

UNCLASSIFIED



ADOBE COLDFUSION 11 SECURITY TECHNICAL IMPLEMENTATION GUIDE (STIG) OVERVIEW

Version 1, Release 4

26 January 2018

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. CONCEPTS AND TERMINOLOGY CONVENTIONS	4
2.1 Adobe ColdFusion Enterprise 11	4
2.2 ColdFusion-Based Applications.....	4
2.2.1 ColdFusion Markup Language	4
2.2.2 Web-Based Applications	4
2.2.3 Mobile-Based Applications	5
2.3 User Accounts	5
2.3.1 ColdFusion Administrator Accounts	6
2.3.1.1 Authentication	6
2.3.1.2 Root Administrator Account	6
2.3.2 Operating System Accounts	6
2.3.3 Data Source Accounts.....	7
3. GENERAL SECURITY REQUIREMENTS	8
3.1 Operating System User Roles	8
3.2 Web Server.....	8
3.3 ColdFusion Management	9
3.4 Sandboxing.....	9
3.5 Application Design and Development	10
3.6 Configuration Management	10
3.7 Software Installation	10

LIST OF TABLES

	Page
Table 1-1: Vulnerability Severity Category Code Definitions	2

1. INTRODUCTION

1.1 Executive Summary

The Adobe ColdFusion 11 Security Technical Implementation Guide (STIG) is published as a tool to improve the security of Department of Defense (DoD) information systems. The STIG will address ColdFusion Enterprise 11 in a production environment hosted on either Linux or Windows. The Adobe ColdFusion 11 STIG will not address development environments, hosted applications, or other external applications such as web servers, the Java Virtual Machine (JVM), databases or email servers, but the communications between these different entities and Adobe ColdFusion 11 will be addressed.

This document is a requirement for all DoD-administered systems and all systems connected to DoD networks. These requirements 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. This guidance supports DoD system design, development, implementation, certification, and accreditation efforts.

While researching to write the Adobe ColdFusion 11 STIG, the *ColdFusion 11 Lockdown Guide*, written by Pete Freitag, was used. The document gives details on how to install the product correctly and general security settings, but it also goes into great detail on how to lockdown ColdFusion 11 when used with different support application setups, and it also sheds light on areas that should be tuned, depending on hardware and user load, that will lockdown the system even further. The guide can be found at <http://www.adobe.com/content/dam/Adobe/en/products/coldfusion/pdfs/cf11/cf11-lockdown-guide.pdf>

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.

1.3 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. CONCEPTS AND TERMINOLOGY CONVENTIONS

This overview is not intended to be a comprehensive source of information on Adobe ColdFusion Enterprise 11 security but is meant to provide an understanding of the background, concepts, terminology, and boundaries used in the Adobe ColdFusion 11 STIG. With an understanding of the underlying terms and concepts, the STIG can be better utilized to attain a secure product.

2.1 Adobe ColdFusion Enterprise 11

Adobe ColdFusion Enterprise 11, known simply as ColdFusion and addressed as so for the remainder of this document, is an application server, which means that it is a component-based application that resides in the middle-tier of a multi-tier architecture. An application server handles all application operations between users and the organization's backend business applications or databases.

Adobe markets ColdFusion Enterprise 11 Edition as “a single platform to rapidly build and deploy scalable, high-performing web and mobile enterprise applications. Leverage unique capabilities to develop, test, and debug mobile applications.” The STIG will not address the development, testing and debugging capabilities of ColdFusion. These activities must not be performed in a production environment, and care must be taken to make certain the avenues for these activities are disabled.

2.2 ColdFusion-Based Applications

While the applications based on ColdFusion are not addressed within the STIG directly, the type of applications being hosted and the interaction between those applications and the ColdFusion application server need to be understood fully. This enables the ColdFusion administrator to disable development and unused services and to tune those services that are in use.

2.2.1 ColdFusion Markup Language

ColdFusion Markup Language (CFML) is a proprietary markup and scripting language. In its simplest form, CFML is a tag-based Web scripting language that supports dynamic Web page creation and database access in a web server environment. The CFML tags are embedded in HTML files. The HTML tags determine the page's layout, while the CFML tags import content based on user input or the results of a database query.

CFML has expanded into a scripting language also known as CFScript. CFScript resembles JavaScript. Some ColdFusion developers prefer it since it has less visual and typographical overhead than ordinary CFML.

2.2.2 Web-Based Applications

Dynamic web pages are used to interact with the user and give them a productive and sometimes entertaining experience. ColdFusion gives the web developer the capability to insert CFML or

CFScript code into web pages that are passed to ColdFusion, where it is interpreted or executed to generate HTML code that can be viewed through a web browser. ColdFusion gives developers the capability to build highly interactive and data-rich websites, which, according to Adobe, can do tasks such as:

- Query other database applications for data
- Dynamically populate form elements
- Dynamically generate Flash data
- Provide application security
- Integrate with other systems using standard protocols such as HTTP, FTP, LDAP, POP, and SMTP
- Create shopping carts and e-commerce websites
- Respond with an email message immediately after a user submits a form
- Return the results of keyword searches

2.2.3 Mobile-Based Applications

Mobile applications are developed specifically for use on small, wireless computing devices, such as smartphones and tablets, rather than desktop or laptop computers. ColdFusion divides these into three types:

- **Type 1 - HTML5-based standalone applications** that can be installed on the mobile device as a standalone application. This type of application can be device specific.
- **Type 2 - ColdFusion-deployed web applications** that can be rendered through the mobile device's web browser. There is no support for the device's native functionalities.
- **Type 3 – Hybrid/shell applications** that can run as a standalone mobile application. This type supports the device's native functionalities.

While these three types are important, and each has its own security considerations, the ColdFusion application server can consider them all as just mobile applications with the same security considerations. Each type is making CFML/CFScript calls, which are then generated into HTML/JavaScript or HTML5 and sent back to the mobile device.

2.3 User Accounts

During the installation and daily maintenance of ColdFusion, different types of user accounts are used and created. These accounts must be limited in access and privileges to restrict the user to only those functions and objects needed for the user's role. These accounts live inside ColdFusion as administrator accounts and outside ColdFusion as operating system and data source accounts.

2.3.1 ColdFusion Administrator Accounts

ColdFusion accounts are accounts, managed through the Administrator Console and used to configure and maintain the ColdFusion application server. These accounts can have privileges from as little as just being able to log onto the console all the way up to being able to see and modify data source connections, view and modify logging, and create and modify accounts. Careful consideration of the roles needed for each user must be taken when creating accounts, and proper authentication methods must be used to log the user activities.

2.3.1.1 Authentication

Logging is essential to track which users are accessing the ColdFusion Administrator Console and to track what actions the users are taking. To log the user's actions, the correct authentication method must be used. ColdFusion offers three methods of user authentication. They are:

- **Use a single password only.** This method is the default and must be changed after installation. This method, while affording some security, does not allow for proper logging of user activities or the ability to assign roles to users.
- **Separate username and password authentication.** This method allows there to be multiple users, users must be authenticated, roles can be defined for each user, and logging data can be generated that tie the user to their actions.
- **No authentication needed.** This method is the least secure, relying on firewalls and the internal ColdFusion access list to control which device may access the console. Since the actions taken within the console are not tied to a user, there is no way to tie changes to the configuration to a user, roles cannot be defined for each user, and, since authentication is for a device, there is no guarantee the user on the device should have access.

2.3.1.2 Root Administrator Account

The root administrator account is set up during the installation of ColdFusion. This account is used for initial setup and has privileges to all areas of the Administrator Console. Because of this access, the username and password should be considered carefully and follow any username and password policies for the organization. The password can be changed through the ColdFusion Administrator Console, but the username cannot. When creating the account, the username must not be "admin" or "administrator" with variations on character case. These usernames are easily guessed, allowing an attacker to concentrate his efforts on cracking the password.

2.3.2 Operating System Accounts

There are two types of accounts that are maintained by the hosting operating system that will affect the ColdFusion security posture. The first type is accounts used by operating system users to access ColdFusion utilities and files. The accounts that have access should be limited to those users that have roles that require access. Access is controlled by the operating system through proper assignment of file permissions.

The second type of account is the account ColdFusion will execute as. This account will own all the ColdFusion files and utilities and is usually locked or does not allow users to log on with the account. This account is also set up to have little to no privileges outside the execution of the ColdFusion application server. The document *ColdFusion 11 Lockdown Guide* describes the setup and permissions to secure this type of account for both Windows and Linux.

2.3.3 Data Source Accounts

ColdFusion defines a data source as a complete database configuration that uses a JDBC driver to communicate with a specific database. To establish the communication channel, a username and password are usually used. The username/password credentials are maintained by the database and must follow the restrictions set for account names and password complexity within the appropriate database SRG or STIG, and the account must be limited to only those objects required for application operation. The account should be treated as any user account to the database since compromise, either through ColdFusion misconfiguration or at the database, may cause data leakage and loss or a complete DoS for the applications accessing the database.

ColdFusion can further secure the account and the connection by limiting the commands that are allowed to be executed on the database. Limiting the commands thwarts an attack where an attacker executes ColdFusion code with calls to the data source that perform commands such as creating and deleting objects within the database. If these commands are not allowable commands for the data source, the attacker is unable to create objects for later use or delete database objects.

3. GENERAL SECURITY REQUIREMENTS

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

3.1 Operating System User Roles

The operating system is the foundation that ColdFusion is built upon. By not securing the operating system properly, ColdFusion becomes a gateway to an unsecure system and infrastructure. Through the application server, unauthorized users have a pathway to the organization's network resources and internal applications.

Defining the user roles properly is essential to secure ColdFusion. 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 ColdFusion administrator role, privileges to make application 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 completed before any user accounts are created and before ColdFusion is installed.

3.2 Web Server

ColdFusion can easily be installed with the internal web server. This web server is essentially used to host the ColdFusion Administrator Console. While this web server hosts the console effortlessly, the web server is not easily configurable to encrypt sessions through HTTPS, allowing user credentials to be quickly discovered.

To securely host applications, including the ColdFusion Administrator Console, and enable FIPS 140-2 approved encryption modules, ColdFusion works with several commercial web servers such as IIS and Apache. The use of commercial web server software also guarantees quick and timely security patches and bug fixes from a vendor who is dedicated to supporting the software. This is not guaranteed with the ColdFusion internal web server, where the main purpose of the product, ColdFusion, is to be an application server, not a web server.

To host the ColdFusion Administrator Console to an external web server and have ColdFusion communicate securely with the web server, review the *ColdFusion 11 Lockdown Guide* for the steps and techniques. The appropriate SRG or STIG should also be used for the web server in use to secure the web server and ColdFusion Administrator Console.

3.3 ColdFusion Management

ColdFusion management is the process of providing administrative duties in the configuration, deployment, and sustainment of the ColdFusion software, modules, and data sources. 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 ColdFusion 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 ColdFusion must 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 the ColdFusion Administrator Console, through secure shells and virtual private networks (VPNs), and through dedicated management networks.

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 ColdFusion when implementing remote access technologies.

3.4 Sandboxing

Application isolation allows multiple applications to run on the same hosting operating system, web server, and application server. Typical reasons to isolate applications are to separate different application user bases, data security levels, protect application resources, and to give least privileges to each application to system resources. Application isolation will also contain an application that has been compromised from compromising other hosted applications.

To isolate applications within ColdFusion, sandboxes are set up for applications to execute within. ColdFusion sandboxing allows the administrator to restrict access to data sources, ColdFusion tags and functions, directories, and network access. This allows the administrator to partition the shared hosting environment so that a number of applications with different purposes, and possibly different owners, run securely on a single server. When multiple applications share a host, a separate directory structure is created for each application with rules applied to the structure that only allow an application to access its own data sources and files.

3.5 Application Design and Development

The application design and development process is not often thought of when securing an application server. Decisions made during the application design can significantly affect the security of the overall system. For instance, deciding not to sandbox the application and allowing the application to have access to all the ColdFusion resources may allow an attacker to compromise the hosted application, gain access to backend resources, such as email services, and use those services to generate SPAM to the organization. Security within the hosted applications needs to be part of the overall design process. Documents that must be used during the development process are the *ColdFusion 11 Developer Security Guide* and the *Application Security and Development STIG*.

3.6 Configuration Management

Configuration management is often forgotten when securing ColdFusion. 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 ColdFusion application server from a secure baseline are through a well-documented process or by using Virtual Machine (VM) images.

During the operation of ColdFusion, 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.

Outside the testing of patches for ColdFusion, patches to applications being hosted must be validated before being introduced to the production environment. A module that has not been tested and is moved to production may contain errors or work differently in production than in a development environment. This may lead to excessive resource usage, which may cause applications to slow or eventually lead to a Denial of Service (DoS), buffer overflows, invalidated inputs or parameters, and other risks to the overall ColdFusion security posture.

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

3.7 Software Installation

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

Too often, ColdFusion is installed with the default settings, leading to the installation of sample and demo software, configurations with default settings, and more services installed than are needed. Removing or limiting the unneeded services and code is not difficult to accomplish, but often the administrators are time constrained and never find time to remove the extra software or reanalyze what is needed. With each extra service or bit of code, security flaws can be exposed that may not be discovered or fixed.