Application Proxy is located on a platform that is secured in accordance with this STIG, the UNIX STIG, the Windows STIG, the OS/390 STIG, and DODI 8500.2.*

- *(TMQ.0011: CAT II) The IAO will ensure that the Application Proxy files are restricted from unauthorized update and access and in accordance with Appendix B.8.*

B.7.2.4 Tivoli Monitoring for Business Integration Tasks

The Tivoli Monitoring for Business Integration Utility Task library contains tasks that are used to perform common functions related to managing MQSeries. The Tivoli Monitoring for Business Integration Utility Tasks library is located in the Manager for MQSeries policy region. Most tasks provided in the Tivoli Monitoring for Business Integration task libraries are available on the pop-up menus on the Manager for MQSeries, the management domain, and the queue manager icons. The following table describes the categories of tasks that exist in the library and a description of use.

Task Category	Description
Configure Event Server	Sets up the event server to process WebSphereMQ events.
Create Inventory Policy Region	Creates the WebSphereMQ Inventory policy region within the Tivoli Monitoring for Business Integration policy region.
Create Management Domain	Creates a WebSphereMQ management domain for managing your resources, such as queue managers, queues, or channels.
Display_Endpoint_Environment	Displays all the environment variables from an endpoint.
Set Queue Manager Icon State	Sets or changes the state of a queue manager icon if the icon does not properly represent the status of the queue manager.
Uninstall_Endpoints_ManagedNodes	Uninstalls the Tivoli Monitoring for Business Integration Version 2.4.0 on managed nodes and endpoints.

Task Category	Description
Operational Tasks	Operational tasks to help manage WebSphereMQ resources. These tasks run on distributed, OS/400, or OS/390 systems, or on all systems. To run the operational tasks, an administrator must have the required MQS_*domain_name*_super or MQS_*domain_name*_admin authorization roles.
Managing the Dead-letter Queue	Used to manage messages on the dead-letter queue (DLQ),
OS/390 Operational Tasks	Provides the command service support to run tasks and an OS/390 system.

Most Tivoli Monitoring for Business Integration tasks can run from one region on task endpoints in an interconnected region. Task processing between two interconnected regions requires that the Tivoli Monitoring for Business Integration be installed on both regions and that the regions are interconnected in a two-way connection and share resources.

- *(TMQ.0012: CAT II) The IAO will ensure that the Tivoli Monitoring for Business Integration tasks are restricted to the TMR administrator and authorized personnel as specified in DODI 8500.2.*

B.7.2.5 Tivoli Monitoring for Business Integration for OS/390

The Tivoli Monitoring for Business Integration for OS/390 allows sending MQSeries requests for information from a managed node to an OS/390 system for processing. It can request processing for a monitoring task on a remote OS/390 system. The Tivoli Monitoring for Business Integration for OS/390 processes requests from a Tivoli Monitoring for Business Integration task or monitor that is sent to the MQSeries queue manager on the OS/390 system.

The Tivoli Business Systems Manager task server provides a channel for forwarding commands to Tivoli Monitoring for Business Integration and to receive responses after command execution. Command requests to the WebSphereMQ queue manager travel from the task server to Tivoli NetView for OS/390 for processing. Tivoli NetView for OS/390 routes the output back to the task server, which forwards the response to the requesting Tivoli Monitoring for Business Integration task or monitor.

Functions that the task server can request include running WebSphereMQ commands, issuing OS/390 commands, and discovering all occurrences of MQSeries queue managers on target OS/390 systems. Running the MQSeries event adapter for OS/390 and the Statistical Event Adapter on an OS/390 system enables events to flow from WebSphereMQ active queue managers on OS/390 systems to the event server.

- *(TMQ.0013: CAT II) The IAO will ensure that the Tivoli Monitoring for Business Integration data sets are restricted using an access control product, which is compliant with DODI 8500.2.*

- *(TMQ.0014: CAT II) The IAO will ensure that the Tivoli Monitoring for Business Integration data sets are restricted using from unauthorized update and access in accordance with the OS/390 STIG and DODI 8500.2.*

- *(TMQ.0015: CAT II) The IAO will ensure that the Tivoli Monitoring for Business Integration commands are restricted in accordance with the OS/390 STIG and DODI 8500.2.*

B.7.2.6 Tivoli Monitoring for Business Integration Files

The following table describes the files that are related to the Tivoli Monitoring for Business Integration:

File	Description
Baroc	Contains the event class definition files specific to the MQSeries events.
Doc	Contains the users guide and release notes in postscript format.
Dsl	Contains the dialog for the various MQSeries tasks.
Icons	Contains the icons used by the TME 10 Module for MQSeries.
Rls	Contains the TEC Adapter rules.
Samp	Contains the MQSeries default MQSC script.
Sh	Contains all the TME Module shell scripts.
Tll	Contains all the TEC MQSeries task shell scripts.
MqsServer	Contains files for defining object classes used by the TME 10 Module for MQSeries.
Bin	Contains platform-specific executables.
Sh	Contains shell scripts for callback methods used for some of the task dialogs.

- *(TMQ.0016: CAT II) The IAO will ensure that the Tivoli Monitoring for Business Integration files are restricted using from unauthorized update and access in accordance with the appropriate platform STIG, this STIG, Appendix B.8, and DODI 8500.2.*

B.7.2.7 Platforms

The Tivoli Monitoring for Business Integration software supports the following platforms:

- UNIX
- Windows
- AS400
- OS/390

B.7.3 General Security Considerations

The unique considerations for Tivoli Monitoring for Business Integration include:

- Access to Tivoli Monitoring for Business Integration functions is controlled through the assignment of Tivoli authorization roles to Tivoli users. Without proper assignment of these roles, the confidentiality and integrity of the WebSphere object data could be compromised.

- Because Tivoli Monitoring for Business Integration requires the use of an RDBMS for data storage, the secure configuration of the client access software and the RDBMS are significant.

- As with most software products, the Tivoli Monitoring for Business Integration is composed of program and data files for which proper access controls are essential. Although most of the Tivoli Monitoring for Business Integration files will reside in directories subject to access controls required for the Framework, there are some Tivoli Monitoring for Business Integration directories and files that require specific access controls.

- If the Tivoli Monitoring for Business Integration remote administration facility is used, administrative or user transactions are processed through Tivoli NetView and the Tivoli Business Systems Manager. The integrity of their product files must be assured to maintain transaction security.

- There are two considerations relative to the security of the Tivoli Manager for MQSeries event server:

  - The host OS must be configured securely.
  - The integrity of the event server software must be maintained.

- There are three considerations relative to the security of the RIM host:

  - The host OS must be configured securely.
  - If required on the platform, the OS user account utilized by Tivoli Monitoring for Business Integration functions must be defined properly.
  - The integrity of the RDBMS client software must be maintained.

- There are four considerations relative to the security of the RDBMS server:

    - The host OS must be configured securely.
    - The RDBMS software must be configured securely.
    - If required on the platform, the OS user account utilized by Tivoli Monitoring for Business Integration functions must be defined properly.
    - The RDBMS user account utilized by the Tivoli Monitoring for Business Integration functions must be defined properly.

- There are two considerations relative to the security of the Gateways:

    - The host OS must be configured securely.
    - The integrity of the Tivoli Monitoring for Business Integration Gateway software must be maintained.

- There are four considerations relative to the security of the Tivoli Monitoring for Business Integration and remote administration interfaces:

    - The integrity of the Tivoli Monitoring for Business Integration remote access facility software must be maintained.
    - The proper assignment of the Tivoli authorization role for the link functions must be managed.
    - The confidentiality of the user authentication data that passes over the network connection between the endpoint or managed node client and the TMR server must be maintained.
    - The integrity of the user machine-based must be maintained.

The host OS security consideration applies generically to the host. It is addressed through the platform security requirements described in the applicable platform STIG. Specifically, the RIM host must be compliant with the STIG that covers the OS used on the RIM host.

Password data is encrypted in accordance with *DODI 8500.2, IA Controls for Individual Identification and Authentication*. Passwords must be encrypted both for storage and for transmission. The actions that are required to address these considerations are addressed in the following section of this document.

B.7.4 Specific Security Considerations

This section describes the specific considerations and required actions to help ensure that the Inventory components are implemented in a secure fashion. As noted in the previous section, compliance with the requirements in the Framework section of this document is assumed as the prerequisite for the information here.

Department of Defense (DOD) Instruction 8500.2, Information Assurance (IA) Implementation, documents IA controls that apply to DOD information systems. The following integrity and confidentiality controls from *DODI 8500.2* are the basis for the requirements in this section: Access for Need-to-Know; Audit Trail, Monitoring, Analysis, and Reporting; Changes to Data; Individual Identification and Authentication; Least Privilege; and Privileged Account Control.

The following subsections describe the considerations and requirements of the Tivoli Monitoring for Business Integration. When reviewing these requirements, the following must be noted:

- The specific directory names in the requirements reflect defaults indicated in vendor documentation. If a site or organization deploying Tivoli chooses other names, the requirements apply to the site-specific names.

- The structure of UNIX permissions can make it difficult to restrict file access to multiple groups of users. The approach taken in this section is to allow UNIX “world” access in cases where non-update (e.g., read or execute) access permission is required for users who may be members of multiple groups and there is no specific need for the files to be inaccessible to other users.

B.7.4.1 Authorization Roles

Authorization roles defined within the Management Framework provide role based access control over functions in the Tivoli products. A Tivoli administrator must be assigned the associated roles to be permitted to perform specific product functions.

Authorization roles such as senior or super enable a broad range of capabilities in multiple products. The following tables describe the authorization roles required for Tivoli Monitoring for Business Integration.

Activity	Context	Required Role
Install Tivoli Monitoring for Business Integration	Tivoli Desktop	super
Create Management Domain	TMR	super
Create Queue Manager	Management Domain	super
Install the Application Proxy	Tivoli Desktop	install_product
Install the Application Proxy patches	Tivoli Desktop	install_product
MQS_Uninstall_Endpoints_ManagedNodes	Managed nodes and endpoints	install_product

NOTE: The Tivoli Management Region administrator or SA who is responsible for the installation will require either UNIX root authority or Windows Administrator authority.

- *(TMQ.0017: CAT II) The IAO will ensure that all authorization roles are assigned on a need-to-know basis and in accordance with DODI 8500.2.*

B.7.4.2 File Permissions

The Tivoli Monitoring for Business Integration software is installed in a directory structure that is subordinate to a logical directory known as the BINDIR directory. The physical implementation of the BINDIR directory is dependent on the platform type (e.g., MS Windows or UNIX) and site installation choices. Some of the Tivoli Monitoring for Business Integration software files are installed in directories that are shared with the Management Framework. These directories are protected through the specific Management Framework requirements elsewhere in this document. Requirements for the Tivoli Monitoring for Business Integration - specific directories are also addressed in *Appendix B.8*. Tivoli Monitoring for Business Integration may be installed using the TMF or by using a script.

Tivoli files for Windows NT are, by default, stored under the *\Tivoli* directory on the root of the selected drive for managed nodes and for endpoints, \program files\Tivoli.

- *(TMQ.0018: CAT II) The IAO will ensure that the Tivoli Monitoring for Business Integration files are secured in accordance with Appendix B.8, the appropriate platform STIG, and DODI 8500.2.*

B.8. Tivoli Component Object Permissions

B.8.1 Introduction

The Tivoli Components in this section are divided by Tivoli product. The sub-section number corresponds to the Appendix B section. For example, the sub-section B.8.2 is broken down as follows:

> **B.8** is the number of the Tivoli Component Object Permission section and the number **2** corresponds to Appendix **B.2**. Each product is further subdivided by UNIX and Windows platform specifics. The use of symbolic variables may be used as part of paths names for flexibility purposes. Finally, in some cases, a Tivoli product may not run on a Windows platform and when that is the case, no table exists.

B.8.2 Tivoli Management Framework

B.8.2.1 UNIX File and Directory Permissions

The following system variables may be established during the installation and may be used as part of the paths in the table below:

- $BINDIR - the high level directory structure that holds the subordinate directories in which Tivoli software is installed

- $INTERP - the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems.

- $DBDIR - the high level directory structure on other managed nodes that holds the subordinate directory in which TMF database files.

- $LIBDIR – the high-level directory structure holds the library files.

- $MANPATH – the high level directory on other managed node holds the subordinate directory in which the manual pages are stored.

This page is intentionally left blank.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Management Framework	TMR Server, Managed Node RIM Host Gateway	**$DBDIR**	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	/usr/lib/X11/app-defaults	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**$BINDIR/**../	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server,	**$BINDIR**/../client_bundle	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**$BINDIR**/../generic	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway, Endpoints	**$BINDIR**/../lcf_bundle	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway, Managed Node Endpoints	**$BINDIR**/../lcf_bundle.40	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**$BINDIR**	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**$BINDIR/**../../include	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**$BINDIR/**../../msg_cat	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**$BINDIR/**../../man	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**$BINDIR/**../../doc	[privileged]	[privileged]	755	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	/etc/Tivoli	[privileged]	[privileged]	755	B.2.2.6

B.8.2.2 Windows File and Directory Permissions

The following system variables may be established during the installation and may be part of the paths in the table below.

- **%BINDIR%** - the high level directory structure that holds the subordinate directories in which Tivoli software is installed
- **%INTERP%** - the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems
- **%DBDIR%** - the high level directory structure on other managed nodes that holds the subordinate directory in which the TMF database software resides.
- **%SystemRoot%** - specifies the system root directory.
- **[Tivoli Users]** - a group containing accounts for non-administrative users of Tivoli
- **[Tivoli Admins]** – Policy RegionAdministrators/individual accounts or a group with responsibility for administration of Tivoli on the platform.

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Tivoli Management Framework	TMR Server, Managed Node RIM Host Gateway	**%DBDIR%**\..\	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**\..\	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**\..\lcf_bundle	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**\..\lcf_bundle.40	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**\..\client_bundle	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**\..\generic	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**\..\generic_unix	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**\..\..\Tivoli\include	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**\..\	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Traverse / Execute	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%LIBDIR%**\..\lib	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.2.2.6

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%BINDIR%**\..\..\msg_cat	TMR Administrators SYSTEM [Tivoli Admins]	Full Control Full Control Full Control	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway, Endpoint	**%SystemRoot%**\system32\drivers\etc\Tivoli	TMR Administrators SYSTEM [Tivoli Admins]	Full Control Full Control Full Control	B.2.2.6
Tivoli Management Framework	TMR Server, Managed Node, Gateway	**%SystemRoot%**\system32\drivers\etc\tll.conf	TMR Administrators SYSTEM [Tivoli Admins]	Full Control Full Control Full Control	B.2.2.6
Tivoli Management Framework	Endpoint	**%SystemRoot%**\system32\	tmersrvd	Read Execute	B.2.2.6
Tivoli Management Framework	Endpoint	**%SystemRoot%**\temp\	tmersrvd	Read Execute	B.2.2.6

B.8.3 Windows Registry Permissions

Tivoli Product	Component	Registry Key	Account Assignment	Permissions	STIG Reference
Tivoli Management Framework	Server	HKEY_LOCAL_MACHINE\SOFTWARE \Tivoli\Platform key	TMR Administrators Policy Region Administrator SSO Platform SA Users	Full Control Execute Full Control Full Control Read	B.2.2.6

B.8.4 Tivoli Enterprise Console

B.8.4.1 Tivoli Enterprise Console Server

B.8.4.1.1 UNIX File Permissions

The following system variables may be established during the installation and may be part of the paths in the table below.

- $BINDIR - the high level directory structure that holds the subordinate directories in which Tivoli software is installed.
- $DBDIR - the high level directory structure on other managed nodes that holds the subordinate directory in which Tivoli Enterprise Console database files.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Enterprise Console	Server	**$DBDIR**/../	[privileged]	[privileged]	755	B.3.1.2.7
Tivoli Enterprise Console	Server	**$BINDIR**/../	[privileged]	[privileged]	755	B.3.1.2.7
Tivoli Enterprise Console	Server	**$BINDIR**/../generic_unix	[privileged]	[privileged]	755	B.3.1.2.7
Tivoli Enterprise Console	Server	**$BINDIR**/../../msg_cat	[privileged]	[privileged]	755	B.3.1.2.7

B.8.4.2 Tivoli Enterprise Console User Interface Server

B.8.4.2.1 UNIX File Permissions

The following system variables may be established during the installation and may be part of the paths in the table below.

- $BINDIR - the high level directory structure that holds the subordinate directories in which Tivoli software is installed.
- $DBDIR - the high level directory structure on other managed nodes that holds the subordinate directory in which Tivoli Enterprise Console database files.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Enterprise Console	Server	**$DBDIR**/../	[privileged]	[privileged]	755	B.3.1.2.7
Tivoli Enterprise Console	Server	**$BINDIR**/../	[privileged]	[privileged]	755	B.3.1.2.7

B.8.4.3 Tivoli Enterprise Console Java Console

B.8.4.3.1 UNIX File Permissions

The following system variables may be established during the installation and may be part of the paths in the table below.

- $BINDIR - the high level directory structure that holds the subordinate directories in which Tivoli Enterprise Console Java software is installed.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Enterprise Console	Desktop	**$BINDIR**/../	[privileged]	[privileged]	755	B.3.1.2.7.1

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Enterprise Console	Desktop	**$BINDIR/../**generic_unix/../	[privileged]	[privileged]	755	B.3.1.2.7.1

B.8.4.4 Tivoli Enterprise Console Sample Event Information

B.8.4.4.1 UNIX File Permissions

The following system variables may be established during the installation and may be part of the paths in the table below.

- $BINDIR - the high level directory structure that holds the subordinate directories in which Tivoli Enterprise Console software is installed.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Enterprise Console	Desktop	**$BINDIR**/../	[privileged]	[privileged]	755	B.3.1.2.7.1
Tivoli Enterprise Console	Desktop	**$BINDIR**/../generic_unix/	[privileged]	[privileged]	755	B.3.1.2.7.1

B.8.4.5 ACF

B.8.4.5.1 UNIX File Permissions (Server)

The following system variables may be established during the installation and may be part of the paths in the table below.

- $BINDIR - the high level directory structure that holds the subordinate directories in which Tivoli Enterprise Console Java software is installed.

- $DBDIR - the high level directory structure on other managed nodes that holds the subordinate directory in which Tivoli ACF database files.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Enterprise Console	Server	**$DBDIR**/../	[privileged]	[privileged]	755	B.3.1.2.7.3
Tivoli Enterprise Console	Server	**$DBDIR**/../	[privileged]	[privileged]	755	B.3.1.2.7.3
Tivoli Enterprise Console	Server	**$BINDIR**/../	[privileged]	[privileged]	755	B.3.1.2.7.3
Tivoli Enterprise Console	Server	**$BINDIR**/../generic_unix/../	[privileged]	[privileged]	755	B.3.1.2.7.3
Tivoli Enterprise Console	Server	**$BINDIR**/../../msg_cat/../	[privileged]	[privileged]	755	B.3.1.2.7.3
Tivoli Enterprise Console	Server	**$LIBDIR**	[privileged]	[privileged]	755	B.3.1.2.7.3

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Enterprise Console	Server	**$BINDIR**/../lcf_bundle/bin/ **$INTERP**	[privileged]	[privileged]	755	B.3.1.2.7.3
Tivoli Enterprise Console	Server	**$BINDIR**/../lcf_bundle/bin/ **$INTERP**/TME	[privileged]	[privileged]	755	B.3.1.2.7.3

B.8.4.5.2 Windows File and Directory Permissions

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Tivoli Enterprise Console	Server	%**BINDIR%**\..\lcf_bundle\bin\**%INTERP%**	TMR Administrators Policy Region Administrator SSO Platform SA Users	Full Control Full Control Full Control Full Control Read & Execute	B.3.1.2.7.3
Tivoli Enterprise Console	Server	**%BINDIR%**\..\lcf_bundle\bin\**%INTERP%** \TME\	TMR Administrators Policy Region Administrator SSO Platform SA Users	Full Control Full Control Full Control Full Control Read & Execute	B.3.1.2.7.3

B.8.4.5.3 UNIX File Permissions (Server)

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Enterprise Console	Server	**$ORACLE_HOME/**	[privileged]	[privileged]	755	B.3.1.2.7.3
Tivoli Enterprise Console	Server	**$ORACLE_HOME**/bin	[privileged]	[privileged]	755	B.3.1.2.7.3

B.8.4.5.4 Windows File and Directory Permissions

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Tivoli Enterprise Console	Server	**%ORACLE_HOME%**\	TMR Administrators Policy Region Administrator SSO Platform SA Users	Full Control Full Control Full Control Full Control Read & Execute	B.3.1.2.7.3
Tivoli Enterprise Console	Server	**%ORACLE_HOME%**\bin	TMR Administrators Policy Region Administrator SSO Platform SA Users	Full Control Full Control Full Control Full Control Read & Execute	B.3.1.2.7.3

B.8.5 IBM Tivoli Monitoring

B.8.5.1 UNIX File and Directory Permissions

The following system variables may be established during the installation and may be part of the paths in the table below.

- **$LCF_DATDIR -** the high level directory structure on endpoints that holds the subordinate directory in which the config file is stored.
- **$LCF_BINDIR -** the high level directory structure on endpoints that holds the subordinate directory in which software is installed.
- **$LCF_CATDIR -** the high level directory structure on endpoints that holds the subordinate directory in which message files are located.
- **$LCFROOT -** the high level root directory.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
ITM for Business Integration	TMR Server Gateways	**$LIBDIR**/../	[privileged]	[privileged]	755	B.4.2.2
ITM for Business Integration	Endpoints	**$LCF_DATDIR**/../../	[privileged]	[privileged]	755	B.4.2.2
ITM for Business Integration	Endpoints	**$LCF_BINDIR**/../	[privileged]	[privileged]	755	B.4.2.2
ITM for Business Integration	Endpoints	**$LCF_LIBDIR**/../	[privileged]	[privileged]	755	B.4.2.2
ITM for Business Integration	TMR Server Gateway Endpoints	**$LCF_CATDIR**	[privileged]	[privileged]	755	B.4.2.2

B.8.5.2 Windows File and Directory Permissions

The following system variables may be established during the installation and may be part of the paths in the table below.

- **%LCF_DATDIR% -** the high level directory structure on endpoints that holds the subordinate directory in which the config file is stored.
- **%LCF_BINDIR% -** the high level directory structure on endpoints that holds the subordinate directory in which software is installed.
- **%LCF_CATDIR% -** the high level directory structure on endpoints that holds the subordinate directory in which message files are located.
- **%LCFROOT% -** the high level root directory.

- **[Tivoli Users]** - a group containing accounts for non-administrative users of Tivoli
- **[Tivoli Admins] –** Policy RegionAdministrators/individual accounts or a group with responsibility for administration of Tivoli on the platform.

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
ITM for Business Integration	TMR Server Gateways Endpoints	**%LCF_DATDIR%**\..\..\	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.4.2.2
ITM for Business Integration	TMR Server Gateway Endpoints	**%LCF_BINDIR%\..**	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.4.2.2
ITM for Business Integration	TMR Server Gateway Endpoints	**%LCF_LIBDIR%\..**	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.4.2.2
ITM for Business Integration	TMR Server Gateway Endpoints	**%LCF_CATDIR%**	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.4.2.2

B.8.6 Tivoli Inventory

B.8.6.1 UNIX File and Directory Permissions

The following notation is used in this section:

- $ORACLE_HOME – the high level directory structure that holds the subordinate directories in which the Oracle RDBMS client software resides
- $BINDIR – the high level directory structure that holds the subordinate directories in which Tivoli software resides
- $INTERP – the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems
- $DBDIR – the high level directory structure on other managed nodes that holds the subordinate directories in which Tivoli scanner software and output resides
- $OID – the Tivoli object ID of the node
- $LCF_BASE_DIR – the high level directory structure on endpoints that holds the subordinate directories in which Tivoli scanner software and output resides.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Inventory	RIM Host	**$ORACLE_HOME**/	[privileged]	[privileged]	755	B.5.3.2.1
Tivoli Inventory	RIM Host	**$ORACLE_HOME**/bin	[privileged]	[privileged]	751	B.5.3.2.1
Tivoli Inventory	Server	**$BINDIR**/TME/INVENTORY	[privileged]	[privileged]	755	B.5.3.4
Tivoli Inventory	Server	**$BINDIR**/../generic/TME/INVENTORY	[privileged]	[privileged]	755	B.5.3.4
Tivoli Inventory	Server	**$BINDIR**/../generic/HTTPd/Inventory	[privileged]	[privileged]	755	B.5.3.4
Tivoli Inventory	Server	**$BINDIR**/../generic/HTTPd/Inv	[privileged]	[privileged]	755	B.5.3.4
Tivoli Inventory	Server	**$BINDIR**/../generic/HTTPd/UserLink	[privileged]	[privileged]	755	B.5.3.4
Tivoli Inventory	Server	**$BINDIR**/TAS/HTTPd/cgi-bin/Inventory	[privileged]	[privileged]	751	B.5.3.4
Tivoli Inventory	Server	**$BINDIR**/TAS/HTTPd/cgi-bin/Inv	[privileged]	[privileged]	751	B.5.3.4
Tivoli Inventory	Server	**$BINDIR**/TAS/HTTPd/cgi-bin/UserLink	[privileged]	[privileged]	751	B.5.3.4
Tivoli Inventory	Gateway	**$BINDIR**/../lcf_bundle/bin/**$INTERP** /inv	[privileged]	[privileged]	755	B.5.3.5
Tivoli Inventory	Gateway	**$BINDIR**/../lcf_bundle/bin/**$INTERP** /TME/INVENTORY	[privileged]	[privileged]	755	B.5.3.5

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Tivoli Inventory	Scanner	**$DBDIR**/inventory	[privileged]	[privileged]	755	B.5.3.6
Tivoli Inventory	Scanner	**$DBDIR**/inventory/**$OID**	[privileged]	[privileged]	755	B.5.3.6
Tivoli Inventory	Scanner	**$LCF_BASE_DIR**/inv	[privileged]	[privileged]	755	B.5.3.6
Tivoli Inventory	Scanner	**$LCF_BASE_DIR**/inv/SCANNER	[privileged]	[privileged]	751	B.5.3.6
Tivoli Inventory	UserLink	…/UserLink.htm | …/userlink.htm	[privileged]	[privileged]	755	B.5.3.7

B.8.6.2 Windows File and Directory Permissions

The following notation is used in this section:

- %ORACLE_HOME% - the high level directory structure that holds the subordinate directories in which the Oracle RDBMS client software resides
- %BINDIR% - the high level directory structure that holds the subordinate directories in which Tivoli software is installed
- %INTERP% - the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems
- %SCANDIR% - the high level directory structure on PC managed nodes that holds the subordinate directories in which Tivoli scanner software and output resides
- %DBDIR% - the high level directory structure on other managed nodes that holds the subordinate directories in which Tivoli scanner software and output resides
- %OID% - the Tivoli object ID of the node
- %LCF_BASE_DIR% - the high level directory structure on endpoints that holds the subordinate directories in which Tivoli scanner software and output resides
- [Tivoli DB account] – account used by the RIM Host to access the Inventory Oracle database
- [Tivoli Users] – a group containing accounts for non-administrative users of Tivoli
- [Tivoli Admins] – individual accounts or a group with responsibility for administration of Tivoli on the platform.

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Tivoli Inventory	RIM Host	**%ORACLE_HOME%**\	Administrators SYSTEM [Tivoli Admins] [Tivoli DB account]	Full Control Full Control Full Control Read & Execute	B.5.3.2.1
Tivoli Inventory	RIM Host	**%ORACLE_HOME%**\bin	Administrators SYSTEM [Tivoli Admins] [Tivoli DB account]	Full Control Full Control Full Control Traverse / Execute	B.5.3.2.1
Tivoli Inventory	Server	%**BINDIR%**\TME\INVENTORY	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.4
Tivoli Inventory	Server	%**BINDIR%**\..\generic\TME\INVENTORY	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.4
Tivoli Inventory	Server	%**BINDIR%**\..\generic\HTTPd\Inventory	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.4
Tivoli Inventory	Server	%**BINDIR%**\..\generic\HTTPd\Inv	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.4
Tivoli Inventory	Server	%**BINDIR%**\..\generic\HTTPd\UserLink	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.4

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Tivoli Inventory	Server	%**BINDIR%**\TAS\HTTPd\cgi-bin\Inventory	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Traverse / Execute	B.5.3.4
Tivoli Inventory	Server	%**BINDIR%**\TAS\HTTPd\cgi-bin\Inv	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Traverse / Execute	B.5.3.4
Tivoli Inventory	Server	%**BINDIR%**\TAS\HTTPd\cgi-bin\UserLink	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Traverse / Execute	B.5.3.4
Tivoli Inventory	Gateway	%**BINDIR%**\..\lcf_bundle\bin\**%INTERP%**\inv	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.5
Tivoli Inventory	Gateway	**%BINDIR%**\..\lcf_bundle\bin\**%INTERP%**\TME\INVENTORY	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.5
Tivoli Inventory	Scanner	**%SCANDIR%**	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.6
Tivoli Inventory	Scanner	**%SCANDIR%**\OUTPUT	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Modify	B.5.3.6

Tivoli Product	**Component**	**Object**	**Account Assignment**	**Permissions**	**STIG Reference**
Tivoli Inventory	Scanner	**%DBDIR%**\inventory	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.6
Tivoli Inventory	Scanner	**%DBDIR%**\inventory\**%OID%**	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Modify	B.5.3.6
Tivoli Inventory	Scanner	**%LCF_BASE_DIR%**\inv	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.5.3.6
Tivoli Inventory	Scanner	**%LCF_BASE_DIR%**\inv\SCANNER	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Modify	B.5.3.6
Tivoli Inventory	UserLink	…\UserLink.htm | …\userlink.htm	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read	B.5.3.7

B.8.6.3 Windows Registry Permissions

The following notation is used in this section:

- [Tivoli DB account] – account used by the RIM Host to access the Inventory Oracle database
- [Tivoli Admins] – individual accounts or a group with responsibility for administration of Tivoli on the platform.

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Tivoli Inventory	RIM Host	HKLM\SOFTWARE\ORACLE (include all subkeys)	Administrators SYSTEM [Tivoli Admins] [Tivoli DB account]	Full Control Full Control Full Control Read	B.5.3.2.1

B.8.7 Tivoli Software Distribution

B.8.7.1 UNIX File and Directory Permissions

The following notation is used in this section:

- $BINDIR – the high level directory structure that holds the subordinate directories in which Tivoli software is installed.
- $INTERP – the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Software Distribution	Server	**$BINDIR**/TME/COURIER	[privileged]	[privileged]	755	B.6.3.2.1
Software Distribution	Server	**$BINDIR**/../generic/TME/COURIER	[privileged]	[privileged]	755	B.6.3.2.1
Software Distribution	Server	**$BINDIR**/TAS/HTTPd/cgi-bin/Courier	[privileged]	[privileged]	751	B.6.3.2.1
Software Distribution	Gateway	**$BINDIR**/../lcf_bundle/bin/**$INTERP** /TME/COURIER	[privileged]	[privileged]	755	B.6.3.4

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
Software Distribution	TEC Integration	…/tecad_sd.conf	[privileged]	[privileged]	750	B.6.3.5
Software Distribution	TEC Integration	…/tecad_sd.baroc	[privileged]	[privileged]	750	B.6.3.5
Software Distribution	Extension API	[Software Distribution Extension API directory]	[privileged]	[privileged]	750	B.6.3.6
Software Distribution	Historical DB	**$BINDIR**//TME/COURIER/SCRIPTS	[privileged]	[privileged]	750	B.6.3.7
Software Distribution	UserLink	…/UserLink.htm | …/userlink.htm	[privileged]	[privileged]	755	B.6.3.10

B.8.7.2 Windows File and Directory Permissions

The following notation is used in this section:

- %BINDIR% - the high level directory structure that holds the subordinate directories in which Tivoli software is installed
- %INTERP% - the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems
- [Tivoli Users] – a group containing accounts for non-administrative users of Tivoli
- [Tivoli Admins] – individual accounts or a group with responsibility for administration of Tivoli on the platform.

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Software Distribution	Server	**%BINDIR%**\TME\COURIER	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.6.3.2.1
Software Distribution	Server	**%BINDIR%**\..\generic\TME\COURIER	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.6.3.2.1

Tivoli Product	Component	Object	Account Assignment	Permissions	STIG Reference
Software Distribution	Server	**%BINDIR%**\TAS\HTTPd\cgi-bin\Courier	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Traverse / Execute	B.6.3.2.1
Software Distribution	Gateway	**%BINDIR%**\..\lcf_bundle\bin\ **%INTERP%** \TME\COURIER	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.6.3.4
Software Distribution	TEC Integration	…/tecad_sd.conf	Administrators SYSTEM [Tivoli Admins]	Full Control Full Control Full Control	B.6.3.5
Software Distribution	TEC Integration	…/tecad_sd.baroc	Administrators SYSTEM [Tivoli Admins]	Full Control Full Control Full Control	B.6.3.5
Software Distribution	Extension API	[Software Distribution Extension API directory]	Administrators SYSTEM [Tivoli Admins]	Full Control Full Control Full Control	B.6.3.6
Software Distribution	Historical DB	**%BINDIR%**\TME\COURIER\SCRIPTS	Administrators SYSTEM [Tivoli Admins]	Full Control Full Control Full Control	B.6.3.7
Software Distribution	AutoPack Agent	…/**wsyschg.exe**	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.6.3.9
Software Distribution	UserLink	…/UserLink.htm | …/userlink.htm	Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read	B.6.3.10

B.8.8 IBM Tivoli Monitoring for Business Integration

B.8.8.1 UNIX File and Directory Permissions

The following system variables may be established during the installation and may be used as part of the paths in the table below.

- **$BINDIR** – the high level directory structure that holds the subordinate directories in which Tivoli software is installed
- **$INTERP** – the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems.
- **$DBDIR** – the high level directory structure on other managed nodes that holds the subordinate directory in which ITM database files are located.
- **$LIBDIR** – the high level directory structure holds directory of the interpreter type.
- **$LCF_DATDIR –** the high level directory structure on endpoints that holds the subordinate directory in which config and log files are stored.
- **$LCF_BINDIR –** the high level directory structure on endpoints that holds the subordinate directory in which software is installed.
- **$LCF_CATDIR –** the high level directory structure on endpoints that holds the subordinate directory in which message files are located.
- **$LCFROOT –** the high level root directory.

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
ITM for Business Integration	TMR Server	**$DBDIR**/../	[privileged]	[privileged]	755	B.7.4.2
ITM for Business Integration	TMR Server Gateways	**$BINDIR**/../	[privileged]	[privileged]	755	B.7.4.2
ITM for Business Integration	TMR Server Gateways	**$LIBDIR**/../	[privileged]	[privileged]	755	B.7.4.2

Tivoli Product	Component	Object	Owner	Group	Permissions	STIG Reference
ITM for Business Integration	Endpoints	**$LCF_DATDIR**/../	[privileged]	[privileged]	755	B.7.4.2
ITM for Business Integration	Endpoints	**$LCF_BINDIR**/../	[privileged]	[privileged]	755	B.7.4.2
ITM for Business Integration	Endpoints	**$LCF_LIBDIR**/../	[privileged]	[privileged]	755	B.7.4.2
ITM for Business Integration	TMR Server Gateway Endpoints	**$LCF_CATDIR**	[privileged]	[privileged]	755	B.7.4.2

B.8.8.2 Windows File and Directory Permissions

The following system variables may be established during the installation and may be part of the paths in the table below.

- **%BINDIR%** - the high level directory structure that holds the subordinate directories in which Tivoli software is installed
- **%INTERP%** - the interpreter type, an OS-specific identifier that is used to separate files required to support different operating systems.
- **%DBDIR%** - the high level directory structure on other managed nodes that holds the subordinate directory in which the TMF database files resides.
- **%LIBDIR% –** the high level directory structure holds directory of the interpreter type.
- **%LCF_DATDIR% -** the high level directory structure on endpoints that holds the subordinate directory in which ____________ are stored.
- **%LCF_BINDIR% -** the high level directory structure on endpoints that holds the subordinate directory in which software is installed.
- **%LCF_CATDIR% -** the high level directory structure on endpoints that holds the subordinate directory in which message files are located.
- **%LCFROOT% -** the high level root directory.
- **[Tivoli Users] –** a group containing accounts for non-administrative users of Tivoli.
- **[Tivoli Admins] –** Policy RegionAdministrators/individual accounts or a group with responsibility for administration of Tivoli on the platform.

Tivoli Product	**Component**	**Object**	**Account Assignment**	**Permissions**	**STIG Reference**
ITM for Business Integration	TMR Server	**%DBDIR%**\..\	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.7.4.2
Tivoli Management Framework	TMR Server, Managed Node Gateway	**%BINDIR%\..**	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.7.4.2
ITM for Business Integration	TMR Server Managed Node Gateway	**%DBDIR%\..**	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.7.4.2
ITM for Business Integration	Endpoints	**%LCF_DATDIR%**\..\	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.7.4.2
ITM for Business Integration	Endpoints	**%LCF_BINDIR%\..**	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.7.4.2
ITM for Business Integration	Endpoints	**%LCF_LIBDIR/%\..**	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.7.4.2
ITM for Business Integration	Endpoints	**%LCF_CATDIR%**	TMR Administrators SYSTEM [Tivoli Admins] [Tivoli Users]	Full Control Full Control Full Control Read & Execute	B.7.4.2

APPENDIX C. MICROSOFT SYSTEMS MANAGEMENT SERVER (SMS)

C.1 Microsoft Systems Management Server (SMS)

C.1.1 Systems Management Server Overview

Microsoft's Systems Management Server (SMS) is software that addresses the configuration management functions of Enterprise System Management for Windows-based environments. Limited support for some network infrastructure such as switches and routers is included and third-party support for non-Windows hosts is available. SMS is part of Microsoft's infrastructure management product group that includes the Microsoft Operations Manager (MOM) product and the Application Center product.

The following information provides an overview of SMS functions, its architecture, ESM elements it implements, and the industry standards used in the product. A brief summary of similar Microsoft products and the relevant product release are noted to put this discussion in proper context.

Microsoft describes the functions that SMS performs in terms of four areas:

- Inventory includes the collection of hardware and software data for managed systems.
- Provisioning includes capabilities to deploy and update software as well as perform configuration changes.
- Troubleshooting includes remote viewing and manipulation of client systems.
- Reporting includes the capability to generate and view detailed reports based on vendor-supplied and locally developed report specifications.

SMS is based on a flexible architecture that supports distributed server functions, with a logical site as the organizing unit. Complex hierarchies that include primary and secondary sites and parent and child sites can be designed to reflect the operating environment. However, as the complexity of the server distribution and hierarchy implementation increases, the need for careful security configuration and procedures increases as well.

The SMS implementation maps well to the elements discussed in *Section 2.1.2, ESM Implementation Elements*. The correspondence can be described as:

- Manager – ESM manager functions are primarily implemented in SMS through applications that run as Windows Services on SMS server machines. The SMS Executive service is the primary element that fulfills this role.

- Agent – ESM agent functions are implemented in SMS through applications that run as Windows Services on SMS client machines. The SMS Advanced Client implements this role through the SMS Agent Host service.

- Console – The ESM console function is implemented in SMS through the SMS Administrator Console; it is built on the Microsoft Management Console (MMC) framework. The SMS Administrator Console is always installed on the SMS Site Server and can be installed on other server or client machines.

- Management Data Repository – The ESM management data repository is implemented in SMS through the SMS Site Database. This database service is provided by the Microsoft SQL Server product and typically runs on the server machine identified as the SMS Site Server.

SMS employs technologies based on common industry standards:

- SMS inventory functions can optionally use the Simple Network Management Protocol (SNMP) to gather information from Windows clients that are running an SNMP agent.

- SMS relies heavily on the use of Windows Management Instrumentation (WMI) that is implemented in current versions of Windows operating systems. WMI is Microsoft's implementation of the Web-Based Enterprise Management (WBEM) standards set. SMS uses WMI capabilities to communicate with and manage clients, as well as a way to provide a standard interface to its own methods and data.

Microsoft offers other product components and technologies that provide a subset of the configuration management functions found in SMS. Some brief notes on these components helps to put SMS into perspective.

Windows Update is a public, Internet-based web service hosted by Microsoft. It is intended to allow individual clients to download patches, updates, and service packs for certain Windows operating system versions from a public Microsoft web site. Windows Update does not provide features such as selective targeting, network use optimization, distribution control, reporting, or deployment planning.

Software Update Services (SUS) and its announced successor, Windows Update Services (WUS), are optional components for current Windows server operating systems. These components allow administrators to enable deployment of updates from their own servers to clients within their infrastructure. SUS provides the ability to deploy critical updates, security patches, and service packs for current Windows operating systems. WUS will expand on this by offering the capability to update other Microsoft products such as Office, SQL Server, and Exchange. SUS and (to a greater extent) WUS offer limited capabilities for some additional features such as network use optimization, distribution control, and reporting.

In contrast to Windows Update and SUS\WUS, the SMS product is intended as an enterprise-class solution to system management issues including and beyond deploying updates. SMS provides many additional features and an extensible architecture that allows administrators to use the tools to perform functions not supported by the other solutions.

Please note that this document is based on SMS 2003. While the information may be applicable to earlier releases of SMS, it has not been verified for those releases. The primary source for this information is the *Microsoft Systems Management Server 2003 Concepts, Planning, and Deployment Guide*, the *Microsoft Systems Management Server 2003 Operations Guide*, and the *Scenarios and Procedures for Microsoft Systems Management Server 2003: Security* document. Refer to *Appendix A, Related Publications*, for additional documentation and the addresses of web sites with more information.

C.1.1.1 SMS Component Overview

SMS is a complex software product with a number of components and an architecture that allows it to be implemented in a variety of configurations. This section will describe some of the basic parts and concepts that should be understood in order to configure SMS and to understand the security issues.

A *site* can be thought of as the basic logical unit in the hierarchy of an SMS implementation. A site is related to the span of control in SMS. Different types of sites can be created:

Primary site – Every SMS implementation must have at least one primary site. A primary site includes a site database that holds information for the site and the child sites that report to it. A primary site can report to another primary site.

- Secondary site – An SMS implementation may have one or more secondary sites. A secondary site does not have its own database; the site forwards its information to, and is administered from, the primary site to which it reports.

- Central site – A central site is the primary site at the top of the SMS hierarchy. All other sites report directly or indirectly to it.

- Parent site – A parent site is a site that has one or more child sites that report to it. A parent site must be a primary site.

- Child site – A child site is a site that reports to another site. A child site may be a primary or a secondary site.

At the time an SMS site is created, it must be assigned a three-character site code. This code identifies the site and is used in the names of Windows accounts that SMS automatically creates. The SMS site code is also used as a way of indicating the site to which a specific client is assigned.

Each site must be configured with boundaries that specify the span of control. Site boundaries are specified in terms of IP subnets and / or Active Directory sites. A site manages the clients within its site boundaries. When Advanced Clients are used, roaming boundaries are specified to control the Distribution Point servers to which a roaming client is allowed to connect.

Each SMS site is created by the installation of SMS software on one or more Windows server machines. Specific terminology is used to describe the SMS servers and functions:

- Site server – A site server is the principal server machine and could be the only server within the site on which the SMS server software is installed. Some or all of the SMS server functions or roles are performed on the site server.

- Site system – A site system is a server machine within the site on which some of the SMS server functions or roles are performed.

- Site system role – A site system role is a specific function that a site system performs for that site. The roles defined in SMS include: site server, site database (or SQL) server, SMS Provider, Client Access Point (CAP), Distribution Point, Management Point, Server Locator Point, and Reporting Point. Site system roles can be distributed across multiple server machines.

Some important points to note about SMS servers include the following:

- The site server and site database server roles are commonly implemented on the same physical machine. There are performance and security advantages to this configuration.

- The Management Point, Server Locator Point, and Reporting Point roles require Internet Information Services (IIS) to be installed and operational on the server. Separating these roles to run on a different machine than the site server is recommended for this reason.

The goal of implementing an SMS site is to establish the capability to manage clients. SMS 2003 supports the definition of two types of clients: Legacy Clients and Advanced Clients.

Legacy Clients have the following characteristics:

- Legacy Clients are carried over from earlier releases of SMS and support older Windows operating systems including Windows 9x and Windows NT.

- Legacy Clients require more and more highly privileged Windows application accounts to be defined.

- Microsoft has announced their intent to drop support for Legacy Clients in a future release of SMS.

Advanced Clients have the following characteristics:

- Advanced Clients run on Windows 2000 and later operating systems.

- Advanced Clients require fewer Windows application accounts.

- Advanced Clients support transmission integrity through digital signature checks on SMS policy and content.

- Advanced Clients support the concept of roaming, which is the ability to move a client machine from one IP subnet or Active Directory site to another. This can decrease the complexity of managing mobile machines.

- Advanced Clients support the use of the Background Intelligent Transfer Service (BITS). This simplifies firewall port configuration and provides greater efficiency in receiving file distributions.

Figure C-1 depicts how a simple SMS site using Advanced Clients could be configured.

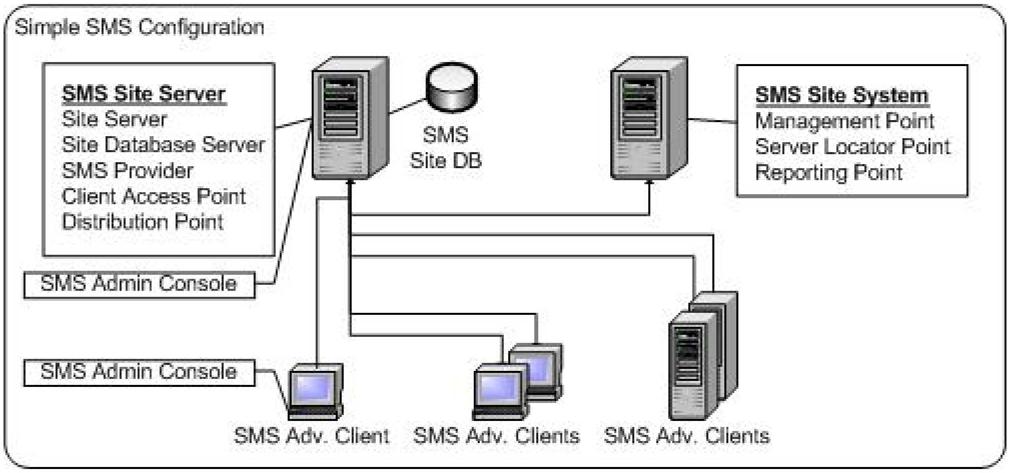

Figure C-3. Simple SMS Site

SMS capabilities are implemented through functions that interact with the server and client components. A basic understanding of the following functions helps to understand SMS security issues: resource discovery, the SMS status system, logging, maintenance tasks, and the use of WMI.

Resource discovery consists of methods that SMS uses to capture and maintain basic data about objects within a site’s boundaries. The types of SMS Discovery fit roughly into three categories:

- Discovery can gather information from Windows domain controller repositories such as Active Directory containers.

- Discovery can query individual machines using Windows function calls relating to file sharing capabilities.

- Discovery can gather information through standard TCP/IP network facilities such as the Internet Control Message Protocol (ICMP), Dynamic Host Configuration Protocol (DHCP), or SNMP.

The SMS status system provides the primary tool for determining the operational condition of SMS server components. Status messages are generated by individual SMS components and stored in the site database. Status filter rules control retention, reporting, replication, and external program actions. Status summarizer components use status messages and other site database information to create summaries of the operational state of the SMS hierarchy. Status messages are identified as one of three types: milestone, detail, or audit.

Many SMS components include a logging capability that collects detailed data about SMS activities. The status, location, and size of a component's log are configurable. While most of the component logging might be enabled only for detail-level troubleshooting, logging for the SMS Site Backup would be considered routine.

SMS maintenance tasks provide a method to automate a schedule of routine actions needed to keep SMS operating normally. The product comes with several pre-defined tasks. The most notable of these is the Backup SMS Site Server task.

The use of Windows Management Instrumentation (WMI) by SMS deserves note because WMI is supplied as a component of current Windows operating systems, but SMS depends heavily on it to gather and manage information. As noted earlier, WMI is an implementation of the WBEM standards set. As such WMI incorporates methods and data structures related to system management. Therefore access to WMI methods and data is a security concern for SMS.

WMI data is stored in logical structures known as namespaces. Namespaces are organized hierarchically, expressed as parent and child namespaces. Certain namespaces are defined and populated by the Windows OS. When an SMS server or client component is installed on a machine, additional namespaces are created. Access is controlled at the namespace level, with support for the concept of inheriting namespace access permissions from parent to child. The WMI Control, a plug-in to the Microsoft Management Console, provides access to WMI namespace access settings.

A choice of security modes in which to operate was added in SMS 2003. Standard security mode is essentially the same as the only security option in SMS 2.0. Advanced security mode was added to address the complexity and administrative effort associated with Windows application account definitions in the older SMS security mode. The subject of security modes is addressed in the next section, but some important issues should be noted:

- Advanced security mode requires the implementation of Windows Active Directory for all SMS servers within a site and at all network locations.

- While it is possible to operate a hierarchy of SMS sites with a mixture of security modes, certain sites in the hierarchy (parents of other Advanced security mode sites) would be required to operate in Advanced security mode.

Although it is not a component that currently comes with the basic SMS installation materials, the SMS Toolkit should be considered required for SMS at all installations. The toolkit can be downloaded from Microsoft at the web site noted in *Appendix A, Related Publications*. Three elements in the Toolkit are necessities:

- The SMS Trace tool is a viewer for the component log files. While the files are text files that can be viewed using other applications, the SMS Trace tool provides formatting, filtering, and highlighting that are essential to efficiently finding information in the logs.

- The IIS Lockdown Template is used to configure the Windows IIS component more securely. The SMS Toolkit version is specifically tailored for SMS environments.

- The URLScan Template is also used to increase IIS security in terms of file filtering. This Toolkit version is also specifically tailored for SMS servers.

C.1.1.2 General Security Considerations

Enhancing the security of SMS configurations requires the application of a number of known best practices for information systems. This task is made somewhat simpler because the components run on a homogenous platform base, Microsoft Windows operating systems. At the same time, the distributed nature of SMS, the potential for different hierarchical deployments, and various product options complicate an attempt to provide straightforward security guidance.

As a preface to the next section in this document, the following paragraphs discuss these general security considerations for SMS:

- SMS server roles and server partitioning
- SMS configuration options: security mode and client type
- Database and web server software
- OS accounts required by SMS
- SMS program and data files
- SMS object access control
- WMI namespace access control
- SMS Feature components
- Status and logging mechanisms
- Network issues
- Installation issues
- Backup and recovery issues

It was noted in *Section C.1.1.1, SMS Component Overview*, that SMS server functions are defined in terms of site system roles. This is relevant to security because some of the roles have specific software and OS account requirements that, if combined in certain machine configurations, could degrade overall security. As a result certain server partitioning requirements and recommendations apply.

SMS 2003 offers two security modes: Standard and Advanced. The choice of mode directly impacts security because it results in significant differences in the types of OS accounts required for SMS to operate. Standard security employs typical administrator-defined user accounts; Advanced security uses Windows LocalSystem and computer accounts.

SMS 2003 offers two types of clients: Legacy and Advanced. There are a number of security considerations for this choice and Microsoft documentation explicitly states that the Legacy Client "…is not considered a secure environment."

SMS requires the use of the Microsoft SQL Server database product and the Windows IIS web server component. These software packages must be configured to comply with other applicable security implementation guides.

SMS permits flexibility in the OS accounts that it uses for access between servers and clients. Exposure can be reduced by following requirements for more accounts, but with the least privilege principle applied to each.

As with the implementation of all software packages, there are program files and data files for which access protection must be controlled in order to maintain the desired confidentiality, integrity, and availability characteristics of SMS.

Use of SMS administrative privileges is controlled through SMS object access definitions. In anything but the simplest administrative environments, an access control strategy must be employed to ensure that separation of duties is maintained.

The use of WMI namespaces by SMS was briefly described earlier. Limiting access to those namespaces is needed to maintain SMS availability and integrity.

Microsoft identifies some of the SMS product components as SMS Features. This includes Reporting, Remote Tools, Software Distribution, and Inventory Collection. The current coverage in this document of security considerations for the SMS Features is limited. However, due to identified security vulnerabilities in the Remote Tools feature, a specific requirement covering that feature is necessary.

SMS offers status and logging capabilities that are directly related to maintaining SMS availability. In addition, the use of certain status features is needed to maintain an audit trail of activity.

The distributed architecture of SMS and its management mission cannot be implemented without the use of networks for communication. The TCP/IP ports used by SMS are subject to other security guidance that restricts their use outside of enclave boundaries. The data being transported is also subject to confidentiality and integrity requirements.

During the installation phases of SMS, issues can arise from the setup and configuration options that are selected. Although the results of undesirable choices can be corrected later, the better choices can eliminate windows of vulnerability during installation:

- The SMS setup process defines two types: express and custom. Microsoft characterizes express setup as "appropriate for setting up evaluation sites on an isolated network" because it enables SMS configuration options that could cause serious negative impact to a production environment. Specifically, the SMS Discovery methods, client installation types, or Remote Tools feature that are enabled by express setup could negatively impact the integrity and availability of the environment.

- When SMS setup is executed, access to some administrative privileges is automatically granted to the Windows account used to perform the installation. An administrative account strategy can help to ensure that this access is appropriately granted.

- Early implementation steps for SMS can include the use of Network Discovery to detect assets. Without adequate planning and coordination, the generated network activity might be interpreted incorrectly as a form of network attack.

An instance of reduced availability of the SMS system could lead to further, more serious security problems. For example, the inability to deploy a critical security patch could allow many unprotected client systems to be compromised. This is one of the reasons why backup and recovery must be considered even more important for SMS server systems than for some other categories of servers.

C.1.2 Systems Management Server Specific Configuration

The following subsections describe considerations and requirements for securing SMS configurations. The information is organized according to the subject areas defined in *Department of Defense (DOD) Instruction 8500.2, Information Assurance (IA) Implementation,* and corresponds to the organization of *Section 3, Enterprise System Management Security,* in this document.

When reviewing the requirements, the following must be noted:

- The specific file and directory names referenced in the requirements and specified in the appendix reflect defaults indicated in vendor documentation. If a site or organization deploying SMS chooses other names, the requirements apply to the site-specific names.

- The term Site Server refers to machines on which the SMS site server role executes. The term Site System refers to machines running Client Access Point (CAP), Distribution Point, Management Point, Server Locator Point, and Reporting Point roles.

C.1.2.1 Security Design and Configuration

This section describes the specific considerations and requirements for SMS that are related to the IA controls in the Security Design and Configuration subject area. The following information is addressed:

- Cryptographic algorithms
- Protection of SMS software and limits on privileged program use
- Software release maintenance
- Application partitioning
- Network port usage

SMS can use encryption, hashing, and signing algorithms for some data transfer operations. However, there are no individual SMS controls that allow the selection of specific FIPS 140-compliant algorithms. This issue is addressed by following the requirements in the *Windows NT\2000\XP Addendum* and the associated NSA guides that address OS security configuration.

SMS software resides in directories and files that are installed on server and client systems. The assignment of directory and file access permissions for the SMS software is restricted in accordance with the *DODI 8500.2, IA System Library Management Controls*. Access permissions must be assigned only to process and user accounts that require the associated access to perform designated functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as the accounts used by the SMS clients, are allowed execute access to program files and read access to other objects.

- *(EMS.0010: CAT II) The SA will ensure that access to SMS server software libraries (including executable and configuration files) is restricted in accordance with the permissions in Appendix C.1.3.*

- *(EMS.0020: CAT III) The SA will ensure that access to SMS client software libraries (including executable and configuration files) is restricted in accordance with the permissions in Appendix C.1.3.*

The SMS server software libraries include tools that are used for special maintenance tasks. These tools have the capability to significantly impact the operation of an SMS site and clients. Included in these tools are the following:

- The ACL Reset tool (**ACLreset.exe**) is primarily used during recovery operations to synchronize directory permissions with Windows application accounts that are being redefined.

- The Hierarchy Maintenance tool (**PreInst.exe**) is a multi-purpose utility used for diagnosis, repair, recovery, and key management tasks. Its use may be required in the event disaster recovery must be performed.

- The Remote Tools command line program (**Remote.exe**) enables Remote Tools to be run outside of the Administrator console.

Access to the ACL Reset and Hierarchy Maintenance programs is appropriate only for administrators with responsibility for SMS site maintenance because of the powerful functions they perform. Access to the Remote Tools program must be closely restricted because of the security issues associated with the Remote Tools feature. Additional information on Remote Tools is found in *Section C.1.2.3.2, Account Management*.

- *(EMS.0030: CAT II) The SA will ensure that access to the SMS ACL Reset, Hierarchy Maintenance, and Remote Tools programs is restricted in accordance with the permissions in Appendix C.1.3.*

As Microsoft has continued development of the SMS product, new releases have been made available. In conjunction with the Microsoft Support Lifecycle policy, there are limits to the length of time a product release remains supported. At the end of this period, security updates are no longer provided. Release 1 of SMS has already reached this point.

To prevent situations in which an organization becomes vulnerable to a security problem that the vendor will not repair or confirm, requirements for the use of supported products and for migration planning are necessary.

- *(EGA.0110: CAT I) The IAO will ensure that SMS software is removed or upgraded prior to the vendor dropping support for the installed release.*

- *(EGA.0120: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading SMS software prior to the date the vendor drops security patch support.*

The architecture of SMS allows it to be installed in a number of server configurations. It is possible to combine all the server functions on a single machine or distribute the system roles among a number of network-connected machines. This offers flexibility, but can also create security vulnerabilities. The following items must be considered:

- The Windows account under which the primary SMS server service (SMS Executive) runs must be a member of the Administrators group for the machine on which it runs. Because of the account structure on Windows domain controller machines, the SMS server software would gain domain-wide Administrator privileges if installed on a domain controller.

- The supporting software required for SMS servers includes MS SQL Server and IIS. Installing or enabling these components on a Windows domain controller has negative security implications.

- The SMS Site Server stores most of its data on the Site Database Server. The nature and amount of data transferred between these logical roles, the network port requirements, the authentication activity, and the availability requirement are factors in determining the best configuration of these two server roles.

- The Management Point, Server Locator Point, Reporting Point, and optionally the Distribution Point require that IIS be installed. The use of the Windows LocalSystem account and the access that might be assigned to it on SMS Site Servers can cause concern when IIS also uses this account.

- In an SMS single site configuration using Advanced Clients, but without the Active Directory extensions, the clients authenticate the Site Server using data on the Management Point. If the Site Server and Management Point are combined on a single machine and that machine fails, clients may be unable to authenticate the rebuilt Site Server.

In consideration of these items, there are a number of SMS server configuration requirements related to partitioning the SMS application.

- *(EMS.0040: CAT II) The SMS Administrator will ensure that the SMS Site Server, Site Database Server, and any other Site Systems are not installed on Windows domain controllers.*

- *(EMS.0050: CAT II) The SMS Administrator will ensure that the instance of SQL Server on the SMS Site Database Server is dedicated to SMS.*

- *(EMS.0060: CAT III) The SMS Administrator will ensure that SMS Site System roles that require IIS are not enabled on the Site Server machine.*

- *(EMS.0070: CAT II) The SMS Administrator will ensure that, for single site environments with Advanced Clients and without the Active Directory extensions, the SMS Management Point is not enabled on the Site Server machine.*

In addition to the preceding requirements, the following recommendation is noted:

- The SMS Site Database Server should be installed on the same machine as the Site Server. This varies from normal application partitioning guidance, but the SMS application is an acknowledged exception. Combining these roles reduces exposure of SMS data on the network, reduces authentication transactions, and simplifies security administration.

SMS components use a variety of network ports for communicating among the servers and clients. Microsoft documentation (including Knowledge Base article 826852) provides the following information about TCP/IP port usage by SMS:

Port	Component
80*	Advanced Client to Management Point
135	Administrator Console to servers and clients
137	Remote Control to clients
138	Remote Control to clients
139	Remote Control to clients
389	Site Server to Site Database Server Site Server to child site Proxy Management Point to Site Database Server Advanced Client to Management Point
445	Site Server to Site Database Server Site Server to child site
636	Site Server to Site Database Server Site Server to child site Proxy Management Point to Site Database Server Advanced Client to Management Point
1433	Management Point to Site Database Server
1723	RAS Sender
2701	Remote Tools to Advanced Client
2702	Remote Tools to Advanced Client
2703	Remote Tools to Advanced Client
2704	Remote Tools to Advanced Client
3268	Advanced Client to Management Point

* NOTE: Service Pack 1 for SMS 2003 allows the port used for communications from an Advanced Client to Management Points, Server Locator Points, and BITS-enabled Distribution Points to be changed.

In addition to these, some Windows OS services used by SMS require the following ports to be enabled: 53 (DNS), 67 (DHCP), 135 (RPC), 138 (WINS, NetBIOS), and 139 (NetBIOS). Specific port usage depends on the local network configuration and the SMS components used.

One example of port usage involves Advanced Clients in SMS environments without the Active Directory extensions. In this environment, WINS must be enabled in order for clients to find the Server Locator Point (SLP). Advanced Clients use the SLP to find their assigned Management Point.

The requirements specified under *DOD Instruction 8551.1, Ports, Protocols, and Services Management (PPSM)*, designate several of the ports utilized by SMS components as RED or YELLOW. As described in *Section 3.2.3, Network Access*, specific action is required for applications using these ports across enclave boundaries.

- *(EMS.0080: CAT II) If an SMS configuration is deployed across DOD enclave boundaries and the deployed components utilize PPSM-designated RED or YELLOW ports, the IAO will ensure that the specific SMS configuration has been approved by the Defense Information System Network (DISN) Designated Approving Authorities (DAAs).*

C.1.2.2 Identification and Authentication

This section describes the specific considerations and requirements for SMS that are related to the IA controls in the Identification and Authentication subject area. The following information is addressed:

- Unique SMS administrator and user accounts
- SMS server and client application account options
- Default SMS application accounts

The functions performed by SMS are controlled by the actions of SMS administrators. An SMS administrator is someone authorized for privileged access to the SMS servers, software, and data. Because SMS has broad capabilities to affect the configuration and availability of potentially a large number of client systems, best security practices related to least privilege and separation of duties must be applied to SMS administrator accounts.

Access to some SMS functions may not be privileged. For example, an organization may allow some users to access SMS reporting capabilities. This type of access can be referred to as SMS user access. However, because SMS data could be sensitive, control of SMS user accounts is still required.

In accordance with the *DODI 8500.2, IA Controls, entitled Individual Identification and Authentication*, access to the SMS systems must be gained through an individual identifier.

- *(EGB.0010: CAT II) The IAO will ensure that each SMS administrator or user account is associated with an individual identifier and password.*

SMS does not perform user identification and authentication (I&A) services itself, but relies primarily on services provided by the Windows operating system. Therefore the ESM requirements for applications providing I&A, described in *Section 3.3, Identification and Authentication*, do not apply to SMS.

However, when SMS is configured to use Standard security mode, there is an issue for Windows account password guidance that needs to be addressed for SMS to operate reliably. The following information summarizes this issue:

- Accounts used by the SMS software to access server and client machines are considered to be application accounts.

- *Windows NT/2000/XP Addendum* security guidance requires domains to be configured with an account lockout policy. In some circumstances SMS could trigger this policy, resulting in degraded SMS availability.

- To avoid lockouts, the SMS application accounts should be configured with the *Password never expires* option.

- Passwords for accounts that are automatically created by SMS do not have to be changed by an administrator. The potential risk from this practice is mitigated by the strong password generation capability and the site reset function used by SMS. These accounts are identified in *Section C.1.2.3.2, Account Management*.

- Passwords for SMS application accounts created by the System Administrator must conform to the yearly change requirement for application accounts.

Although the primary access points to SMS functions and data use identification and authentication (I&A) services in the Windows OS logon interface, the Report Viewer web application does not. Refer to *Section C.1.2.3.5, Network Access*, for more information on this interface and the password transmission issue. In accordance with the *DODI 8500.2, IA Controls for Individual Identification and Authentication*, passwords must be encrypted for transmission.

- *(EMS.0090: CAT II) The SMS administrator will ensure that the Reporting Point server is configured to use the HyperText Transfer Protocol Secure (HTTPS) protocol or that an alternate network encryption mechanism is used for user connections to the Report Viewer application.*

SMS maintains elements of a Public Key Infrastructure (PKI) to perform host authentication. Advanced Clients use these elements to authenticate data from a Management Point. Refer to the certificate infrastructure and certificate management topics in the Microsoft document, *Scenarios and Procedures for Microsoft Systems Management Server 2003: Security*, for information. It is not possible to utilize the DOD PKI for this facility and there are no SMS options to control it.

Configured in Standard security mode, SMS uses several application accounts to access server and client machines. The *DODI 8500.2, IA Controls entitled Individual Identification and Authentication*, requires that factory-set, default, or standard user IDs be removed or changed. Although SMS does not permit the names of all its accounts to be changed from the defaults, it is possible to change the most important account used on the Site Server.

The Windows account under which the primary SMS server service (SMS Executive) executes is named SMSService by default. By defining the account before SMS installation and supplying the name to the installation procedure, it is possible to easily avoid using this default.

- *(EMS.0100: CAT III) The SMS Administrator will ensure that the Windows account used for the SMS Executive service is not named SMSService.*

It was noted above that SMS uses some PKI elements for host authentication. As with that PKI implementation, key management within SMS cannot utilize an external Key Management Infrastructure (KMI) and there are no SMS options to control key management.

C.1.2.3 Enclave and Computing Environment

This section describes the specific considerations and requirements for SMS that are related to the IA controls in the Enclave and Computing Environment subject area. Because of the extent of the information in this area, it is organized in the following subsections:

- Data protection
- Account management
- Application Customization
- Auditing
- Network Access

C.1.2.3.1 Data Protection

During installation and operation, SMS software creates directories and files on server and client machines. The assignment of directory and file access permissions for the SMS data is restricted in accordance with the *DODI 8500.2, IA Controls for Access for Need-to-Know and Changes to Data*. Access permissions must be assigned only to process and user accounts that require the associated access to perform designated functions. Privileged system accounts used for administrative functions are allowed full access. Non-privileged accounts, such as the accounts used by SMS users, are allowed execute access to program files and read access to other objects.

- *(EMS.0110: CAT II) The SA will ensure that access to the SMS server data directories and files is restricted in accordance with the permissions in Appendix C.1.3.*

- *(EMS.0120: CAT III) The SA will ensure that access to the SMS client data directories and files is restricted in accordance with the permissions in Appendix C.1.3.*

During installation, SMS software configures some directories on server machines as shared folders. As a defense-in-depth measure, the shared folder permissions are restricted as a supplement to the standard directory and file security. Accounts requiring access are allowed change permission.

- *(EMS.0130: CAT III) The SA will ensure that access to shared SMS folders on SMS servers is restricted in accordance with the permissions in Appendix C.1.3.*

In *Section C.1.1.1, SMS Component Overview*, the use of namespaces to store WMI data was briefly discussed. All Windows systems on which the WMI services are active store WMI data in namespaces. SMS uses this Windows data as well as using WMI services to store additional SMS-specific data. WMI namespaces are organized hierarchically, with the Root namespace at the top. When SMS is installed, the Root\SMS and Root\NetworkModel namespaces are created on the Site Server and the Root\CCM namespace is created on Site Systems and Advanced Clients.

In addition to the security provided for the WMI files by Windows, an additional layer of security is used to protect namespaces at each level in the hierarchy. As with Windows directory security, namespace security can be inherited down through the hierarchy. The possible namespace access privileges include: Execute Methods, Full Write, Partial Write, Provider Write, Enable Account, Remote Enable, Read Security, and Edit Security.

In order to maintain the integrity of Windows and SMS namespace data, access to the WMI namespaces must be restricted. All accounts are allowed **Execute Methods**, **Provider Write**, and **Enable Account** access; the SMS administrators group is also allowed **Remote Enable** access; and privileged system accounts used for administrative functions are allowed all access privileges.

- *(EMS.0140: CAT II) The SA will ensure that access to WMI namespaces on SMS servers and clients is restricted in accordance with the permissions in Appendix C.1.3.*

According to the *DODI 8500.2, IA Controls for Changes to Data*, logging access and changes to SMS data is required for some environments. Logging can be accomplished through two facilities:

- Current Windows operating systems have the capability for file and directory auditing. The generated data specifically identifies the data access attempted. *Windows NT/2000/XP Addendum* security guidance has requirements that address file and directory access logging. Implementation of that guidance is assumed.

- SMS client and server components record process information in individual log files. Although this information is not intended to be a data access log, elements of it can be used for that purpose.

Because SMS component logging is designed as a troubleshooting tool and the output can be voluminous as a result, there are no specific requirements related to data access logging at this time. Organizations implementing SMS should evaluate and enable logging appropriate to their environment.

By virtue of the client/server architecture of SMS, it is given that data will traverse one or more networks as it is collected and processed. When this data reflects detailed configuration information, such as the output of a hardware or software inventory process, that data assumes the sensitivity characteristics of the systems from which it is derived.

The network configuration in which SMS is implemented, combined with the sensitivity of the SMS data, can generate the need for protective measures. The *DODI 8500.2, IA Controls for Encryption for Confidentiality (Data in Transit)*, specify the relevant circumstances.

SMS does not provide for encryption of all data in transit. Therefore an organization deploying SMS is required to take steps such as the implementation of a Virtual Private Network (VPN) to provide encryption when appropriate to the data classification.

- *(EGC.0030: CAT II) The IAO will ensure that SMS data that includes unclassified, sensitive information (including system hardware or software configuration data) that traverses a commercial or wireless network is encrypted, at a minimum, using NIST-certified cryptography.*

- *(EGC.0040: CAT II) The IAO will ensure that SMS data that includes classified systems' hardware or software configuration data that traverses a network cleared to a lower level than the SMS data is encrypted using NSA-approved cryptography.*

C.1.2.3.2 Account Management

Account management spans a number of issues related to the definition of accounts and the assignment of privileges to those accounts. In this section, the following issues and related requirements are addressed:

- Application, administrator, and user accounts
- Account groups
- SMS objects
- SMS Features.

The complex nature of SMS is reflected in the Windows account structure used in SMS operations. The accounts can be grouped in three categories:

- The SMS application accounts are used for server-to-server, server-to-client, and client-to-server access.

- SMS administrator accounts are used by individuals responsible for some set of installation, operation, and maintenance tasks for SMS.

- SMS user accounts are used by individuals designated to receive some level of non-privileged access to SMS features such as reporting.

It must be noted that the applicability of this information varies by product release. The information in this section is based on SMS 2003.

The usage of SMS application accounts is highly dependent on two key options selected for the deployment of SMS. These options were discussed briefly in *Section C.1.1.1, SMS Component Overview*; the important points are:

- The choice of security mode, Standard or Advanced, has significant impact. In Standard mode, Windows user accounts are used to run services and access other systems. In Advanced mode, SMS services run under the Windows LocalSystem security context and access between systems is done using Windows computer accounts.

- The type of clients, Legacy or Advanced, has some impact on account usage. Legacy Clients use privileged Windows user accounts for key tasks such as installing software on clients. Advanced Clients use the Windows LocalSystem security context and the computer account for the same tasks. It must also be noted that, as of the implementation of SMS 2003 Service Pack 1, Microsoft does not support use of the Legacy Client on Windows 2000, Windows XP, or Windows Server 2003.

Managing Windows application accounts is an administrative burden and potential source of vulnerability. If an application account is assigned a weak password and the account is compromised, the confidentiality, integrity, and availability of the application could be negatively impacted. When the account is privileged, entire platforms can be affected. For an ESM application such as SMS, this could be the enabling factor in a major security incident.

If account password maintenance is not carefully coordinated with application execution, the use of an invalid password could quickly trigger the Windows account lockout policy and some function of the application would be disabled. For SMS this could result in the inability to distribute a critical security patch, resulting in a large number of client machines becoming vulnerable to an attack.

In addition to the Windows account management issues and the Microsoft support statement, Legacy Clients do not include the support found in the Advanced Client to authenticate SMS data using digital signatures. These factors make the use of the Legacy Client a poor security practice.

- *(EMS.0150: CAT II) The SMS Administrator will ensure that the SMS Legacy Client is not installed.*

Guidance for the selection of security mode is a more difficult issue. Although Advanced security reduces the number of required accounts, there is a significant prerequisite to its use. Advanced security mode requires that all SMS server machines within a site are in Active Directory and Active Directory is available at all network locations. Because of the planning and coordination required to correctly implement Active Directory, many organizations have not yet fully enabled this technology.

Because of the Active Directory prerequisite, a specific security mode is not required at this time. However, to take advantage of the improved security, organizations should deploy SMS using Advanced security mode when they are able.

Other issues related to SMS application accounts necessitate a discussion of how the accounts are used. The following table describes the accounts used for SMS server connections in a configuration using Standard mode security with Advanced clients.

Windows Account [Default Name]	**Function Notes**
SMS Service [SMSService]	- Used as identity for SMS services on site server - Used to access the site database when it resides on the site server - Used instead of the Site Address and Site System Connection accounts if they are not defined - Creates files and directories on site server
Server Connection [SMSServer_*sc*]	- Used by site system(s) to access the site server - Created by default during site server installation - Multiple accounts could be created by administrator
Site Address	- Used in multi-site configurations for communication between sites - If not defined, SMS Service is used
Site System Database [SMS_SQL_RX_*sc*]	- Used by the site system(s) (Server Locator, Management Point) to access the site database server - Created by default during site server installation
Site System Connection	- Used by the site server to access site system(s) - Not created by default - If not defined, SMS Service is used - Creates files and directories on site system(s)
Remote Service [SMSSvc_*sc…*]	- Used as identity for SMS services on site system(s) - Created when site system role is assigned

- NOTE: The letters *sc* represent the assigned SMS site code.

It is important to note that the SMS software does not require Site Address and Site System Connection accounts. When they are not defined, the SMS Service account is used instead.

To ensure that privileges are appropriately assigned, attention to some details of how these accounts are defined must be considered. These details include:

- The location of the security database where the accounts are defined impacts the scope of the assigned privileges. Accounts defined in Windows Administrator groups on domain controllers have privileges over all the systems within the domain.

- The Windows user groups assigned to the accounts also impact the assigned privileges. Several SMS accounts must be assigned to the Windows Administrators group on specific machines, but others do not require this privilege.

- Some accounts require specific Windows user rights to be assigned.

- An administrator should not change passwords for SMS application accounts that are automatically created and maintained by SMS.

The following table summarizes this information for each account.

Windows Account [Default Name]	**Where Defined**	**Privilege Notes**
SMS Service [SMSService]	Domain	Group membership: - local Administrators – site server Rights: - Log on as a service – site server
Server Connection [SMSServer_*sc*]	Local - site server	Group membership: - Site System to Site Server Connection Passwords: - Not to be changed by administrator – only default accounts
Site Address	Local - site servers	Group membership: Site to Site Connection
Site System Database [SMS_SQL_RX_*sc*]	Local - site database server	Group membership: - Site System to SQL Server Connection Password: - Not to be changed by administrator
Site System Connection	Local - site system	Group membership: - local Administrators – site system
Remote Service [SMSSvc_*sc…*]	Local - site system	Group membership: - local Administrators – site system Rights: - Log on as a service – site system Password: - Not to be changed by administrator

- NOTE: The letters *sc* represent the assigned SMS site code.

As indicated in the table, most of the SMS accounts can be defined on local systems where they are used. This helps to further reduce the attack surface that is presented by these accounts. The notable exception to this strategy is the SMS Service account. This account is used in several contexts (such as SMS Discovery) that require Windows domain access.

In consideration of this information and to ensure that the *DODI 8500.2, IA Controls for Least Privilege*, is followed, requirements to reduce the access privileges of SMS application accounts are defined.

- *(EMS.0160: CAT II) The SA will ensure that none of the following SMS accounts are defined in any Windows administrator-level group on a domain controller: SMS Service, Server Connection, Site Address, Site System Database, Site System Connection, and Remote Service.*

- *(EMS.0170: CAT II) The SMS Administrator will ensure that, for supported functions, Site System Connection and Site Address accounts are used instead of the SMS Service account.*

In addition to the SMS application accounts for server connections already discussed, there are two accounts for communication between servers and Advanced Clients. The following table describes these accounts.

Windows Account	**Function Notes**
Advanced Client Network Access	- Used by clients to access shares on non-SMS servers - Used for Client Push installation in NT domains
Client Push Installation	- Used by the Client Push installation method to install software on clients - If not defined, SMS Service is used.

The need for these accounts depends on the environment and the configuration of the Windows clients. The Advanced Client Network Access account would not be needed where the user accounts utilized on clients already have access to any required network shares. It may be required if SMS Roaming is being utilized. The Client Push Installation account would not be needed if Client Push Installation is not used, users already have administrative rights, or a different strategy is used to manage the privileges required for product installations.

For the same reasons noted for the other SMS application accounts, the location where the accounts are defined and the Windows user group membership impact the privileges assigned. The following table summarizes some of the account definition details for these accounts.

Windows Account	**Where Defined**	**Privilege Notes**
Advanced Client Network Access	Domain	Group membership: - Domain Users
Client Push Installation	Domain or Local – client	Group membership: - Domain Admins or - [local] Administrators – client

It is important to note that the Client Push Installation account may have Windows administrator-level privileges over a wide span of clients. To mitigate the risk associated with this definition, organizations should consider a plan to selectively disable this account for periods when it is not being used.

In consideration of this information and to ensure that the *DODI 8500.2, IA Controls for Least Privilege*, is followed, the following requirements to reduce the access privileges of SMS application accounts are defined.

- *(EMS.0180: CAT II) The SA will ensure that the SMS Advanced Client Network Access account is not defined in any Windows administrator-level group on a domain controller*.

- *(EMS.0190: CAT II) The SMS Administrator will ensure that, for supported functions, a Client Push Installation account is used instead of the SMS Service account.*

Based on other requirements, there are certain SMS application accounts that are prohibited. The source of these considerations is:

- The *Database Security Technical Implementation Guide* requires the use of Windows authentication for Microsoft SQL Server.

- An earlier requirement in this section prohibits the use of the Legacy Client.

- *(EMS.0200: CAT II) The SMS Administrator will ensure that internal Microsoft SQL Server accounts (such as the sa pseudo database account) are not used as an SMS Site Database account.*

- *(EMS.0210: CAT III) The SA will ensure that the following SMS Legacy Client accounts are not defined or are disabled: Client Connection, Client Services (DC), Client Services (non-DC), Client User Token (DC), Client User Token (non-DC), CCM Boot Loader (DC), CCM Boot Loader (non-DC), and Legacy Client Software Installation.*

  - NOTE: The letters DC represent domain controller.

It should be noted that the Client Connection account is automatically created during SMS installation and must be manually deleted or disabled.

The second category of Windows accounts defined for SMS are SMS administrator accounts. Because of the level of control that SMS has over client machines, SMS administrators are effectively System Administrators (SAs) for all the clients. This fact underscores the need to document and control the assignment of SMS administrator privileges.

The following considerations should be noted concerning the Windows account used during SMS installation:

- This account requires local Windows Administrator privileges on the site server.
- This account automatically becomes an SMS administrator that is granted control over all SMS objects.

The final category of Windows accounts defined for SMS are SMS user accounts. As noted earlier, an SMS user account is an account used by individuals for non-privileged access to an SMS feature such as reporting. Although this level of access does not imply control over SMS, it does imply access to SMS data. As such, this access must also be documented and controlled.

Guidelines are necessary in order to manage SMS access privileges for all three of the SMS account types: application, administrator, and user. While any processes that implement these guidelines do not have to be specific to SMS accounts, the nature of SMS makes it particularly important to ensure these guidelines are followed to manage those accounts.

- *(EGC.0080: CAT II) The IAM will ensure that a process is documented and implemented for the management of SMS accounts.*

- *(EGC.0090: CAT II) The IAM will maintain documentation of the assignment of SMS administrator and user accounts.*

- *(EGC.0050: CAT II) The IAO will ensure that SMS application accounts are assigned the minimum privileges required.*

- *(EGC.0050: CAT II) The IAO will ensure that SMS administrator accounts are assigned the minimum privileges required for the user's job function, as indicated in the local account documentation.*

- *(EGC.0060: CAT III) The IAO will ensure that SMS administrator accounts are not used for non-privileged functions.*

During the installation of SMS, some Windows user groups are automatically created. Access to SMS privileges and data can be assigned to groups that represent roles associated with SMS operations. Using these groups in a role based access control strategy can be more efficient and secure than assigning access rights to individual accounts. The following table describes these groups.

Windows Group [Default Name]	**Function Notes**
SMS Administrators [SMS Admins]	- Used for SMS administrator accounts - The account used during the installation of SMS on the site server is placed in this group by default.
Site System to Site Server Connection [SMS_SiteSystemToSite ServerConnection_*sc*]	- Used for Server Connection account(s) - The default Server Connection account, SMSServer_*sc*, is placed in this group by default.
Site System to SQL Server Connection [SMS_SiteSystemToSQL Connection_ *sc*]	- Used for Site System Database account(s) - The default Site System Database account, SMS_SQL_RX_*sc*, is placed in this group by default.
Site to Site Connection [SMS_SiteToSite Connection_*sc*]	- Used for Site Address account(s)
Reporting Users [SMS Reporting Users]	- Used for users authorized to access SMS reporting

- NOTE: The letters *sc* represent the assigned SMS site code.

The location where these groups are defined and the members in the group are security considerations. The following table describes some of the definition details for these groups:

Windows Group [Default Name]	**Where Defined**	**Group Members**
SMS Administrators [SMS Admins]	Local - site server	Group members: - Users with SMS administrative privileges
Site System to Site Server Connection [SMS_SiteSystemTo SiteServerConnection_*sc*]	Local - site server	Group members: - Server Connection account(s) - Site Server (except Distribution Point) computer account(s)
Site System to SQL Server Connection [SMS_SiteSystemTo SQLConnection_ *sc*]	Local - site database server	Group members: - Site System Database account(s)
Site to Site Connection [SMS_SiteToSite Connection_*sc*]	Local - site servers	Group members: - Site Address account(s)
Reporting Users [SMS Reporting Users]	Local - Reporting Point	Group members: - Non-privileged users with SMS reporting privileges

Because membership in a Windows user group implicitly provides some type of access, the members of the group must be controlled.

- *(EMS.0220: CAT II) The IAO will ensure that only documented SMS administrator accounts are members of the SMS Administrator group.*

- *(EMS.0230: CAT II) The SA will ensure that only SMS application accounts or SMS server computer accounts are members of the Site System to Site Server Connection, Site System to SQL Server Connection, and Site-to-Site Connection Windows groups.*

- *(EMS.0240: CAT II) The SA will ensure that only accounts of documented SMS reporting users are members of the Reporting Users group.*

Based on the earlier requirement in this section that prohibits the use of the Legacy Client, there is a Windows group that is not needed and is therefore also prohibited.

- *(EMS.0250: CAT III) The SA will ensure that the Legacy Clients Internal Client Group Windows group is not defined.*

SMS objects represent logical elements used in the course of SMS operations. As with other technologies that use object-oriented design, SMS defines specific object classes. An object instance is an individual item that is a member of an object class. A consequence of this design is that actions on an object can be controlled using the context of a class or instance. For example, an SMS administrator may be permitted access to change configuration options for all SMS Collections or only for specific Collections.

The following object classes are defined in SMS:

- Advertisement
- Collection
- Package
- Query
- Report
- Site
- Software Metering Rule
- Status Message

SMS allows objects to be secured through the definition of object security rights. An object right represents the ability to perform certain actions on an object. For most SMS objects, rights can be controlled at both the class and instance level. So it is possible to permit a user or group of users the right to perform a certain action on all the objects in a class or only on certain object instances. This architecture enables a role based access control strategy to be applied to object permissions.

The following table summarizes the SMS object rights and the object classes to which they apply.

Object Right	**Adver-tisement**	**Collec-tion**	**Packge**	**Query**	**Report**	**Site**	**Software Metering Rule**	**Status Msg**
Administer	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**
Advertise		**X**						
Create	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**
Delegate	**X**	**X**	**X**	**X**	**X**	**X**	**X**	
Delete	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**
Delete Resource		**X**						
Distribute			**X**					
Manage SQL Commands						**X**		
Manage Status Filters						**X**		
Meter						**X**		
Modify	**X**	**X**	**X**	**X**	**X**	**X**	**X**	
Modify Resource		**X**						
Read	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**
Read Resource		**X**						
Use Remote Tools		**X**						
View Collected Files		**X**						

Please refer to the *Microsoft Systems Management Server 2003 Concepts, Planning, and Deployment Guide*, for detailed information on SMS objects and object rights.

An example of Collection object security shows how SMS object security can be used. One instance of an SMS Collection object might be a group of client machines that are related by user organization. All client machines that belong to the Johnson Logistics group could be in a Collection named JHN-Log. By defining instance security for the JHN-Log Collection, a System Administrator for that group could be allowed to control SMS functions for only that group of clients.

The following characteristics apply to SMS object security:

- The SMS Administrator Console is the primary tool used to assign SMS object permissions.
- Class-level rights apply to all object types within the class.
- Instance-level rights apply to specific instances of an object type.
- Windows accounts that have object security rights must also have access to the related SMS WMI namespace.
- The account used to install SMS and the LocalSystem security context on the site server are granted permissions to all SMS objects by default.

Assigning SMS object access permissions to specific Windows accounts or groups represents a privilege assignment that must be documented and controlled to ensure that it remains limited.

- *(EGC.0090: CAT II) The IAM will maintain documentation of the assignment of SMS object rights.*
- *(EGC.0050: CAT II) The IAO will ensure that only documented SMS object rights are assigned to SMS administrators.*

Because SMS implementation can vary significantly, it would be unrealistic to mandate requirements for all the object rights in every environment. However, the following specific considerations are relevant:

- There must always be at least one account with the class-level **Administer** right for an object class.
- Users who create an instance of an object are automatically assigned **Read**, **Modify**, and **Delete rights** for that object.
- Creating local Collection objects and applying instance security provides a way to scope administrative span of control.
- The **Delegate** object right allows a user or group to grant rights to other users for the objects that the first user or group creates.
- The **Delete** object right for Status Message objects could allow a user or group to prematurely delete audit data.
- The **Manage SQL Commands** object right allows SQL Server commands to be created through the SMS Administrator console. These commands have complete privileges to the SMS site database.

In view of these considerations, the following guidance is recommended with respect to assigning SMS object rights:

- Rights should be assigned at the instance level when possible. This is particularly desirable for Collection objects.

- The Delegate right should be assigned on a limited basis.

- The Delete right for Status Message objects should be assigned on a limited basis and at the individual account rather than group level.

- The Manage SQL Commands right should be assigned on a limited basis and at the individual account rather than group level.

There is one object right for which a specific requirement is necessary. The **Use Remote Tools** right controls the right to use the SMS Remote Tools feature on the Collection object class and instances. Because Remote Tools is a significant issue from the security perspective, some brief comments are required.

Remote Tools is an SMS feature composed of several tools designed for help desk assistance and troubleshooting support. The tools provide full control over the client machine, allowing an administrator to perform operations remotely as if they were physically present at the machine. The Remote Tools suite consists of: Remote Control, Remote Reboot, Remote Chat, Remote File Transfer, Remote Execute, SMS Client Diagnostics, and Ping Test.

Although Remote Tools offers a powerful capability, there are security issues in the implementation that make the use of the feature unacceptable. The problems include:

- Control of Windows administrative privileges may be bypassed for the tools user. The Microsoft *Scenarios and Procedures for Microsoft Systems Management Server 2003: Security* document states, “SMS Remote Tools also allow a non-administrator to execute programs in an administrative context, if they have sufficient permission to remote tools.”

- Control of Windows administrative privileges may be bypassed for the client user. The *Microsoft Systems Management Server 2003 Operations Guide* states, “When an administrator uses Remote Execute to perform operations on the client, the user who is logged on to the client will also have elevated permissions and can then gain access to the same directories and files as the administrator.” To aggravate this problem, if a network session failure occurs, this privilege elevation would not be terminated.

- SMS Collection security may be bypassed. The Remote.exe tool used to establish a Remote Tools connection includes a command line option (SMS:NOSQL). This option allows a connection to the client without accessing the SMS Site Database.

In view of these serious security issues, the use of Remote Tools must be prohibited. Access to the tools via the Remote.exe program is addressed through the file permission requirement specified in *Section C.1.2.1, Security Design and Configuration*. The following requirement addresses the Use Remote Tools object right.

- *(EGC.0050: CAT II) The SMS administrator will ensure that the Use Remote Tools right is not granted to any account.*

C.1.2.3.3 Application Customization

Even though SMS is a highly configurable product, there may be instances in which an organization determines that the way in which a specific component functions does not meet the needs of the organization. In this case, a decision might be made to modify SMS functions by replacing or adding programs.

Microsoft offers an SMS software development kit (SDK) that includes "…documentation and sample applications for developing SMS applications that integrate with, or use, the SMS Administrator Console, SMS Provider (server-side applications), and client-side applications." This SDK provides tools to assist in modifying the designed function of SMS.

Because of the elevated privileges used during many SMS operations, the installation and propagation of improperly modified SMS functions could cause serious compromises of the confidentiality, integrity, or availability of a single server or a large number of client machines. Adherence to a configuration management process can help to mitigate this risk.

- *(EGC.0110: CAT III) The IAO will ensure that any modified or added programs installed in an SMS configuration and used by a user or process with elevated privileges has been processed through a documented configuration management (CM) process.*

C.1.2.3.4 Auditing

Auditing for an information system is defined as a way to assess the adequacy of system controls and the degree of compliance with policies and procedures. To enable auditing, data must be continually collected. Because of the possible impact of SMS operations on the security posture of many client systems, maintaining audit data for SMS is essential.

To a large extent, SMS relies on the auditing support provided by the Windows operating system on which the SMS components execute. The application, security, and system logs maintained by Windows, when configured according the Windows OS security guidance, enable auditing of SMS account and data access.

The SMS software includes a Status System with features that supplement the audit functions in Windows. The Status System monitors SMS activity for a site and generates status messages.

The following characteristics are relevant to security:

- Status System messages are identified with a message severity level of error, warning, or informational.

- Status System messages are identified with a message type of milestone, detail, or audit.

- Audit messages provide an audit trail of actions taken in the SMS Administrator console that result in objects being added, modified, or deleted. All audit messages have a severity level of informational.

- Configuration options are available to control how status messages are processed for display and how long they are retained.

- Status filter rules can be configured to determine processing options such as disposition for status messages.

- The system includes status summarizers (Component, Site, Package, Advertisement) that provide a quick method of establishing the operational state of SMS.

- SMS can be configured to replicate status messages, status summaries, both, or neither up in the SMS hierarchy to the parent site.

The SMS Component Status and Site System Status summarizers can be used by administrators to detect of a loss of availability that may be due to a system compromise. For this reason, these components or an effective alternate need to be enabled.

- *(EMS.0260: CAT III) The SMS administrator will ensure that the Component Status and Site System Status summarizers are enabled or that an alternate on-line monitoring capability is maintained.*

- *(EMS.0270: CAT II) The SMS administrator will ensure that the Component Status and Site System Status summarizers or their alternate are regularly reviewed for indications of inappropriate or unusual activity.*

SMS Status System audit data may be instrumental in determining the source of a compromise of the SMS application. In consideration of this fact and to ensure that the *DODI 8500.2, IA Controls for Audit Record Retention,* is followed, steps must be taken to ensure that audit messages are retained.

- *(EMS.0280: CAT II) The SMS administrator will ensure that status filter rules and the Delete Aged Status Messages maintenance task are configured so that audit status messages are not deleted until they have been backed up.*

- *(EGC.0130: CAT II) The SMS administrator will ensure that SMS audit data is retained for at least one year.*

- *(EGC.0170: CAT II) The SMS administrator will ensure that SMS audit data is backed up not less than weekly onto a different system or media than the system on which the SMS server executes.*

C.1.2.3.5 Network Access

Most SMS functions are performed by interactions of servers and clients over network connections. This access generally occurs in one of the following ways:

- SMS Administrators access administrative controls for the SMS server software.

- SMS Reporting users access an SMS Reporting Point server to review SMS data.

- SMS application software on client machines accesses servers to send or receive status or configuration data.

- SMS application software on server machines accesses other servers to communicate status or configuration data.

It was noted in *Section C.1.2.2, Identification and Authentication*, that SMS does not perform user identification and authentication (I&A) services itself, but relies primarily on services provided by the Windows operating system. Therefore the ESM requirements for applications providing I&A, described in *Section 3.4.5*, *Network Access,* do not apply to SMS.

This section describes the considerations and requirements related to the security of these network access scenarios that are not specifically related to I&A services.

SMS administrators access the SMS server administrative controls through the SMS Administrator console. This client application is installed on the Site Server and on client machines used by SMS administrators. Due to the physical access restrictions for the SMS servers, access using a client machine would be more common. The administrator's access credentials, which would have been authenticated by the Windows OS at logon time, are passed to the Site Server for access.

To access SMS reports, users invoke the web browser on their client machine to access the Report Viewer web application on the SMS Reporting Point site system. The application uses IIS as the web server platform. Users are required to enter their Windows account and password to be authenticated for access.

Access to the web server on the Reporting Point site system could be from any browser client with network connectivity. This connection might represent access to a Government information system from a non-Government client. In consideration of the *DODI 8500.2, Warning Messages IA Controls*, privacy and security notices must be displayed to users.

- *(EMS.0290: CAT III) The SMS administrator will ensure that the web server on the Reporting Point site system is configured to present a warning banner advising the user that:*

  - *The system is a DOD system.*
  - *The system is subject to monitoring, recording, and auditing.*
  - *Monitoring is authorized in accordance with applicable laws and regulations and conducted for purposes of systems management and protection, protection against improper or unauthorized use or access, and verification of applicable security features or procedures.*
  - *Use of the system constitutes consent to monitoring.*
  - *The system is for authorized US Government use only.*

Until the implementation of Service Pack 1 for SMS 2003, it is not possible to configure web server-based (e.g., SSL) encryption for the Reporting Point. Without such a capability, user passwords are transmitted in the clear when authenticating to the Reporting Point system. Please refer to *Section C.1.2.2, Identification and Authentication*, for the required action that addresses this issue.

The sensitivity of SMS data transmitted between servers and clients and the detail configuration data retained in the SMS Site Database would generally be classified at a level that reflected the data on the client machines. Although it might be technically possible for SMS to manage clients operating at different classification levels over a controlled interface, this has not been evaluated and has not been determined an acceptable configuration.

There are also potential network security issues when an SMS site hierarchy spans DOD and non-DOD networks. The ability for such SMS configurations to conform to the *DODI 8500.2, IA Controls for Interconnections among DOD Systems and Enclaves,* has not been established.

- *(EGC.0220: CAT II) The IAM will ensure that an SMS site hierarchy is not implemented across DOD information systems operating at different classification levels or across DOD and non-DOD systems or networks.*

As with any application host, the SMS servers are possible targets for attack. Because of the potential damage to the server and possibly a large number of clients that could result from a successful attack, the importance of detecting an attack on an SMS server is high. As described in the *Enclave Security STIG*, a host-based intrusion detection system (HIDS) provides this support.

- *(EGC.0230: CAT II) The IAO will ensure that SMS servers are protected by a host-based intrusion detection system.*

The loss of integrity for SMS data transmitted between servers and clients can result in a number of problems. Data corruption in the SMS Site Database could result in invalid information being used to administer clients. It could be possible for clients to receive malicious code or data that would cause a loss of their confidentiality, integrity, or availability characteristics.

SMS provides mechanisms to help ensure that transmission integrity is maintained. These mechanisms are based on the use of PKI elements that support the use of digital signatures.

- SMS 2003 signs and authenticates data sent between sites using internally managed private/public encryption key pairs. Or environments with mixed SMS 2.0 and 2003 releases, an option exists to reject unsigned data from SMS 2.0 sites.

- Multiple methods are available for transmitting keys between sites. When the automatic method is used, a secure exchange option can be used to ensure the integrity of the keys. The manual method involves the use of the Hierarchy Maintenance tool (PreInst.exe) to unload and reload the keys.

Using the SMS integrity mechanisms helps to prevent data corruption and to deter some types of network attacks. This use conforms to the *DODI 8500.2, IA Controls for Transmission Integrity Controls*.

- *(EMS.0300: CAT II) The ESM administrator will ensure that the "Do not accept unsigned data from sites that are running SMS 2.0 SP4 and earlier" Site Properties option is enabled.*

- *(EMS.0310: CAT II) The ESM administrator will ensure that the "Require secure key exchange between sites" Site Properties option is enabled.*

Restrictions on access to the Hierarchy Maintenance tool are specified in *Section C.1.2.1, Security Design and Configuration*.

Another transmission integrity mechanism that SMS can take advantage of is server message block (SMB) signing. This facility ensures that data transferred using the SMB protocol is not altered during transmission. SMB signing is provided by the Windows OS platform and is enabled when the requirements in the *Windows NT\2000\XP Addendum* document and the associated NSA guides that address OS security configuration are followed.

C.1.2.4 Enclave Boundary Defense

This section describes the specific considerations and requirements for SMS that are related to the IA controls in the Enclave Boundary Defense subject area.

An organization may determine that remote access to administrative functions in SMS is necessary to maintain the availability of the SMS configuration. This access would probably involve the use the SMS Administrator console application. Access to the SMS Report Viewer web application might also be deemed necessary for non-privileged users.

The requirements described in *Section 3.5, Enclave Boundary Defense*, apply to remote access to SMS and there are no unique considerations specific to SMS. The requirements address the *DODI 8500.2, IA Controls for Remote Access for Privileged Functions, Remote Access for User Functions, and VPN Controls.*

- *(EGD.0010: CAT II) The IAO will ensure that remote access to SMS applications is secured through the following:*

    - *Use of a managed access control point such as a remote access server in a DMZ*
    - *Session encryption using, according to the data classification, NIST-certified or NSA-approved cryptography*
    - *Strong user authentication that resists spoofing, such as a two-factor system.*

- *(EGD.0020: CAT II) The IAO will ensure that remote access for SMS administrators uses:*

    - *Session security measures such as a VPN configured in blocking mode to discard all but authorized traffic*
    - *A process that creates an audit log for each remote session.*

- *(EGD.0030: CAT II) The IAM/IAO will review the audit log for every remote session of an SMS administrator.*

- *(EGD.0040: CAT II) The IAO will ensure that VPN traffic for remote SMS administrator sessions is visible to a network intrusion detection system (IDS).*

C.1.2.5 Physical and Environmental

This section describes the specific considerations and requirements for SMS that are related to the IA controls in the Physical and Environmental subject area.

The functions performed by SMS applications require elevated privileges to be used during operations. The proper use of some of these functions can have a significant, positive impact on the client machines that are served. A primary example of this impact would be the deployment of vendor security patches or updated antivirus signatures that would enhance the security of the client machines.

An improper use of SMS functions could have a proportional or more significant negative impact on client machines. Compromise of programs or data sent to clients could ultimately lead to a serious loss of the confidentiality, integrity, or availability characteristics of a large number of client machines.

In recognition of these significant possible impacts, attention to physical access restrictions for the SMS servers is more important than for typical application hosts. As a result, adherence to the *DODI 8500.2, IA Controls for Access to Computing Facilities*, is essential.

- *(EGE.0010: CAT II) The IAO will ensure that physical access to SMS application hosts is restricted to specifically authorized personnel.*

C.1.2.6 Continuity

This section describes the specific considerations and requirements for SMS that are related to the IA controls in the Continuity subject area.

SMS can be used to provide business or mission essential functions such as the deployment of vendor security patches or antivirus signature updates. In this case, a loss of SMS availability could be a serious concern because a large number of client machines might be left vulnerable to some form of attack.

To ensure that the availability of SMS is maintained and in accordance with the *DODI 8500.2, IA Controls for Disaster and Recovery Planning and Identification of Essential Functions,* requirements for recovery and restoration planning are needed.

- *(EGF.0010: CAT II) The IAM will ensure that the disaster recovery plan for the enclave includes appropriate provision for the continuity of SMS if it is used to provide essential information assurance functions such as security patch management or antivirus signature deployment.*

    - *For SMS applications serving MAC III systems, resumption within five days of activation*
    - *For SMS applications serving MAC II systems, resumption within 24 hours of activation*
    - *For SMS applications serving MAC I systems, transfer to an alternate site for the duration of an event with little or no loss of operational continuity.*

- *(EGF.0020: CAT II) The IAM will ensure that any SMS configuration that provides essential information assurance functions such as security patch management or antivirus signature deployment is identified for priority restoration planning.*

Having a backup of SMS data is critical whether recovering from ordinary hardware problems, unexpected software errors, or a major computing facility event. The considerations relative to SMS data backup are as follows:

- The strategy and tools used to back up SMS data are directly related to how well an SMS configuration can be recovered. Because SMS data spans a number of individual files as well as tables residing in the Site Database, a specific tool must be used to get properly synchronized data in the backup.

- The frequency with which data is backed up is directly related to the difficulty or sometimes possibility of resuming operations.

The SMS product includes a Backup SMS Site Server task to properly capture SMS data for backup. It is described in the *Microsoft Systems Management Server 2003 Operations Guide*. This tool has the following characteristics:

- It is defined as one of the SMS Maintenance tasks and is disabled by default.
- It executes under the SMS_SITE_BACKUP service and stops basic SMS services on the Site Server while the backup is in progress.
- Its input includes the Site Database, SMS data files, registry keys, and system configuration information.
- Its input does not include data on Site Systems as that data could be re-created by the Site Server.
- Its output is referred to as the backup snapshot, consists of a number of directories and files, and is stored under a parent SMS backup directory specified by the administrator.
- The SMSbkup.ctl file contains site-specific configuration information for the task.
- The AfterBackup.bat file is an optional file that allows an administrator to specify commands to execute when the task completes. This could be used to create an archival copy of the backup data.
- It does <u>not</u> create a complete backup of the Site System or Site Servers and therefore must be used in conjunction with another backup tool.

The following requirements are designed to ensure that SMS data is backed up properly and that the backup data is protected in accordance with the *DODI 8500.2, IA Controls for Data Backup Procedures*.

- *(EGF.0030: CAT II) The SMS Administrator will ensure that SMS data is managed appropriately to the MAC of the systems served by the ESM application.*

  - *For SMS configurations managing MAC III client systems, SMS data is backed up at least weekly.*

  - *For SMS configurations managing MAC II client systems, SMS data is backed up daily and the recovery media is stored at an off-site location that affords protection in accordance with the mission assurance category and confidentiality level of the data.*

  - *For SMS configurations managing MAC I client systems, SMS data is backed up by maintenance of a redundant secondary system, not collocated, that can be activated without loss of data or disruption to the operation.*

- *(EMS.0320: CAT II) The SMS Administrator will enable the Backup SMS Site Server task as part of the backup process for the SMS configuration.*

- *(EMS.0330: CAT II) The SMS Administrator will ensure that logging is enabled for the SMS_SITE_BACKUP service.*

- *(EMS.0340: CAT II) The SMS Administrator will ensure that the parent SMS backup directory is not located on the same logical drive partition as the parent SMS data directory.*

- *(EMS.0350: CAT II) The SA will ensure that access to the SMS backup directories and files is restricted in accordance with the permissions in Appendix C.1.3.*

Copies of the SMS software could be critical when attempting to continue or resume operations after a disruptive event. If the operational servers and the original SMS installation materials are lost or inaccessible, installation of a backup copy of the software may be the only way to recover in the required timeframe. Protecting this copy is in accordance with the *DODI 8500.2, IA Controls for Backup Copies of Critical Software*.

- *(EGF.0040: CAT II) The IAO will ensure that backup copies of software for any SMS configuration that provides essential information assurance functions such as security patch management or antivirus signature deployment are stored in a fire-rated container or otherwise not collocated with the operational software.*

C.1.2.7 Vulnerability and Incident Management

This section describes the specific considerations and requirements for SMS that are related to the IA controls in the Vulnerability and Incident Management subject area.

As with any application, it is possible that new vulnerabilities will be discovered in SMS components. Continued vigilance and quick action to implement security patches are the chief means to deter attacks based on the exploitation of product vulnerabilities.

Section 3.8, *Vulnerability and Incident Management*, describes the need for a vulnerability management process. Requirements that ensure utilization of the process are necessary to comply with the *DODI 8500.2, IA Controls for Vulnerability Management*.

- *(EGG.0010: CAT II) The IAM will ensure that a vulnerability management process that encompasses SMS applications and server hardware is documented and implemented.*

- *(EGG.0020: CAT II) The IAO will ensure that all security related patches to SMS are applied and that completion is documented for each applicable asset.*

The current security patches applicable to SMS can be researched through Microsoft's Security Bulletin Search web site at http://www.microsoft.com/technet/security/current.aspx.

C.1.3 Systems Management Server Permissions

C.1.3.1 Windows Directory and File Permissions

The following notation is used in this section:

- [*SMS Folder*] – the high level directory structure that holds the subordinate directories in which SMS software programs and data reside
- [*SMS Data Folder*] – the high level directory structure that holds the subordinate directories in which the SMS SQL database resides
- [*SMS Toolkit Folder*] – the directory in which the SMS Toolkit software is installed
- [SITE_BACKUP_DESTINATION] – the directory in which the SMS_SITE_BACKUP service places SMS data backup files for the Site Server and the Site Database Server
- %SystemRoot% - the directory in which Windows is installed, e.g. C:\winnt
- [SMS Limit. Admins] – one or more groups of SMS administrators with a limited scope of authority
- [SQL Server account] – the account under which SQL Server executes
- [anonymous web account] – the account set up for anonymous access to IIS – commonly IUSR_*computername*
- [web application account] – the account set up for web applications – commonly IWAM_*computername*
- [*sc*] – the assigned SMS site code.

Object	Account Assignment	Permissions	STIG Reference
…\[*SMS Folder*]	Administrators SYSTEM SMS Admins SMS_SiteSystemTo SiteServerConnection_*sc* [SMS Limit. Admins]	Full Control Full Control Full Control Read & Execute Full Control	C.1.2.1 C.1.2.3
…\[*SMS Folder*]\bin\…\ACLreset.exe	Administrators SYSTEM	Full Control Full Control	C.1.2.1
…\[*SMS Folder*]\bin\…\PreInst.exe	Administrators SYSTEM	Full Control Full Control	C.1.2.1
…\[*SMS Folder*]\bin\…\Remote.exe	Administrators SYSTEM	Full Control Full Control	C.1.2.1
…\[*SMS Data Folder*]	Administrators SYSTEM SMS Admins SMS_SiteSystemTo SQLConnection_ *sc* [SQL Server account]	Full Control Full Control Full Control Full Control Full Control	C.1.2.3
…\[*SMS Toolkit Folder*]	Administrators SYSTEM SMS Admins [SMS Limit. Admins]	Full Control Full Control Full Control Full Control	C.1.2.1
[SITE_BACKUP_DESTINATION]	Administrators SYSTEM SMS Admins	Full Control Full Control Full Control	C.1.2.6
…\SMS_CCM	Administrators SYSTEM INTERACTIVE [anonymous web account] [web application account]	Full Control Full Control Read & Execute Read & Execute Read & Execute	C.1.2.3

Object	Account Assignment	Permissions	STIG Reference
…\SMSADMIN	Administrators SYSTEM SMS Admins [SMS Limit. Admins]	Full Control Full Control Full Control Full Control	C.1.2.1
…\SMSADMIN\bin\…\Remote.exe	Administrators SYSTEM	Full Control Full Control	C.1.2.1
…\SMSADMIN\bin\…\PreInst.exe	Administrators SYSTEM	Full Control Full Control	C.1.2.1
…\SMSADMIN\bin\…\ACLreset.exe	Administrators SYSTEM	Full Control Full Control	C.1.2.1
%SystemRoot%\system32\ccmcore.dll	Administrators SYSTEM INTERACTIVE [anonymous web account] [web application account]	Full Control Full Control Read & Execute Read & Execute Read & Execute	C.1.2.1
%SystemRoot%\system32\SMSAccountSetup.ini	Administrators SYSTEM SMS Admins	Full Control Full Control Full Control	C.1.2.1
%SystemRoot%\system32\CCM	Administrators SYSTEM INTERACTIVE	Full Control Full Control Read & Execute	C.1.2.1

C.1.3.2 Windows Shared Folders Permissions

The following notation is used in this section:

- [*sc*] – the assigned SMS site code.

Object	Account Assignment	Permissions	STIG Reference
CAP_[*sc*] *	Administrators	Full Control	C.1.2.3
SMS_[*sc*]	Authenticated Users Domain Computers	Change Change	C.1.2.3
SMS_Site	Authenticated Users Domain Computers	Change Change	C.1.2.3
SMS_SUIAgent	Authenticated Users Domain Computers	Change Change	C.1.2.3
SMSClient	Authenticated Users Domain Computers	Change Change	C.1.2.3

* - The CAP_[*sc*] share is not used in SMS configurations having only Advanced Clients.

C.1.3.3 WMI Namespace Permissions

The following notation is used in this section:

- [all] – all available permissions
- [SMS Limit. Admins] – one or more groups of SMS administrators with a limited scope of authority

Namespace	Account Assignment	Permissions	STIG Reference
Root	Administrators	[all]	C.1.2.3
	Authenticated Users	Execute Methods Provider Write Enable Account	
Root\CCM	Administrators	[all]	C.1.2.3
	Authenticated Users	Execute Methods Provider Write Enable Account	
Root\CIMV2	Administrators	[all]	C.1.2.3
	Authenticated Users	Execute Methods Provider Write Enable Account	
Root\SMS	Administrators	[all]	C.1.2.3
	Authenticated Users	Execute Methods Provider Write Enable Account	
	SMS Admins	Execute Methods Provider Write Enable Account Remote Enable	
	[SMS Limit. Admins]	Execute Methods Provider Write Enable Account Remote Enable	

Namespace	Account Assignment	Permissions	STIG Reference
Root\NetworkModel	Administrators Authenticated Users	[all] Execute Methods Provider Write Enable Account	C.1.2.3

APPENDIX D. LIST OF ACRONYMS

ACF	Adapter Configuration Facility
ADE	Application Development Environment
AEF	Application Extension Facility
API	Application Programming Interface
ARF	Application Registration File
BDT	Bulk Data Transfer
BITS	Background Intelligent Transfer Service
C&A	Certification and Accreditation
CAP	Client Access Point
CCB	Configuration Control Board
CCITT	International Telegraph and Telephone Consultative Committee
CDS	Class Definition Statement
CIM	Common Information Model
CIO	Chief Information Officer
CLI	Command Line Interface
CM	Configuration Management
CMIP	Common Management Information Protocol
CMOT	CMIP Over TCP
CNSS	Committee on National Security Systems
CORBA	Common Object Request Broker Architecture
COTS	Commercial-Off-the-Shelf
DAA	Designated Approving Authority
DBMS	Database Management System
DC	Domain Controller
DEN	Directory Enabled Networks
DHCP	Dynamic Host Configuration Protocol
DISA	Defense Information Systems Agency
DISN	Defense Information Systems Network
DM	Distributed Monitoring
DMI	Desktop Management Interface
DMTF	Distributed Management Task Force
DMZ	Demilitarized Zone
DOD	Department of Defense
DODI	DOD Instruction
EIF	Event Integration Facility
EKMS	Electronic Key Management System
ESM	Enterprise System Management

FCAPS	Fault management, Configuration management, Accounting management, Performance management, Security management
FIPS	Federal Information Processing Standard
FSO	Field Security Operations
GEM	Global Enterprise Manager
GID	Group ID
GOTS	Government-Off-the-Shelf
HIDS	Host-based Intrusion Detection System
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
I&A	Identification and Authentication
IA	Information Assurance
IAM	Information Assurance Manager
IAO	Information Assurance Officer
IASE	Information Assurance Support Environment
IATF	Information Assurance Technical Framework
IAVA	Information Assurance Vulnerability Alert
IAVM	Information Assurance Vulnerability Management
IBM	International Business Machines
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force
IIS	Internet Information Services
ILT	Instrumentation Library Type
IT	Information Technology
ITM	IBM Tivoli Monitoring
ITU	International Telecommunication Union
JTF-GNO	Joint Task Force - Global Network Operations
KMI	Key Management Infrastructure
LDAP	Lightweight Directory Access Protocol
MAC	Mission Assurance Category
MDist	Multiplexed Distribution
MFA	Management Functional Areas
MIF	Management Information Format
MMC	Microsoft Management Console
MOM	Microsoft Operations Manager

NIAP	National Information Assurance Partnership
NIPRNET	Non-classified Internet Protocol Router Network
NIST	National Institute of Standards and Technology
NSA	National Security Agency
NSTISSP	National Security Telecommunications and Information Systems Security Policy
OID	Object Identifier
OOB	Out of Band
ORB	Object Request Broker
OS	Operating System
OSI	Open Systems Interconnection
OSS	Open Source Software
PCIM	Policy Core Information Model
PDI	Potential Discrepancy Item
PKI	Public Key Infrastructure
PM	Program Manager
POC	Point of Contact
PPS	Ports, Protocols, and Services
PPSM	Ports, Protocols, and Services Management
RBAC	Role-Based Access Control
RDBMS	Relational Database Management System
RFC	Request for Comment
RIM	RDBMS Interface Module
RME	Resource Model Engine
S/MIME	Secure Multipurpose Internet Mail Extensions
SA	System Administrator
SDID	Short Description ID
SDK	Software Development Kit
SIPRNet	Secret Internet Protocol Router Network
SMB	Server Message Block
SMS	Systems Management Server
SNMP	Simple Network Management Protocol
SSL	Secure Sockets Layer
STIG	Security Technical Implementation Guide
SUS	Software Update Services
TBSM	Tivoli Business Systems Manager
TCP/IP	Transmission Control Protocol / Internet Protocol
TEC	Tivoli Enterprise Console
TMA	Tivoli Management Agent
TME	Tivoli Management Environment
TMF	Tivoli Management Framework

TMR	Tivoli Management Region
TNR	Tivoli Name Registry
UI	User Interface
UID	User ID
URL	Uniform Resource Locator
USM	User-based Security Model
VMS	Vulnerability Management System
VPN	Virtual Private Network
WBEM	Web-Based Enterprise Management
WINS	Windows Internet Naming Service
WMI	Windows Management Instrumentation
WUS	Windows Update Services
XML	Extensible Markup Language