

UNCLASSIFIED



VMware ESXi SERVER 5.0 SECURITY TECHNICAL IMPLEMENTATION GUIDE (STIG) OVERVIEW

Version 1, Release 10

27 January 2017

Developed by DISA for the DoD

UNCLASSIFIED

Trademark Information

Names, products, and services referenced within this document may be the trade names, trademarks, or service marks of their respective owners. References to commercial vendors and their products or services are provided strictly as a convenience to our users, and do not constitute or imply endorsement by DISA of any non-Federal entity, event, product, service, or enterprise.

TABLE OF CONTENTS

	Page
1. INTRODUCTION.....	1
1.1 Executive Summary	1
1.2 Authority	1
1.3 Vulnerability Severity Category Code Definitions	1
1.4 STIG Distribution.....	2
1.5 SRG Compliance Reporting.....	2
1.6 Document Revisions	2
1.7 Other Considerations.....	2
1.8 Product Approval Disclaimer.....	3
2. ASSESSMENT CONSIDERATIONS.....	4
2.1 Security Assessment Information	4
2.1.1 VMware ESXi Server 5.0 Security Technical Implementation Guide.....	4
2.1.2 VMware vCenter Server Security Technical Implementation Guide.....	4
2.1.3 VMware Virtual Machine Security Technical Implementation Guide.....	4
2.2 Command Examples	4
2.3 File Paths	4
2.4 Alternate Software	5
2.5 Requirements for Disabled Functions.....	5
3. CONCEPTS AND TERMINOLOGY CONVENTIONS.....	6
3.1 File Permissions	6
3.1.1 File Ownership.....	6
3.1.2 Access Modes	6
3.1.3 Links	8
3.2 Role-Based Access Control.....	9
3.3 Services and Daemons	9
3.4 The ESXi 5 Server and Virtual Machines	9
3.5 ESXi 5 Virtual Networking.....	11
3.6 The ESXi 5 Shell and SSH.....	13
3.7 The ESXi 5 Server Local Access and Lockdown Mode.....	13
4. SOFTWARE PATCHING GUIDELINES	15
5. OPEN SOURCE SOFTWARE (OSS) USE	16

LIST OF TABLES

	Page
Table 1-1: Vulnerability Severity Category Code Definitions	2
Table 3-1: Access Mode Examples.....	6
Table 3-2: Binary Representation of the Octal Mode Number	8

1. INTRODUCTION

1.1 Executive Summary

This VMware ESXi Server 5.0 Security Technical Implementation Guide (STIG) provides the technical security policies, requirements, and implementation details for applying security concepts to this UNIX-like hypervisor.

The VMware vSphere 5 Server STIG contains product-specific, best-practices requirements for VMware ESXi Server 5.0. This STIG describes the ESXi 5 built-in security features and the measures to safeguard ESXi 5 from attack. This STIG may be used to secure the vSphere 5 environment for VMware vCenter Server 5 and VMware ESXi Server 5.0. This guide, along with the general-purpose Operating System (OS) Security Requirements Guide (SRG) and the Operating System (OS) SRG (UNIX Version), was used as input into this STIG.

The security requirements contained within this STIG are designed to assist Security Managers (SMs), Information System Security Managers (ISSMs), Information System Security Officers (ISSOs), and System Administrators (SAs) with configuring and maintaining security controls in a VMware vSphere environment.

This document is not a guide to UNIX, UNIX-like, or ESXi 5 system administration. From this point on, any reference to UNIX will also include UNIX-like operating systems as well.

1.2 Authority

DoD Instruction (DoDI) 8500.01 requires that “all IT that receives, processes, stores, displays, or transmits DoD information will be [...] configured [...] consistent with applicable DoD cybersecurity policies, standards, and architectures” and tasks that Defense Information Systems Agency (DISA) “develops and maintains control correlation identifiers (CCIs), security requirements guides (SRGs), security technical implementation guides (STIGs), and mobile code risk categories and usage guides that implement and are consistent with DoD cybersecurity policies, standards, architectures, security controls, and validation procedures, with the support of the NSA/CSS, using input from stakeholders, and using automation whenever possible.” This document is provided under the authority of DoDI 8500.01.

Although the use of the principles and guidelines in these SRGs/STIGs provide an environment that contributes to the security requirements of DoD systems, applicable NIST SP 800-53 cybersecurity controls need to be applied to all systems and architectures based on the Committee on National Security Systems (CNSS) Instruction (CNSSI) 1253.

1.3 Vulnerability Severity Category Code Definitions

Severity Category Codes (referred to as CAT) are a measure of vulnerabilities used to assess a facility or system security posture. Each security policy specified in this document is assigned a Severity Category Code of CAT I, II, or III.

Table 1-1: Vulnerability Severity Category Code Definitions

	DISA Category Code Guidelines
CAT I	Any vulnerability, the exploitation of which will, directly and immediately result in loss of Confidentiality, Availability, or Integrity.
CAT II	Any vulnerability, the exploitation of which has a potential to result in loss of Confidentiality, Availability, or Integrity.
CAT III	Any vulnerability, the existence of which degrades measures to protect against loss of Confidentiality, Availability, or Integrity.

1.4 STIG Distribution

Parties within the DoD and Federal Government's computing environments can obtain the applicable STIG from the Information Assurance Support Environment (IASE) website. This site contains the latest copies of any STIGs, SRGs, and other related security information. The address for the IASE site is <http://iase.disa.mil/>.

1.5 SRG Compliance Reporting

All technical NIST SP 800-53 requirements were considered while developing this STIG. Requirements that are applicable and configurable will be included in the final STIG. A report marked For Official Use Only (FOUO) will be available for those items that did not meet requirements. This report will be available to component Authorizing Official (AO) personnel for risk assessment purposes by request via email to: disa.stig_spt@mail.mil.

1.6 Document Revisions

Comments or proposed revisions to this document should be sent via email to the following address: disa.stig_spt@mail.mil. DISA will coordinate all change requests with the relevant DoD organizations before inclusion in this document. Approved changes will be made in accordance with the DISA maintenance release schedule.

1.7 Other Considerations

DISA accepts no liability for the consequences of applying specific configuration settings made on the basis of the SRGs/STIGs. It must be noted that the configurations settings specified should be evaluated in a local, representative test environment before implementation in a production environment, especially within large user populations. The extensive variety of environments makes it impossible to test these configuration settings for all potential software configurations.

For some production environments, failure to test before implementation may lead to a loss of required functionality. Evaluating the risks and benefits to a system's particular circumstances and requirements is the system owner's responsibility. The evaluated risks resulting from not

applying specified configuration settings must be approved by the responsible Authorizing Official. Furthermore, DISA implies no warranty that the application of all specified configurations will make a system 100% secure.

Security guidance is provided for the Department of Defense. While other agencies and organizations are free to use it, care must be given to ensure that all applicable security guidance is applied both at the device hardening level as well as the architectural level due to the fact that some of the settings may not be able to be configured in environments outside the DoD architecture.

1.8 Product Approval Disclaimer

The existence of a STIG does not equate to DoD approval for the procurement or use of a product.

STIGs provide configurable operational security guidance for products being used by the DoD. STIGs, along with vendor confidential documentation, also provide a basis for assessing compliance with Cybersecurity controls/control enhancements which supports system Assessment and Authorization (A&A) under the DoD Risk Management Framework (RMF). DoD Authorizing Officials (AOs) may request available vendor confidential documentation for a product that has a STIG for product evaluation and RMF purposes from disa.stig_spt@mail.mil. This documentation is not published for general access to protect vendor's proprietary information.

AOs have the purview to determine product use/approval IAW DoD policy and through RMF risk acceptance. Inputs into acquisition or pre-acquisition product selection include such processes as:

- National Information Assurance Partnership (NIAP) evaluation for National Security Systems (NSS) (<http://www.niap-ccevs.org/>) IAW CNSSP #11
- National Institute of Standards and Technology (NIST) Cryptographic Module Validation Program (CMVP) (<http://csrc.nist.gov/groups/STM/cmvp/>) IAW Federal/DoD mandated standards
- DoD Unified Capabilities (UC) Approved Products List (APL) (<http://www.disa.mil/network-services/ucco>) IAW DoDI 8100.04

2. ASSESSMENT CONSIDERATIONS

There are three product-specific STIGs related to ESXi Server 5.0. All three must be utilized in securing a VMware vSphere 5 site installation. They refer to the ESXi 5 bare-metal hypervisor running virtual machines, the vCenter Server used to manage the hypervisor, and virtual machines configured to run on the hypervisor.

2.1 Security Assessment Information

2.1.1 VMware ESXi Server 5.0 Security Technical Implementation Guide

The VMware ESXi Server 5.0 Security Technical Implementation Guide may be used as a guide for enhancing the security configuration of the ESXi 5 Server system, including the server's virtual machines and virtual networking components. This Overview is for the ESXi Server 5.0 STIG.

2.1.2 VMware vCenter Server Security Technical Implementation Guide

The VMware vCenter Server Security Technical Implementation Guide may be used as a guide for enhancing the security configuration of the vCenter Server system, including the vSphere Update Manager.

2.1.3 VMware Virtual Machine Security Technical Implementation Guide

The VMware Virtual Machine Security Technical Implementation Guide may be used as a guide for enhancing the security configuration of the ESXi 5 Server's virtual machines.

2.2 Command Examples

Some check and fix procedures contain example commands that can be used to obtain information regarding compliance with a requirement or to change a setting to attain compliance with a requirement. These example commands assume use of a standard UNIX shell operating as the root user. The commands used in the ESXi Server 5.0 STIG are intended to only use the busybox utility, non-busybox ESXi 5 command line commands, vSphere Client GUI software, and required parameters. If the busybox, command line, or GUI software used by these commands is not present on the system, or does not recognize the specified options, the SA or the reviewer is responsible for determining compliance with the requirement using the tools available on the system. Check procedures also contain instructions for evaluating compliance based on the output of these commands.

2.3 File Paths

The check and fix procedures for the ESXi Server 5.0 STIG use absolute file paths where possible.

2.4 Alternate Software

vSphere 5 systems offer extreme flexibility in components provided by the vendor to meet operational needs. Many of the check and fix procedures in the ESXi Server 5.0 STIG assume the use of a specific command line or GUI utility provided by the vendor. If an alternate software method is used to provide a function ordinarily provided by the default specified command line or GUI utility, the specific check and fix information for that function is no longer valid. The SA or the reviewer is responsible for evaluating the requirements based on documentation available for the alternate method used. The system accreditation package must contain information pertaining to the use of the alternate check and fix methodology used.

2.5 Requirements for Disabled Functions

The ESXi Server 5.0 STIG defines requirements for the further hardening and configuration of system functions that are required to be disabled. These requirements exist to address vulnerabilities in the system resulting from accidental activation, malicious intentional activation, or intentional activation of the system function based on acceptance of risk by the Authorizing Official (AO). Requirements for a system function remain applicable even when the system function is disabled. Requirements pertaining to software that is not installed on the system, and which has no remaining configuration files on the system, may be evaluated as not applicable.

3. CONCEPTS AND TERMINOLOGY CONVENTIONS

The ESXi Server 5.0 STIG assumes familiarity with some common vSphere 5 concepts and terminology. Some of these concepts and terms are defined and explained in this document in order to facilitate uniform interpretation of the requirements.

3.1 File Permissions

Files are assigned access permissions with standard access modes or additionally with access control lists (ACLs). The sometimes cumbersome and restrictive nature of the standard file access modes is not always suitable for certain environments. ACLs provide a greater degree of file access control and a more granular level of file protection, allowing certain privileges to either specific users or specific groups of users. This granular level of file protection provides more flexibility for ensuring file and directory access restrictions.

3.1.1 File Ownership

Files have two ownership attributes representing the user and group owners. The user owner of a file retains access to a file regardless of access control settings. Generally, user owners are not permitted to change the user ownership of a file, but they can change the group owner of files they own.

3.1.2 Access Modes

An access mode represents the standard and special access modes associated with a file. Access modes can be represented as a 3- or 4-digit octal number, a symbolic string as produced by **ls -l**, or as a symbolic combination of classes and permissions as used by **chmod(1)**. The table below provides some examples of different representations of access modes.

Table 3-1: Access Mode Examples

Permissions	Octal Access Mode	ls -l symbolic	chmodsymbolic
User read, write, execute	700 or 0700	-rwx-----	u+rwx
User, group, and other have read, write, execute	777 or 0777	-rwxrwxrwx	a+rwx
SetUID; User full access; Group and other read and execute	4755	-rwsr-xr-x	u+rwx,go+rx

3.1.2.1 Standard Access Modes

The standard file protection model assigns three permissions (read, write, and execute) for each of three classes of users (user, group, and other). Each class may be granted access to a file

using any combination of the read, write, and execute permissions.

The read permission allows the ability to read a file or list the contents of a directory. The write permission allows the ability to modify a file or add or delete a directory entry. The execute permission allows the ability to execute a file or traverse a directory.

The “user” class represents only the assigned owner of the file. The “group” class represents any member of the group owning the file. The “other” class, also referred to as “world”, represents any user on the system not covered by the “user” or “group” classes.

3.1.2.2 Special Access Modes

The fourth (left-most) octal digit of an access mode is used to represent three settings used to alter how permissions are processed: setuid, setgid, and the sticky bit.

3.1.2.2.1 Set User ID

The “set user ID” attribute is also commonly known as “setuid” or “suid”. When a file with the setuid attribute is executed, it inherits the privileges of the owner of the file and not the user executing the file. If the owner of the file is root, then the user, while executing that file, has all the powers of root, at least for the scope of the program being executed. Setuid is ordinarily used for system utilities, such as **passwd(1)**, that allow users to modify files, such as /etc/shadow, for which they have no permissions. It is, therefore, extremely important that any file that has the setuid bit set is of known origin and scope.

Refer to the vendor’s ESXi 5 documentation for details concerning setuid programs. Commercial and Government-supplied applications may also contain programs with the setuid bit set. If so, the vendor/proponent instructions must be followed.

3.1.2.2.2 Set Group ID

The “set group ID” attribute is also known as “setgid” or “sgid”. Similar to the setuid attribute, when a file with the setgid attribute is executed, it inherits the privileges of the group owner of the file. It is, therefore, extremely important that any file that has the setgid bit set is of known origin and scope.

The set group ID has another function when applied to a directory. New files that are created in a directory with the setgid bit set are group-owned by the directory’s group, instead of the group of the process creating the file. This can simplify the use of shared directories. The setgid bit conveys no additional privileges when set on a directory.

Refer to the ESXi 5 documentation for details concerning setgid. Commercial and Government-supplied applications may also supply programs with the setgid bit set. If so, then vendor/proponent instructions must be followed.

3.1.2.2.3 Sticky Bit

Directories that are world-writable (that is, the “other” user class has write permission) can be accessed and changed by any user with access to the system. Users could populate these directories with erroneous, malicious, and harmful files. In addition, users could also delete files belonging to other users contained in these directories. In the event a directory is required to allow all users permission to write to this directory, such as in the case of public directories (e.g., /tmp), the sticky bit must be set.

The sticky bit protects the files within this directory by preventing a user from deleting other users’ files also located in this public directory. When the sticky bit has been set on a directory, the owner of the file, owner of the directory, or root, may only delete a file. For that reason, world-writable directories will only be allowed if they are public directories and have the sticky bit set.

3.1.2.3 Comparing Access Modes

The ESXi Server 5.0 STIG uses the terms “less permissive” and “more permissive” when comparing the access modes of files to a defined value. An access mode is considered “more permissive” than the defined value if it has any permissions set that the defined value does not. If the mode is not equal to the defined value and is not “more permissive”, it is considered “less permissive” than the defined value.

Access modes defined in the ESXi Server 5.0 STIG are expressed as 4-digit octal numbers. To determine whether a given mode number is more or less permissive than another, the binary representation of the modes can be compared.

An example is a check that states that “The /etc/example file must have mode 0600 or less permissive”. If the /etc/example file on the system has a mode of 0466, this is a finding, as a mode of 0466 is more permissive than 0600. Consider the binary representation of the octal mode numbers:

Table 3-2: Binary Representation of the Octal Mode Number

Defined Access Mode				Actual Access Mode			
0	6	0	0	0	4	6	6
000	110	000	000	000	100	110	110
	rw-	---	---		r--	rw-	rw-

The file’s mode has read and write permissions for the “group” and “other” classes, while the defined mode does not have these permissions.

3.1.3 Links

From a file system perspective, links refer to multiple paths that point to a single physical file. File systems generally support two kinds of links: hard links and symbolic links.

3.1.3.1 Hard Links

Hard links are directory entries pointing to the same physical file within a single file system. There is no distinction between the original file and a hard link pointing to the file. Ownership and modes are attributes of the physical files and therefore only need to be checked or changed using one path. If permissions on intermediate traversed directories are used to restrict access to a file, each path to the file must be examined to verify correct configuration.

3.1.3.2 Symbolic Links

Symbolic links, often referred to as symlinks, are special files that reference another path on the system. Unlike hard links, symlinks may link to paths on different file systems and have ownership attributes that are distinct from the referenced file. As with ordinary files, symlinks may be removed or deleted by any user with permission to write to the directory containing the file. The only permission given to the symlink owner is the ability to remove or rename a link when located in a directory with the sticky bit set. All other permissions are controlled by the mode of the target file. When evaluating the ownership or mode of a path that is a symlink, the attributes of the referenced path must be evaluated instead of the attributes of the symlink itself. This can be accomplished using the “-L” option of ls(1).

3.2 Role-Based Access Control

ESXi 5 provides the root user, and only the root user, with full control of all system administrative functions. ESXi 5 also allows these administrative functions to be delegated to non-root user administrative accounts through the use of role-based access control. This functionality can also be used to provide “separation of duty” requirements, where authorizations from multiple users are required to perform a specific action.

3.3 Services and Daemons

As a multi-tasking server operating system, ESXi 5 has a concept of background processes that provide services to clients, both local to the system and remotely through the network. The ESXi Server 5.0 STIG uses the terms “service” and “daemon” to refer to this concept. Services are commonly started automatically by the system-run control scripts and may have processes in place to provide for automatic restart in the event of a software failure. Some daemons provide the capability for reloading their configuration upon receiving a signal from another process, permitting configuration changes without interruption of service.

3.4 The ESXi 5 Server and Virtual Machines

To best understand the basic architecture of a virtualized system, the following definitions are provided:

Supervisor: A computer program (usually the kernel part of an operating system), that controls the execution of high-level routines and regulates scheduling, I/O operations, and system errors and regulates the flow of work in a data processing system. Historically, this term was

associated with IBM's Mainframe OS/360. In other non-mainframe operating systems (OS), the supervisor is referred to as the kernel.

Hypervisor: A virtual machine manager/host (VMM) that allows multiple operating systems (also referred to as guests) to run concurrently on a host computer. It is so named because it is conceptually an abstraction of a supervisory program. The hypervisor presents, to the guest operating systems, a virtual operating platform, or virtual machine (VM). While ESXi 5 is an OS in the strictest sense, it is of a highly specialized and singularly purposed nature, not unlike an “appliance” with some initial configuration required. ESXi 5 is therefore atypical of what users would expect of an OS, due to the nature of how little it will actually support.

Hypervisor Type 1: A Type 1 (or native, bare-metal) hypervisor runs directly on the host's hardware to control the hardware and to manage guest operating systems. The guest OS runs one level above the hypervisor. ESXi 5 is a Type 1 hypervisor.

Hypervisor Type 2: A Type 2 (or hosted) hypervisor runs within a conventional OS environment. With the hypervisor layer as a distinct second software level, guest operating systems run at the third level above the hardware. This does not apply to the ESXi 5 hypervisor.

In the ESXi 5 architecture, all of the VMware agents run directly on the ESXi 5 Server's VMkernel. Infrastructure services are provided natively through modules included in the VMkernel. Other authorized third-party modules, such as hardware drivers and hardware monitoring components, can run in the VMkernel as well. Only modules that have been digitally signed by VMware are allowed on the system, creating a tightly locked-down architecture. Preventing arbitrary code from running on the ESXi 5 Server greatly improves the security and stability of the system.

A virtual machine is a tightly isolated software container managed by the ESXi 5 Server and can run its own operating systems and applications as if it were a physical computer. A virtual machine behaves exactly like a physical computer and contains its own virtual (i.e., software-based) CPU, RAM hard disk, and network interface card (NIC).

An operating system cannot tell the difference between a virtual machine and a physical machine, nor can applications or other computers on a network. Even the virtual machine thinks it is a “real” computer. Nevertheless, a virtual machine is composed entirely of software and contains no hardware components whatsoever. As a result, virtual machines offer a number of distinct advantages over physical hardware.

The <virtual_machine>.vmx file contains all of the configuration information and hardware settings of the virtual machine. Whenever the settings of a virtual machine are modified, all of that information is stored in text format in this file. This file can contain a wide variety of information about the VM, including its specific hardware configuration (i.e., RAM size, network interface card info, hard drive info, and serial/parallel port info), advanced power and resource settings, VMware tools options, and power management options. While possible to manually modify the file to make changes to a VM's configuration, it is not recommended to be done while the server VM is running, under normal operating conditions. Precautions must be taken to ensure the vCenter Server is “aware” of all changes. The <virtual_machine>.vmx file

and contents (keywords and settings) are addressed in the virtual machine section of the ESXi 5 Virtual Machine STIG as the primary focus for virtual machine hardening.

All virtual disks are made up of two files, a large data file equal to the size of the virtual disk and a small text disk descriptor file, which describes the size and geometry of the virtual disk file. The descriptor file also contains a pointer to the large data file as well as information on the virtual disks drive sectors, heads, cylinders, and disk adapter type. In most cases, these files will have the same name as the data file that it is associated with (i.e., vm_1.vmdk and vm_1-flat.vmdk).

The three different types of virtual disk data files that can be used with virtual machines are covered below:

- The -flat.vmdk file
This is the default large virtual disk data file that is created when you add a virtual hard drive to your VM that is not an RDM. When using thick disks, this file will be approximately the same size as what you specify when you create your virtual hard drive. One of these files is created for each virtual hard drive that a VM has configured, as shown in the examples below.
- The -delta.vmdk file
These virtual disk data files are only used when snapshots are created of a virtual machine. When a snapshot is created, all writes to the original -flat.vmdk are halted and it becomes read-only; changes to the virtual disk are then written to these -delta files instead. The initial size of these files is 16 MB, and they are grown as needed in 16 MB increments as changes are made to the VM's virtual hard disk. Because these files are a bitmap of the changes made to a virtual disk, a single -delta.vmdk file cannot exceed the size of the original -flat.vmdk file. A delta file will be created for each snapshot created for a VM, and their file names will be incremented numerically (i.e., vm-000001-delta.vmdk, vm-000002-delta.vmdk). These files are automatically deleted when the snapshot is deleted after they are merged back into the original -flat.vmdk file.
- The -rdm.vmdk file
This is the mapping file for the RDM that manages mapping data for the RDM device. The mapping file is presented to the ESXi Server as an ordinary disk file, available for the usual file system operations. However, to the virtual machine, the storage virtualization layer presents the mapped device as a virtual SCSI device. The metadata in the mapping file includes the location of the mapped device (name resolution) and the locking state of the mapped device. In a directory listing, these files will appear to take up the same amount of disk space on the VMFS volume as the actual size of the LUN that it is mapped to, but, in reality, the files just appear that way while the actual size is very small. One of these files is created for each RDM that is created on a VM.

3.5 ESXi 5 Virtual Networking

vSphere virtual networking provides services to the vSphere ESXi Server and virtual machines. There are four types of network services enabled in vSphere networking:

- Connecting virtual machines to each other within a single vSphere host
- Connecting virtual machines to the physical network
- Connecting VMkernel services (such as NFS, iSCSI, or vSphere vMotion) to the physical network
- Networking for the management interface, which runs management services for vSphere hosts (set up by default during installation)

vSphere networking consists of two logical building blocks: Virtual Ethernet Adapters and Virtual Switches.

A virtual machine can be configured with one or more virtual Ethernet adapters. Virtual Ethernet adapters are presented to the guest OS by the virtual machine hardware. These virtual adapters are seen by the guest OS as common network interface cards and will use standard drivers available for the OS.

Virtual switches allow virtual machines on the same vSphere ESXi Server to communicate with each other using the same protocols used with physical switches. The virtual switch emulates a traditional physical Ethernet network switch to the extent that it forwards frames at the data link layer.

The design of a virtual network with vCenter Server and ESXi 5 is very similar to building a physical network. Some of the factors that affect the design of a virtual network:

- vSphere Standard Switch – A VMkernel software-based switch that provides virtual machine traffic management. Standard switches must be managed independently on a host-by-host basis.
- vSphere Distributed Switch - A VMkernel software-based switch that provides traffic management for virtual machines and the VMkernel. Distributed switches are shared by, and managed across, ESXi Server clusters.
- Ports and Port Groups – A logical object on a vSwitch that provides general-to-specialized services for the VMkernel and/or virtual machines. A virtual switch will contain a VMkernel port or a virtual machine port group but must never contain both. An exception, in extreme circumstances where hardware must be shared due to limited resources, must ensure isolation of production virtual machine traffic from management traffic through the use of VLANs for production virtual machine traffic.
- VMkernel Port – This is a specialized virtual switch port type that is configured with an IP address defined to implement functions such as vMotion, network attached storage (NAS), Network File Systems (NFS), and ESXi Server management connectivity.
- Virtual Machine Port Group – This is a group of virtual switch ports that share a common configuration, allowing virtual machines to access other virtual machines and/or the physical network.
- Uplink ports - Uplink ports provide the logical connectivity between the virtual switch and the physical adapters installed in the host. Uplink ports are connected to specific physical adapters based on configuration.
- Uplinks - Uplinks are physical Ethernet adapters that serve as bridges between the virtual and physical network. Currently, a host may have up to 32 uplinks.

3.6 The ESXi 5 Shell and SSH

The ESXi 5 Shell is a simple shell intended for advanced troubleshooting under the guidance of technical support. When remote command-line tools are not capable of addressing a particular issue, the ESXi 5 Shell provides an alternative. Similar to how the COS is used to execute diagnostic commands and fix certain low-level problems, the ESXi 5 Shell enables users to view log and configuration files, as well as to run certain configuration and utility commands to diagnose and fix problems. The ESXi 5 Shell is not based on Linux. Rather, it is a limited-capability shell compiled especially for ESXi 5.

In addition to being available on the local console of a host, the ESXi 5 Shell can be accessed remotely through SSH. The ESXi 5 SSH implementation is not a full-featured version. Access to the ESXi 5 Shell is controlled in the following ways:

- Both SSH and the ESXi 5 Shell can be enabled and disabled separately in both the DCUI and the vSphere Client or through vSphere PowerCLI.
- Any authorized user, not just root, can use the ESXi 5 Shell. Users become authorized when they are granted the administrator role on a host (through Active Directory membership in a privileged group and through other methods).
- All commands issued in the ESXi 5 Shell are logged through syslog, providing a full audit trail. If a syslog server is configured, this audit trail is automatically included in the remote logging.
- A timeout for the ESXi 5 Shell (including SSH), will automatically disable the interface(s) after the configured time. Changes to the SSH timeout will apply only to new sessions. Existing sessions will not be timed out, but any new session is disallowed after the timeout period.

The ESXi 5 Shell is vendor-recommended for use primarily for vendor support, initial configuration of complex configuration files, troubleshooting, and break-fix situations. It also can be used as part of a scripted installation, as described in a previous section. All other uses of the ESXi 5 Shell, including running custom scripts (i.e., development), are not recommended in most cases. Instead, the vSphere vCLI or vSphere PowerCLI must be used.

Note: Both the ESXi 5 Shell and SSH are disabled by default.

3.7 The ESXi 5 Server Local Access and Lockdown Mode

ESXi 5 provides the ability to fully control all direct access to the host via vCenter Server 5. After a host has been joined to vCenter Server 5, every direct communication interface with the host is configurable as an independent service in the Configuration tab for the host in the vSphere Client. This includes the following interfaces:

- DCUI (Direct Console User Interface)
- ESXi 5 Shell

- SSH

Each of these can be turned on and off individually through either the vSphere Client/vCenter interface or the DCUI.

When Lockdown Mode is enabled on the host, all direct remote access to the host is blocked, including:

- Any vSphere API client
- ESXi 5 Shell
- SSH

Even when Tech Support Mode is enabled, Lockdown Mode effectively overrides this by preventing any connection from succeeding. The only way to manage the host remotely is through vCenter Server. The interaction between the host and vCenter Server occurs through a special-purpose account called “vpxuser”; all other accounts, including root, can no longer connect remotely.

With Lockdown Mode enabled, the only direct access to the host that remains open is through the DCUI. The DCUI allows you to interact with the host locally using text-based menus. You can use the Direct Console User Interface to enable local and remote access to the ESXi 5 Shell. This provides a way to perform limited administrative tasks outside of vCenter Server. The DCUI can also turn off Lockdown Mode, disabling it without going through vCenter Server. This might be useful if vCenter Server is down or otherwise unavailable and users want to revert to direct management of the host. To log in to the DCUI in Lockdown Mode, however, the root password is required. No other user can log in, even if they have been granted an administrator role.

The requirement is that Lockdown Mode, including the DCUI, be enforced in ordinary, day-to-day operations but that it be temporarily disabled, as required, for a host, if the need arises to interact with it directly.

4. SOFTWARE PATCHING GUIDELINES

Maintaining the security of an ESXi 5 system requires frequent reviews of security bulletins. Many security bulletins and IAVM notifications mandate the installation of software patches to overcome noted security vulnerabilities. The SA will be responsible for ensuring the installation of all such patches. The ISSO will ensure the vulnerabilities have been remedied. DISA guidelines for remediation, including IAVMs, are as follows:

- Apply the applicable patch, upgrade to required software release, or remove the binary/application to remediate the finding.
- Or, the mode of the vulnerable binary may be changed to 0000 to downgrade the finding (for example, a CAT I finding may be downgraded to a CAT II).

SAs and ISSOs will regularly check the operating system's vendor and third-party application vendor websites for information on new vendor-recommended updates and security patches that are applicable to their site. All applicable vendor-recommended updates and security patches will be applied to the system. A patch is deemed applicable if the product is installed, even if it is not used or is disabled.

Operating system, virtual machine, and virtual network patching is accomplished through the use of three VMware products:

- vCenter Server
- VMware's Update Manager
- VMware's Update Manager Download Service/Server

The vCenter Server (vCS) is the central management system for ESXi 5. The VMware Update Manager (vUM) is responsible for updating the ESXi 5 server and VMs. The VMware vSphere Update Manager Download Service (vUMDS) is an optional module of the Update Manager. The vUMDS downloads software upgrades for virtual appliances, patch metadata, patch binaries, and notifications that would not otherwise be generally available to the Update Manager server. Virtual machine guest operating systems cannot be updated by the vCS, vUM, or vUMDS. A separate product, VMware Protect, centralizes patch management and asset inventory for Windows and third-party applications for both virtual and physical machines. The details of VMware Protect are not discussed within the scope of the ESXi Server 5.0 STIG.

5. OPEN SOURCE SOFTWARE (OSS) USE

On October 16th 2009, DoD CIO provided clarifying guidance regarding Open Source Software (OSS), reminding the DoD to take advantage of the capabilities available in the Open Source community as long as certain prerequisites are met. The 2009 memo can be found online at: <http://dodcio.defense.gov/Portals/0/Documents/FOSS/2009OSS.pdf>

It reads:

“Software for which the human-readable source code is available for use, study, reuse, modification, enhancement, and redistribution by the users of that software. In other words, OSS is software for which the source is “open.””

Additionally, per Section 2(d) of the 16 Oct 2009 DoD CIO memo:

“the use of *any* software without appropriate maintenance and support presents an information assurance risk. Before approving the use of software (including OSS), system/program managers, and ultimately Designated Approval Authorities (DAAs), must ensure that the plan for software support (e.g. commercial or Government program office support) is adequate for mission need.”