WEB SERVER
SECURITY REQUIREMENTS GUIDE (SRG)
TECHNOLOGY OVERVIEW

Version 2, Release 3

26 April 2019

Developed by DISA for the DoD

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

Web Server SRG Overview, V2R3
26 April 2019

DISA
Developed by DISA for the DoD

iii

UNCLASSIFIED

LIST OF TABLES

1. INTRODUCTION

1.1 Executive Summary

This Web Server Security Requirements Guide (SRG) provides the technical security policies and requirements for applying security concepts to servers used to deliver web content to a client. Delivery of web content to a client includes session control, encryption of data during transmission, cookies, and communication protocols. The communication methods discussed are standardized communications in the HTTP 1.1 protocol, the latest HTTP version in release. The requirements do not prevent or mitigate all attacks against a poorly designed application which uses a web server. Please refer to the Application Security and Development STIG for application requirements.

1.1.1 Security Requirements Guides (SRGs)

SRGs are collections of requirements applicable to a given technology family. SRGs represent an intermediate step between Control Correlation Identifiers (CCIs) and Security Technical Implementation Guides (STIGs). CCIs represent discrete, measurable, and actionable items sourced from Information Assurance (IA) controls defined in a policy, such as the National Institute of Standards and Technology (NIST) Special Publication (SP) 800-53. STIGs provide product-specific information for validating and attaining compliance with requirements defined in the SRG for that product's technology area.

There are four core SRGs: Application, Network, Operating System, and Policy. Each addresses the applicable CCIs in the context of the technology family. Subordinate to the core SRGs, there are Technology SRGs developed to address the technologies at a more granular level.

This Web Server SRG is based on the Application SRG. This Web Server SRG contains general check and fix information that can be utilized for products for which STIGs do not exist.

The STIGs based on this SRG will provide the product-specific technical implementation guidance for that product. The STIG will contain the specific check and fix information for the product it covers.

SRG Hierarchy example:

> *Application SRG*
> *|__Database SRG*
> > *|__MS SQL Server 2005 STIG*

The SRG relationship and structure provides the ability to identify requirements that may be considered not applicable for a given technology family and provide appropriate justification. It also provides the structure to identify variations in specific values based on the technology family. These variations will be captured once and will propagate down to the Technology SRGs and then to the STIGs. This will eliminate the need for each product-specific STIG to address items that are not applicable.

1.1.2   SRG Naming Standards

In an effort to establish consistency across the SRGs, a naming standard for the Group Title and STIGIDs has been established.

Technology SRG Naming Standards

For Technology SRG Group Title and STIGIDs the following applies:

{Core SRG value}-*{Technology SRG}*-{5- or *6-digit numeric sequence number*}

Examples:

> *SRG-NET-000001-RTR-000001*
> *SRG-APP-000001-COL-000001*
> *SRG-NET-000001-VVSM-00001*
> *SRG-OS-000001-UNIX-000001*

Checks/Fixes will be included at this level in a general form. These checks and fixes will apply for any STIGs that are created for products that do not have product-specific check and fix guidance.

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.

1.2.1   Relationship to STIGs

The SRG defines the requirements for various technology families, and the STIGs are the technical implementation guidelines for specific products. A single SRG/STIG is not all-inclusive for a given system, which may include, but is not limited to: Database, Web Server,

and Domain Name System (DNS) SRGs/STIGs. For a given system, compliance with all (multiple) SRGs/STIGs applicable to a system is required.

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   SRG and 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   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.

1.6   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.

2. ASSESSMENT CONSIDERATIONS

2.1 NIST SP 800-53 Requirements

All applicable baseline technical NIST SP 800-53 requirements and security best practice requirements are included in this SRG.

CNSSI 1253 defines the required controls for DoD systems, based on confidentiality, integrity, and availability (baseline) of the given information system. In all cases, CNSSI 1253, along with required baselines, will serve as the policy requirement for any given asset or information system.

2.2 General Procedures

This SRG has procedures that are intended to provide appropriate evaluation and remediation functions for a typically configured system. These procedures are not product specific and are intended for use when a product-specific STIG is not available.

2.3 Security Assessment Information

Web servers are no longer just found on the Internet serving static applications to users, but they can now be found as part of many devices delivering administrative applications for device setup, frontends to operating systems, and parts of larger application server designs. When there is no vendor-related STIG for a web server, or when the web server is embedded and not easily recognizable, the SRG becomes a document that can be used to evaluate and remediate vulnerabilities for a generic but secure installation.

The SRG contains specific wording in each vulnerability discussion, check, and fix to signify the scope of the requirement as it applies to the Web Server SRG and STIG.

3. CONCEPTS AND TERMINOLOGY CONVENTIONS

This overview is not intended to be a comprehensive source of information on web server security but is meant to provide an understanding of the background, concepts, terminology, and boundaries used in the Web Server SRG. With an understanding of the underlying terms and concepts, the SRG can be better utilized to attain a secure product.

3.1 Web Server

Defining what a web server is and is not becomes essential to understand the boundaries of the Web Server SRG. A web server, in its simplest form, is software used to deliver content on the request of a client. Document DoDI 8520.03 defines a web server as "an automated information system that manages a website by passing web pages to web browsers over a network. The web server may provide information stored locally on the server or may act as a portal to access information from other linked information systems."

All web servers provide the capability of content delivery, but during web server evolution, plug-ins or modules have been added to extend the reach of a web server to now allow access to database systems, user management software, and audit systems. Communication to the backend services are addressed, but the functionality and security risks of each backend are not undertaken in the Web Server SRG.

A web server function that is not directly related to content delivery is hosted application user verification and user management capabilities. The user management and verification capabilities provided by a web server are usually limited, forcing application developers to use an external user management system designed for the sole purpose of managing user accounts. Since the user management capability is a feature that can be used, the web server, when also acting as a hosted application user management system, must meet the requirements of any user management software for user authentication, credential definition, storage, and management.

3.2 Hosted Web Server Applications

Hosted web server applications are applications that are accessed through a network and are provided to the client by the web server. Applications are typically classified as static or dynamic applications.

Dynamic applications contain web pages that are generated in real time. These applications are written in languages such as Java, JavaScript, ASP, or PHP. Each language is chosen for the strengths it offers in developing the application, but with each language, a different set of security concerns are introduced. The security risks of each language are not discussed in the Web Server SRG but should be considered for the overall security of the web server.

Static applications are applications that do not change, and they display the same information for every user. These applications are typically written in Hypertext Markup Language (HTML). Since there are no user inputs or external back ends, this type of application has less risk than a

dynamic application would have but must still be secured for the overall security of the web server.

In both cases, the Web Server SRG does not address hosted application security features beyond addressing permissions and locations of hosted application files, transmission of data, and configuration of the application within the web server. The hosted application should meet the requirements laid out in the Application SRG or a more specific vendor STIG.

3.3   Operating System User Accounts

The Web Server SRG addresses two types of operating system user accounts, privileged and non-privileged. Document DoD 8580.02-R defines a user as "DL1.52 - A person or entity with authorized access." From this definition of a user, processes can also be defined as users since they perform tasks on a user's behalf. As with any user account, the permissions given to each process must meet the role the process is performing.

A privileged user is an authorized user who has access to system control, monitoring, or administration functions. Privileged users for web server administration may also be privileged users for the operating system, but it is not a requirement. The roles of web server administration and operating system administration can be separated and completed by different users with specialized skills in each technology.

Non-privileged user accounts are accounts that are used to gain access to the hosting operating system. These accounts may have roles that make the account privileged with respect to the operating system or other processes operating on the hosting system, but they do not have privileges to administer the web server.

3.4   Virtual Hosts

Virtual hosting is the mechanism used to host multiple domains on a single web server. Since resources will be shared on the hosting hardware, operating system, and web server, care should be taken to host applications that provide data with the same security posture.

There are several methods that can be used to accomplish the task of virtual hosting:

- Name-based virtual hosting – The assignment of multiple host names to the same IP address.
- IP-based virtual hosting – The assignment of a unique IP address to each hosted application.
- Port-based virtual hosting – The assignment of a different port for each hosted application.

The Web Server SRG does not address each of these methods with security settings and best practices, but it defines the most secure method when hosting multiple domains is necessary.

3.5   Session Management

To understand session management, the definition of a session must be understood. A session between a client and the web server is typically a Hypertext Transfer Protocol (HTTP) session. HTTP is called a stateless protocol because each command or action is executed independently of those that came before it. Without session management and session identifiers, a user would need to re-authenticate with each new action. Through implementation of session identifiers, the web server can recall the state and credentials of a user, allowing the user to move seamlessly through the application.

Session management is the process of keeping track of session identifiers and user information. Session management can be handled by the client or by the web server. When performed by the client, the client must send to the web server, with each request, the session id and user credentials. Sending the user credentials with each transaction increases the possibility of the credentials being discovered and used by an unauthorized user.

When session management is performed by the web server, the user credentials are stored by the web server and the client is only required to send the session identifier with each transaction. The credentials are stored and protected through encryption of the data and security settings implemented by the operating system to protect the database where the information is stored.

4. GENERAL SECURITY REQUIREMENTS

Web server security goes beyond settings made to the configuration of the web server. To secure a web server properly, thought needs to be given to the applications being hosted, who the user community is, and where the web server will reside on the network. By not looking beyond the web server itself, security flaws in the implementation can lead to the compromise of user personally identifiable information (PII) and organization sensitive data and processes and to the compromise of access to other systems and applications within the organization with a trusted relationship to the web server.

4.1 Hosting Operating System

The operating system is the foundation for the web server to be built upon. By not securing the operating system properly, the web server becomes a public frontend to an unsecure system. Through the web server, unauthorized users have a pathway to the organization's network resources.

When securing the web server, care should be taken to secure files and to set user roles and privileges to the least needed for proper operating system and web server operation. The Web Server SRG does address operating system file permissions for web server files, but within an operating system, there are settings and processes that are outside the realm of the Web Server SRG. There may also be instances where a web server requirement can be met, but without the operating system requirement being met, the web server is not fully secure. To secure the operating system properly, the appropriate Operating System SRG or specific vendor STIG should be used.

4.2 Roles

Defining the user roles properly is essential to secure the web server. Too often, all of the operating system users are given the same roles. Giving users more privileges than necessary allows a user who is not part of the web server administrator role privileges to make web server changes. Taking a look at the roles that the organization wants to implement for privileged users and giving users only the roles required for carrying out each user's duties is crucial. The definition and duties of each role should be done before any user accounts are created and before the web server is installed.

4.2.1 File Permissions

When securing an operating system, many of the requirements rely on privileges and ownership of files. The permissions laid forth by the operating system requirements may or may not be stricter than those of the web server requirements for privileges and ownership. Care must be taken to give the least privileges and ownership to web server files and still allow operation of the web server and the hosted applications. To arrive at the proper least privilege settings for the web server and hosted applications, a test environment must be utilized along with a well-developed test plan to ensure the production server operates properly and as expected.

4.3   Web Server Management

Web server management is the process of providing administrative duties in the configuration, deployment, and sustainment of the web server software, modules, and hosted applications. The management duties can be performed through local (i.e., console) or remote access. Remote access can take many forms, such as through the Internet or through a dedicated management network.

When the management is done locally, the hosting hardware and operating system perform the validation of users, assign permissions or privileges to the user, and enforce file protections. The major web server security concern when management is performed locally is constraining the user to only those files and functions needed to perform their duties.

Remote access has the added security concern of the transmission of data. All remote management to a web server should be encrypted. The encryption of the traffic should begin at the start of the transmission session. The loss of administrative credentials during a non-encrypted session would negate any security that encryption of later traffic would add. Several methods of performing administrative activities remotely are: through third-party software that is used specifically to administer the web server, through secure shells and virtual private networks (VPNs), and through dedicated management networks. Once connected to the hosting system, the user validation, user permissions, and enforcement will once again fall under the domain of the operating system.

Remote access must also be controlled and not easily available and viewable by non-administrative users. Where local access can be controlled through physical barriers, remote access needs to be controlled through electronic barriers such as access lists or management networks. Care should be taken not to bypass security measures already in place to protect the web server when implementing remote access technologies.

4.4   Session Management

There are many aspects to proper session management. With improper session management, a user's session that has been authenticated can be hijacked and used to gain access to protected data and processes. Areas of concern when configuring the web server and applications in the management of session data are:

- Does the session identifier need to be logged and part of the auditing process for later use in the investigation of an incident?
- Can a user send a link to the hosted application containing a valid session identifier allowing an unauthorized user access to the application?
- Are the hosted applications permissive, allowing the client to generate the session identifier?
- Are session identifiers transmitted unencrypted?
- Is the session identifier easily read by client scripts, allowing a user to reveal the session identifier?

- Is the session identifier cached and will not expire on public use computers?
- Is the session identifier only valid for one application or for the entire domain?
- Is the session identifier guessable?

These are only a few of the concerns when securing the session data, and while configurations on the web server can meet some of the concerns, testing the hosted applications for session handling is important. To secure the web server and hosted applications properly, a defense-in-depth approach is necessary to protect the session data, from the generation of the identifier until the session is no longer valid.

4.5   Auditing

Auditing is not one of the security settings that secures the web server from unauthorized users, but it becomes the most important setting when an incident occurs. Without the information stored through auditing, analysis and forensics cannot be performed.

Aspects of auditing that need to be considered and configured properly are:

- Do the auditing files have the proper permissions so that the auditing system can write to the files, but an attacker cannot delete or alter the information?
- Is all the information needed for proper forensics captured?
- Which users and roles can access the information?
- Is the data time stamped properly and accurately?
- Are the proper events logged and are all the elements of the event stored?
- Is the audit data part of an organizational auditing review system?
- Is the audit data part of a backup for later review if necessary?

The web server needs to generate the proper data elements for security-relevant events, and when the data is integrated into a larger organizational audit system, an organization can not only see an attack as it develops on the web server, but the organization can also identify an attack as it develops on the organizational level.

4.6   Transmitted Data Protection

Protecting the data as it is transmitted between the web server and the client is just part of the protection of data during transmission. While great thought is given to encryption of data when transmitted for web servers hosting protected data, it is often forgotten that some public-facing web servers require encryption of data during the authentication or login process.

Also needing protection through encryption are cookies. Each cookie should be encrypted before being transmitted even when the web server is not using an encryption method for data transmission, otherwise the data inside the cookie can be compromised. Cookies often contain session and user information that must be protected, such as session identifiers and user credentials.

When dynamic applications are being hosted, there are often plug-ins or modules installed with the web server to allow the dynamic code access to backend services such as database servers, inventory systems, health systems, or payroll systems. To protect user credentials and application data, encryption of the communication channel between the web server plug-in or module and the backend service must be enforced.

4.7   Virtualization

Virtualization is a means to further separate the web server from the hosting operating system and hardware platform. By implementing the web server on a virtual machine (VM), the administrator can further tune the security settings specifically for the web server without exposing the underlying operating system and hardware running the virtual software. Other advantages to implementing the web server virtually are:

- Easy deployment from a known stable and secure baseline.
- Easy transition of upgrades and patches from test to production.
- Capability to roll back to a known stable and secure baseline when necessary.
- The operating system and hardware hosting the VM are protected from the applications running within the VM.

While using virtualization for web server deployment has many advantages, the applications being deployed to the web server should be examined and tested. Care should be taken to ensure the hosted applications operate correctly and that the applications, when hosted in a VM, do not open security risks. Issues that need to be evaluated are:

- Are the hosted applications sharing hardware resources such as hardware memory and drive space?
- Are the web server and hosted application files at more risk being stored on virtual drives?
- Are the operating system and hardware hosting other applications within the VM that may compromise the web server?
- Are all the applications hosted in the VM at the same security level?
- Does the organization have the expertise to manage and secure VM software?

To properly and securely run a web server within a VM, the VM needs to meet the vendor STIG for the software. This extra layer of expertise to secure the VM is often not available. Without properly securing the VM software, the advantages of implementing a VM solution for security reasons may be lost.

4.8   Configuration Management

Configuration management is often forgotten when securing a web server. Being able to consistently install a baselined system to production without error or security flaws is important. Methods used to easily and consistently build a server from a secure baseline are through a well-documented process or by using VM images.

During the operation of the web server, proper testing of patches and upgrades must be performed. Having a documented process for testing and implementation helps move patches and upgrades from the lab to production with very little issue.

Configuration management also includes performing a risk assessment of the web server and hosted applications and then implementing a risk management plan. During the assessment, the impact to the organization of a web server failure is analyzed and contingency planning is performed. During the analysis, plans are formed for tasks such as how often backups are performed and where the backups will be stored, and backup equipment and locations are procured. When the web server is a high-priority system, web clustering may be implemented and disaster exercises performed to limit the exposure to a Denial of Service (DoS) during a failure either due to hardware, natural disaster, human error, or a security incident.

4.9   Software Installation

The first step to secure a web server is completed before the web server is even installed. Determining the services that are needed for the operation of the web server and proper operation of the hosted applications need to be completed, and then the web server software can be installed.

Too often, web servers are installed with the default settings, leading to the installation of sample and demo software, configurations with default settings, and more services, plug-ins, and modules installed than are needed. Removing the unneeded services and code is then either difficult to accomplish or the administrators are time-constrained and never find time to remove the extra software. With each extra service or bit of code, security flaws can be exposed that may not be discovered or fixed since these modules are neither part of the daily patch routine nor used during normal operation.