

UNCLASSIFIED



**SOFTWARE DEFINED NETWORKING (SDN)
CONTROLLER
SECURITY REQUIREMENTS GUIDE (SRG)
TECHNOLOGY OVERVIEW**

Version 1, Release 2

24 April 2020

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
1.1.1 Security Requirements Guides (SRGs)	1
1.1.2 SRG Naming Standards	2
1.2 Authority	2
1.2.1 Relationship to STIGs.....	3
1.3 Vulnerability Severity Category Code Definitions	3
1.4 SRG and STIG Distribution	3
1.5 Document Revisions	3
1.6 Other Considerations	4
1.7 Product Approval Disclaimer.....	4
2. ASSESSMENT CONSIDERATIONS.....	6
2.1 NIST SP 800-53 Requirements	6
2.2 General Procedures	6
3. CONCEPTS AND TERMINOLOGY CONVENTIONS	7
3.1 SDN Overview	7
3.2 SDN Controller	7
3.3 Application Program Interface (API).....	7
3.3.1 Southbound API.....	8
3.3.2 Northbound API.....	8

LIST OF TABLES

	Page
Table 1-1: Vulnerability Severity Category Code Definitions	3

LIST OF FIGURES

	Page
Figure 3-1: APIs.....	8

1. INTRODUCTION

1.1 Executive Summary

Software Defined Networking (SDN) is an approach to networking that separates the control plane from the forwarding plane. SDN is also new paradigm for network virtualization. Most SDN architectures have three layers: an upper layer that includes applications and services that contain security policy, network management automation, orchestration platforms, and business drivers; a lower layer of physical and virtual SDN-aware network devices that forward all traffic within the SDN infrastructure; and, in the middle, SDN controller(s) that communicate with the upper layer via northbound application program interfaces (APIs) and with the lower layer via southbound APIs. The latter is the control plane framework that enables the controller to provide forwarding tables to the network devices. Because the SDN controller is the heart of the SDN infrastructure, it is critical that the controller is secured.

This SDN Controller Security Requirements Guide (SRG) provides the guidelines and requirements for implementing security measures for SDN controllers that will enable network operations to minimize the risk of a network outage or compromise. This SRG does not include security guidelines for the development of northbound or southbound APIs that may be integrated with the SDN controller; those guidelines would be found in the Application SRG. The Network Device Management (NDM) SRG provides security guidelines for the management of the SDN controller.

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 SDN Controller SRG is based on the Network SRG. This SDN Controller 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 provides 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: Router, Switch, Firewall, 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 Cyber Exchange website at <https://cyber.mil/>. This site contains the latest copies of STIGs, SRGs, and other related security information. Those without a Common Access Card (CAC) that has DoD Certificates can obtain the STIG from <https://public.cyber.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 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.

1.7 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-ccavs.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 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.

3. CONCEPTS AND TERMINOLOGY CONVENTIONS

3.1 SDN Overview

Software-defined networking is an emerging network paradigm where the control plane is decoupled from the data plane to improve network flexibility and manageability. The control plane makes decisions as to which way traffic is sent. The control plane function includes the system configuration, management, and exchange of routing table information. The data plane, also known as the forwarding plane, forwards traffic to the next node along the path to the selected destination according to control plane logic. With SDN, forwarding decisions that traditionally are computed by individual network elements will migrate to a controller that abstracts a logical view of the network. Network intelligence and state are now centrally maintained in an SDN controller or cluster of controllers. In some SDN implementations, forwarding decisions can migrate to the control plane and the centralized controller.

3.2 SDN Controller

An SDN controller is the central repository for control instructions, data flow logic, security policies, and business policies required to deploy, configure, and manage network elements to obtain the desired network behavior. The controller provides a programmatic interface for the provisioning of network services using a consistent approach.

The SDN controller uses an open or proprietary protocol to control each network element and the traffic flow on the network. SDN relies heavily on control messages between a controller and the forwarding devices for reliable network operation.

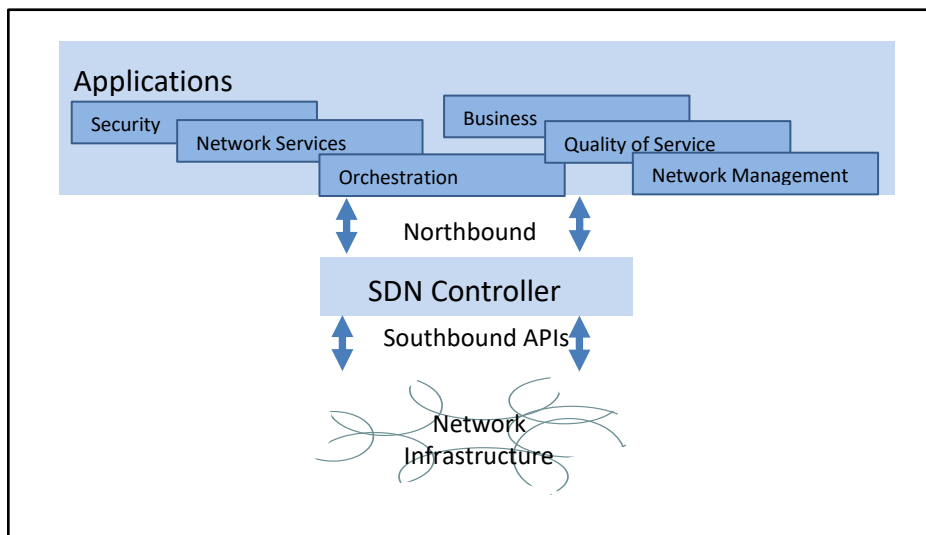
The controller also extracts state information about the network from the network elements and communicates that information back to the application with an abstract view of the network, including statistics and events about what is happening.

3.3 Application Program Interface (API)

SDN is a shift in networking that breaks traditional physical boundaries of network elements through well-defined APIs. An API is an interface presented by software that provides the capability to collect information from, or make changes to, an underlying resource. An API makes it possible to dynamically and programmatically provision and manage a network through software. The API is the control point for each component of the SDN-enabled network infrastructure, including switches, routers, SDN controllers, orchestration systems, and network management systems.

APIs between the SDN controller and the application layer enable business applications to alter network behavior and deploy network services based on business requirements without the burden of network implementation details. As shown in Figure 3-1, the SDN architecture APIs, often referred to as northbound and southbound interfaces, provide the communication between the applications, controllers, and network elements.

Figure 3-1: APIs



3.3.1 Southbound API

Southbound APIs enable the SDN controller to make changes dynamically according to real-time demands and business needs. Southbound API messages can be categorized as either control plane or management plane traffic. A southbound API implemented between the SDN controller and the network elements provides the mechanism to enable the controller to send forwarding table updates to the network elements. These messages would all be considered control plane traffic, whereas management plane traffic consists of messages used by the controller to provision and configure network elements.

The API also provides the vehicle for the controller to receive new state information from the network elements. This can include the concept of flows to identify network traffic based on predefined match rules that can be statically or dynamically programmed by the SDN control software. The controller defines how traffic should flow through network elements based on policy, usage, applications, and available bandwidth.

3.3.2 Northbound API

Via communication with the SDN controller, the northbound API enables applications, management systems, and orchestration systems to program the network. A network abstraction can be presented to applications and management systems via northbound API call to the controller. It enables developers to create network applications without the need to call the southbound API directly.

Northbound APIs are also used to integrate the SDN controller with automation stacks as well as orchestration platforms. The northbound APIs can be used to enable orchestration and automation of the network to align with the needs of different applications. Network services that can be deployed and optimized via this API include security services, load balancing, traffic engineering, and quality of service.