

UNCLASSIFIED



JBOSS ENTERPRISE APPLICATION PLATFORM (EAP) 6.3 SECURITY TECHNICAL IMPLEMENTATION GUIDE (STIG) OVERVIEW

Version 1, Release 2

28 April 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	2
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
3. CONCEPTS AND TERMINOLOGY CONVENTIONS.....	5
3.1 About JBoss Enterprise Application Platform	5
3.2 JBoss Operating Modes.....	5
3.3 JBoss Operating System Environment Variables	7
3.4 JBoss User Accounts	7
3.5 JBoss Management Capability	8
3.6 Role Based Access	9

LIST OF TABLES

	Page
Table 1-1: Vulnerability Severity Category Code Definitions	2
Table 3-1: JBoss Management Access Ports	8
Table 3-2: JBoss Startup Commands	9

LIST OF FIGURES

	Page
Figure 3-1: JBoss EAP Domain Architecture.....	7

1. INTRODUCTION

1.1 Executive Summary

The JBoss Enterprise Application Platform 6.3 (EAP) Security Technical Implementation Guide (STIG) is published as a tool to improve the security of Department of Defense (DoD) information systems. This document is meant for use in conjunction with the Enclave, Network Infrastructure, Web Server, Application Security and Development, and appropriate operating system (OS) STIGs.

This document is intended to be applied to Red Hat JBoss EAP version 6.3 for both Windows and UNIX platforms. Scope for the guidance is limited to the management of the JBoss EAP server and does not extend to applications that are hosted on the server or underlying OS or web server components. There have been numerous reported security vulnerabilities related to the JBoss EAP product. The security issues have been largely attributed to misconfiguration of the product and a lack of timely patching of the product. In the 6.3 version, the management interfaces used to manage the EAP server are secured by default. This prevents remote access to the system until access is specifically configured, which helps to assure a more secure initial deployment. While integrated patch management capability is offered in the form of email notifications and manual application of downloaded patches, an automated patch management solution is not included with EAP. To ensure the EAP servers are kept up to date in regard to patching, patch management tasks must be performed directly by system administrators via the use of underlying OS tools and/or third-party configuration management solutions. Automatically applying patches to any application server without prior testing increases the risk of adversely impacting the hosted applications.

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 configuration 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 percent 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 the 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

2.1 Security Assessment Information

To accurately assess the JBoss server, one must understand the mode in which the server is functioning. JBoss can be run in either a standalone or a domain mode of operation. The operating mode is specified at startup and can be determined while the server is operating by using the provided CLI.

The CLI is a command line based interactive shell that provides the capability to view and modify JBoss configuration settings. Syntax for the CLI will vary based upon the mode the server is running in. If the server is operating in domain mode, many commands will require a prefix that identifies the host and/or the profile the command is intended to be applied to. The STIG will specify checks and fixes according to the operating mode in which the server is running, which is either standalone or domain mode.

A CLI configuration file, `jboss-cli.xml` is loaded each time the CLI is run. It must be located either in the directory `$JBOSS_HOME/bin` or in a directory specified in the system property `"jboss.cli.config"`. The config file can be set up to connect to a specific default controller in the event the controller is not specified when the `"connect"` command is executed. The default controller is `"localhost"`. To obtain a list of available commands when using the CLI, use the `"help"` command.

To identify the role and permissions associated with the user executing the CLI, type the command `"whoami"`.

For additional details on the commands and operations available in the CLI, please reference "Section 3.5; The Management CLI" in the document titled, "RedHat-JBoss-Enterprise-Application-Platform-6.3-Administration_and_Configuration_Guide-en-US.pdf" located at RedHat.com. This document was a primary reference during STIG development. For full product documentation, please visit:

https://access.redhat.com/documentation/en-US/JBoss_Enterprise_Application_Platform/6.3/

3. CONCEPTS AND TERMINOLOGY CONVENTIONS

3.1 About JBoss Enterprise Application Platform

Red Hat JBoss Enterprise Application Platform 6 (JBoss EAP 6) is a middleware platform built on open standards and compliant with the Java Enterprise Edition 6 specification. It integrates JBoss Application Server 7 with high-availability clustering, messaging, distributed caching, and other technologies. The Management Console and Management Command Line Interface make editing XML configuration files unnecessary and add the ability to script and automate tasks. In addition, JBoss EAP 6 includes APIs and development frameworks for quickly developing secure and scalable Java EE applications.

3.2 JBoss Operating Modes

When securing JBoss EAP 6, it is important to understand the operating modes in which JBoss EAP 6 instances can be run. They are either “Standalone Server” or “Managed Domain”. The two modes differ in how servers are managed, not in their capacity to service end-user requests. It is important to note that the high-availability (HA) cluster functionality is available via either operating mode. A group of standalone servers can be configured to form an HA cluster. Due to the nature of the product and the different modes in which it can operate, the check and fix commands provided in the STIG are specified based upon the operating mode of the JBoss server. When assessing the product, you must first determine which mode the server is operating in.

Standalone Server operating mode is an independent process and is analogous to the only running mode available in previous JBoss EAP versions. A JBoss EAP 6 instance running as a standalone server is a single instance only but can optionally run in a clustered configuration.

Managed Domain operating mode allows for management of multiple JBoss EAP 6 instances from a single control point. Centrally managed JBoss EAP 6 server collections are known as members of a domain. All JBoss EAP 6 instances in a domain share a common management policy. A domain consists of one domain controller, one or more host controller(s), and zero or more server groups per host. A domain controller is the central point from which the domain is controlled. It ensures that each server is configured according to the management policy of the domain. The domain controller is also a host controller. A host controller is a physical or virtual host on which the domain.sh or domain.bat script is run. Host controllers are configured to delegate domain management tasks to the domain controller.

A domain controller is the JBoss EAP 6 server instance that acts as a central management point for a domain. One host controller instance is configured to act as a domain controller.

The primary responsibilities of the domain controller are:

- Maintain the domain’s central management policy
- Ensure all host controllers are aware of its current contents
- Assist the host controllers in ensuring that all running JBoss EAP 6 instances are configured in accordance with this policy

By default, the central management policy is stored in the domain/configuration/domain.xml file. This file is in the JBoss EAP 6 installation folder on the domain controller's host's file system. A domain.xml file must be located in the domain/configuration/directory of the host controller set to run as the domain controller. This file is not mandatory for installations on host controllers that are not meant to run as a domain controller. However, the presence of a domain.xml file on such a server does no harm.

The domain.xml file contains the profile configurations that can be run on the server instances in a domain. A profile configuration includes the detailed settings of the various subsystems that comprise a profile. The domain configuration also includes the definition of socket groups and the server group definitions.

A host controller is launched when JBoss is started using the domain.sh or domain.bat script on a JBoss host.

The primary responsibility of a host controller is server management. It delegates domain management tasks and is responsible for starting and stopping the individual application server processes that run on its host. It interacts with the domain controller to help manage the communication between the servers and the domain controller.

Multiple host controllers of a domain can interact with only a single domain controller. Hence, all the host controllers and server instances running on a single domain mode have a single domain controller and must belong to the same domain.

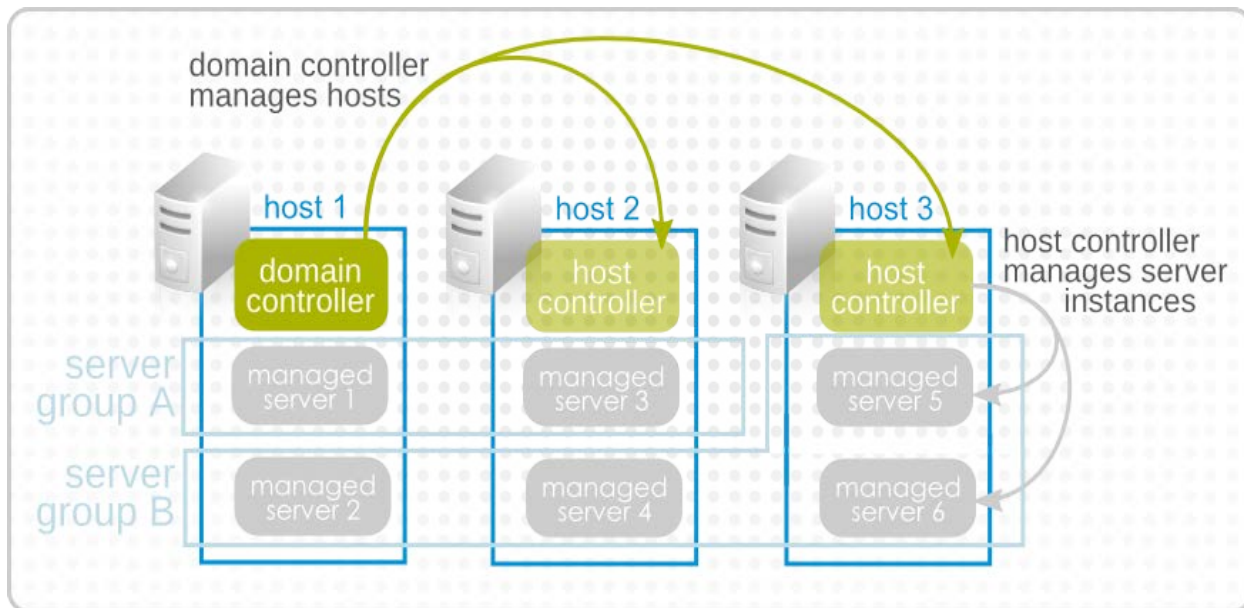
By default, each host controller reads its configuration from the domain/configuration/host.xml file located in the JBoss EAP 6 installation folder on the host's file system. The host.xml file contains the following configuration information that is specific to the particular host:

- The names of the JBoss EAP 6 instances meant to run from this installation.
- Any of the following configurations:
 - o How the host controller contacts the domain controller to register itself and access the domain configuration
 - o How to find and contact a remote domain controller
 - o The host controller is to act as the domain controller
- Configurations specific to the local physical installation. For example, named interface definitions declared in domain.xml can be mapped to an actual machine-specific IP address in host.xml, and abstract path names in domain.xml can be mapped to actual file system paths in host.xml.

JBoss EAP 6 Profiles: The concept of profiles that was used in previous versions of JBoss EAP is no longer used. JBoss EAP 6 now uses a small number of configuration files to hold all information about its configuration. Modules and drivers are now loaded on an as-needed basis. Consequently the concept of a default profile - used in previous versions of JBoss EAP 6 to make the server start more efficiently - does not apply.

At deployment time, module dependencies are determined, ordered, resolved by the server or domain controller, and loaded in the correct order. Modules are unloaded when no deployment needs them any longer. It is possible to disable modules or unload drivers and other services manually by removing the subsystems from the configuration. However, for most cases this is unnecessary. If none of your applications use a module, it will not be loaded.

Figure 3-1: JBoss EAP Domain Architecture



3.3 JBoss Operating System Environment Variables

JBoss must have OS environment variables and path statements set up to identify the location of the underlying Java Development Kit and the corresponding Java Runtime Environment as well as the location of the JBoss installation folder. These variables are as follows:

- **JAVA_HOME** - OS environment variable that identifies the folder location of the JRE on the underlying OS.
- **JBOSS_HOME** - OS environment variable that identifies the folder location of the JBoss home folder.

This information can also be used to identify the location of the running JBoss instance and its associated files on the file system.

3.4 JBoss User Accounts

User accounts for JBoss can be broken down into two basic categories.

- **An operating system user;** JBoss runs under the context of the OS user who launches the JBoss domain or standalone startup scripts provided with the product. By default, JBoss does not require special OS admin privileges in order to run, however, special consideration must be taken to ensure the TCP/IP ports used by JBoss are allowed

through the host firewall if one is installed on the host server. A JBoss user account should be created with restricted OS access and read write permissions to the JBOSS_HOME folder and its underlying folders.

- **JBoss users;** when JBoss is installed, a user will be added to the host container's management realm for administrative purposes. This admin account can be used to access the management console, the management CLI, or other applications that are secured by the management realm. By default, a password is assigned to this user account when the account is created. Once the JBoss server is installed, JBoss users can be added to or deleted from the management realm or other security realms via the management console or the management CLI. This capability is configured via the STIG.

3.5 JBoss Management Capability

JBoss EAP 6 natively offers three different approaches to configure and manage servers: a web interface in the form of a Management Console, a command line client in the form of a Management Command Line Interface (CLI), and a set of XML configuration files. The STIG utilizes the CLI commands whenever possible because the web console does have some limitations regarding configuration and assessment capabilities.

In JBoss EAP 6.3, all server instances and configurations are managed through management interfaces rather than by editing XML files. While the configuration XML files are still available for editing, administration through the management interfaces provides extra validation and advanced features for the persistent management of server instances. Changes made to the XML configuration files while the server instance is running will be overwritten, and any XML comments added will be removed as well. Only the management interfaces should be used for modifying the configuration files while a server instance is running.

To manage servers through a graphical user-interface in a web browser, use the Management Console. Default HTTP ports are provided for informational purposes only. All management access must utilize encryption.

Table 3-1: JBoss Management Access Ports

Management Access Ports	
URL	Description
http://localhost:9990/console https://localhost:8443/console	The Management Console when accessed on the local host, controlling the Managed Domain configuration.
http://hostname:9990/console https://hostname:8443/console	The Management Console when accessed remotely, naming the host and controlling the Managed Domain configuration.
http://hostname:9990/management https://hostname:8443/management	The HTTP Management API runs on the same port as the Management Console, displaying the raw attributes and values exposed to the API.
hostname:9999	The Management CLI connection TCP/IP port. Connection command is invoked from within the CLI.

To manage servers through the command line interface, use the Management CLI. The following table provides the command syntax required to start the CLI based on the host OS platform.

Table 3-2: JBoss Startup Commands

Operating System	Command to start CLI
Linux systems	<code>\$JBASS_HOME/bin/jboss-cli.sh -options</code>
Windows systems	<code>%JBASS_HOME%\bin\jboss-cli.bat -options</code>

Note: The STIG utilizes the CLI commands whenever possible since some commands are not available via the web-based Management Console.

3.6 Role Based Access

Note: RBAC is disabled by default in JBoss EAP 6.3. Role Based Access Controls must be enabled to secure the system and restrict access among management users.

JBoss EAP 6 provides seven predefined user roles-Monitor, Operator, Maintainer, Deployer, Auditor, Administrator, and SuperUser. Each of these roles has a different set of permissions and is designed for specific use cases. The Monitor, Operator, Maintainer, Administrator, and SuperUser roles each build upon one another, with each role having more permissions than the previous one. The Auditor and Deployer roles are similar to the Monitor and Maintainer roles, respectively, but have some additional special permissions and restrictions.

Monitor:

Users of the Monitor role have the fewest permissions and can only read the current configuration and state of the server. This role is intended for users who need to track and report on the performance of the server. Monitors cannot modify server configuration, nor can they access sensitive data or operations.

Operator:

The Operator role extends the Monitor role by adding the ability to modify the runtime state of the server. This means that Operators can reload and shut down the server as well as pause and resume Java Message Service (JMS) destinations. The Operator role is ideal for users who are responsible for the physical or virtual hosts of the application server so they can ensure that servers can be shut down and restarted corrected when needed. Operators cannot modify server configuration or access sensitive data or operations.

Maintainer:

The Maintainer role has access to view and modify runtime state and all configurations except sensitive data and operations. The Maintainer role is the general purpose role that does not have access to sensitive data and operation. The Maintainer role allows users to be granted almost complete access to administer the server without giving those users access to passwords and other sensitive information. Maintainers cannot access sensitive data or operations.

Administrator:

The Administrator role has unrestricted access to all resources and operations on the server except the audit logging system. The Administrator role has access to sensitive data and operations. This role can also configure the access control system. The Administrator role is only required when handling sensitive data or configuring users and roles. Administrators cannot access the audit logging system and cannot change themselves to the Auditor or SuperUser role.

SuperUser:

The SuperUser role has no restrictions and has complete access to all resources and operations of the server, including the audit logging system. This role is equivalent to the administrator users of earlier versions of JBoss EAP 6 (6.0 and 6.1). If RBAC is disabled, all management users have permissions equivalent to the SuperUser role.

Deployer:

The Deployer role has the same permissions as the Monitor but can modify configuration and state for deployments and any other resource type enabled as an application resource.

Auditor:

The Auditor role has all the permissions of the Monitor role and can also view (but not modify) sensitive data and has full access to the audit logging system. The Auditor role is the only role other than SuperUser that can access the audit logging system. Auditors cannot modify sensitive data or resources. Only read access is permitted.