National Information Assurance Partnership

TM

Common Criteria Evaluation and Validation Scheme Validation Report

CA SiteMinder Web Access Manager r12 SP1-CR3

Report Number: CCEVS-VR-VID10317-2009
Version 1.0
12 June 2009

National Institute of Standards and Technology
Information Technology Laboratory
100 Bureau Drive
Gaithersburg, MD 20899

National Security Agency
Information Assurance Directorate
9800 Savage Road STE 6757
Fort George G. Meade, MD 20755-6757

Table of Contents

1. Executive Summary

The Target of Evaluation (TOE) is the CA SiteMinder Web Access Manager r12 SP1-CR3 product. The TOE was evaluated by the Booz Allen Hamilton Common Criteria Test Laboratory (CCTL) in the United States and was completed in May 2009. The evaluation was conducted in accordance with the requirements of the Common Criteria, Version 3.1 Revision 2 and the Common Methodology for IT Security Evaluation (CEM), Version 3.1 Revision 2. The evaluation was for Evaluation Assurance Level 3 (EAL3) augmented with ALC_FLR.1 (Basic Flaw Remediation) and ASE_TSS.2 (TOE Summary Specification). The evaluation was consistent with National Information Assurance Partnership (NIAP) Common Criteria Evaluation and Validation Scheme (CCEVS) policies and practices as described on their web site (www.niap.ccevs.org).

CA SiteMinder product is an enterprise-scale Web access management system that provides access control to Web applications and portals from the organization's employees, customers and business partners. The CA SiteMinder product, when configured as specified in the installation guides and user guides, satisfies all of the security functional requirements stated in the TOE's Security Target.

The cryptography used in this product has not been FIPS certified nor has it been analyzed or tested to conform to cryptographic standards during this evaluation. All cryptography has only been asserted as tested by the vendor.

The technical information included in this report was largely derived from the Evaluation Technical Report and associated test reports produced by the evaluation team. The *CA Siteminder Web Access Manager R12 SP1-CR3 Security Target*, Version 0.8, dated May 29, 2009 identifies the specific version and build of the evaluated TOE. This Validation Report applies only to that ST and is not an endorsement of the CA SiteMinder product by any agency of the US Government and no warranty of the product is either expressed or implied.

2. Evaluation Details

Evaluated Product	*CA SiteMinder Web Access Manager r12 SP1-CR3*
Sponsor & Developer	CA Inc., Islandia, NY
CCTL	Booz Allen Hamilton, Linthicum, Maryland
Completion Date	27 May 2009
CC	*Common Criteria for Information Technology Security Evaluation*, Version 3.1 Revision 2, September 2007
Interpretations	None.
CEM	*Common Methodology for Information Technology Security Evaluation*, Version 3.1 Revision 2, September 2007
Evaluation Class	EAL3 Augmented with ALC_FLR.1 and ASE_TSS.2
Description	The TOE is the CA SiteMinder Web Access Manager r12 SP1-CR3 software, which is a Web Access Control system developed by CA.

Disclaimer	The information contained in this Validation Report is not an endorsement of the SiteMinder product by any agency of the U.S. Government, and no warranty of the SiteMinder product is either expressed or implied.
PP	None
Evaluation Personnel	Justin Fisher John Schroeder Amit Sharma
Validation Body	NIAP CCEVS

2.1. Threats to Security

Table 2 summarizes the threats that the evaluated product addresses.

Table 2 – Threats

Authorized users could gain electronic access to protected network resources by attempting to establish a connection that they are not permitted to perform.
An administrator may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.
A malicious user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.
A malicious user could eavesdrop on network traffic to gain unauthorized access to TOE data.
Users whether they are malicious or non-malicious, could gain unauthorised access to the TOE by bypassing identification and authentication countermeasures.
A user may masquerade as an authorized user or an authorized IT entity to gain access to data or TOE resources.
Users could gain unauthorised access to the web resources by bypassing identification and authentication requirements.

3. Identification

The product being evaluated is the CA SiteMinder Web Access Manager r12 SP1-CR3.

4. Security Policy

4.1. Access Control

The Siteminder WAM Administrative UI authenticates SiteMinder administrator accounts using the Administrator Store. Administrators accessing the Policy Server Management Console are required by the underlying OS to enter a username and password, but they are not required to authenticate to the TOE via one of the authentication schemes mentioned in this document (see section 2.2 of the INT for more information).

Depending on one's role in an organization, SiteMinder administrators have access to different resources and features, and are responsible for different tasks, based on policies created via the Administrative UI. These policies define rules and the tasks which are performed by the domain administrator in the policy domain. At least one administrator must be assigned to each domain. An administrator can be assigned to more than one domain; however, he can only perform the duties within the scope of the domain(s) to which he's assigned. The SiteMinder administrative model implements fine-grained administrative

privileges, so the management of Policy Server objects and SiteMinder tools across a few or many individuals in an organization are organized accordingly.

The remote Super User account is included in the TOE and has all of the same attributes as an administrator. This account is the default account that is set up at installation and is used to create all other Administrator accounts and also assigns the categories, rights, and scope of those accounts. It is not recommended that the Super User account be used for day-to-day administration. An administrator can only create another administrator with the same or lesser privileges. For example, an administrator with GUI and reports privileges cannot create an administrator with GUI, reports privileges and local API privileges. Administrator privileges are determined by the tasks that are enabled for the administrator. These privileges allow administrators to use a set of Policy Server features.

Administrators have the ability to create end users and to assign all end users to groups as needed. These end users and/or groups are then added to policies in order to gain access to protected resources. A policy exists as part of a Policy Domain. The policy defines the type of access for an end user. A policy binding is created between the selected end users and the policy when an administrator adds an end user or group to the policy. When an end user tries to access a protected resource, the policy verifies that the end user is part of its policy binding, and then enforces the rules included in the policy to see if the end user is allowed to access the resource, if any authentication and authorization events must be processed, and if any responses should be generated and returned to SiteMinder Web Agents. When the Policy Server processes rules, it looks for the longest matching string that consists of a resource filter specified in the realm and a resource specified in the rule.

4.2. Identification and Authentication

Each end user must be successfully identified and authenticated before being allowed access to protected resources. The Policy Server verifies an end user's identity by retrieving user attributes contained in the User Store. The TOE employs authentication schemes associated with the resource's realm and protection level. Authentication schemes provide a way to collect credentials and determine the identity of an end user. If an end user tries to access a resource with a higher protection level than the one he is currently accessing, he will be required to re-authenticate. If an end user attempts to access a resource with an equal or lower protection level within the same session, he is not required to re-authenticate. The TOE supports a variety of authentication schemes. In the evaluated configuration the following schemes are used: basic username/password authentication, Windows authentication schemes, and digital certificate authentication (x.509). Simple schemes can be used for low risk network resources, while complex schemes may be employed to ensure added security for critical network resources. Authentication schemes must be configured using the WAM Administrative UI. During authentication, SiteMinder Web Agents communicate with the Policy Server to determine the proper credentials that must be retrieved from an end user who is requesting resources.

4.3. Security Management

Security Management is implemented by the TOE through interactions by the administrator remotely using the web-based WAM Admin UI and on the local machine via the Policy

Server Management Console. For remote access the administrator logs into the WAM Admin UI via an SSL enabled web browser. Once authenticated, the administrator has access to security management tabs within the SiteMinder software that enable him/her to configure user accounts, policies, and other management functions allowed by the TOE.

In the evaluated configuration, the Policy Server Management Console is used solely to perform initial configuration and startup of the TOE. Once the TOE is operational, administrators of the TOE will interact with the WAM Admin UI to perform all necessary management activities. While the Policy Server Management Console has a variety of functions, there is a lot of overlap with the capabilities of the WAM Admin UI and initial setup is the only capability that is within the evaluated configuration.

4.4. Audit

Auditing allows the administrators to track end user and administrative activity as well as analyze and correct security events and anomalies. Objects, events, and activities to be audited are defined by the administrator. At a minimum, the audit record for end user and administrator actions on the TOE stores the date and time the record was created, the remote server host name and ID, the remote server account name responsible for the report creation, the object/resource scanned, status, and the total number of objects/resources scanned. In the evaluated configuration, the date, time, type of event, subject identity, and outcome (success or failure) is logged for each audit event, which includes the startup and shutdown of audit functions. Based on the content of these logs, the TOE is able to associate the event with the user that caused the event.

The audit logs are stored in local audit files in the Operational Environment separate from the Policy Server. By storing the logs separately from the Policy Server, the Operational Environment is able to protect the audit records from unauthorized deletion and protect unauthorized modifications to the records in the audit trail. Both the Policy Server and Web Agent provide separate audit logging. All audit logging relies on the underlying operating system in the Operational Environment to provide reliable time stamps.

The Web Agent uses an auditing feature which allows the administrator to track and log successful authorizations of an end user; these audited events are stored in a local audit log file called the Trace Log. This allows the administrator to track user and role activity, and to measure how often applications on a particular web site are being used. For an administrator to view these logs, they must authenticate to the OS which the Web Agent and Web Server are installed on and have equal or higher privileges then the account used to install the TOE components.

4.5. Load Balancing and Failover

Load balancing and failover in a SiteMinder deployment provide a high level of system availability and improve response time by distributing requests from SiteMinder Agents to Policy Servers. Defining clusters in combination with load balancing and failover further enhances the level of system availability and system response time. A cluster is defined as a set of one or more Policy Servers grouped according to customer-defined criteria, and with load-balancing between the servers. Each Policy Server is completely independent of the other Policy Servers within its cluster and within other clusters.

All load balancing is done via the Web Agent API Layer logic. The Web Agent API Layer is responsible for dynamically balancing the load between Policy Servers in a cluster based on server response time, and for failing-over to another cluster under the failover criteria. For example, if the response time for one Policy Server within a cluster is too slow, the Web Agent API Layer will defer the request to the another Policy Server within that cluster. If the entire primary cluster fails and failover is enabled, a backup cluster takes over policy operations.

4.6. Encrypted Communications

A trusted path is established for all communications and interactions with the TOE to ensure that all traffic communications to and from the TOE are protected from unauthorized access. The TOE relies on the Operational Environment to provide all protected communications from the TOE to a user of the TOE. In the evaluated configuration the Operational Environment will be configured to have SSL 3.0 encryption over these paths.

Communications between the end user and the SiteMinder Web Agent are done through HTTP over SSL v3.0 with SHA-256 on an ASF Apache 2.2 or SunONE 6.1 SP2 web server, or SHA-1 on an IIS 6.0 server.

Communications between the administrator and the SiteMinder WAM Admin UI are performed through the environmental application server using SSL v3.0.

In the case of the Apache web server interface, the end user initiates authentication to the web server components of the TOE using digest authentication from Apache, specifically the Apache module mod_auth_digest controls the encryption of the passwords, and protects the TOE from replay attacks. During the I&A process, the end user performs the SSL protocol handshake, is prompted with a login pop up window, and is allowed to enter I&A credentials. Note that through this interface an end user is authenticated as part of the SSL protocol handshake using an RSA key pair.

4.7. Encrypted Data

The Policy Server creates Session Keys using AES with HMAC-SHA256. The Session Keys are utilized by the Policy Server and SiteMinder agents for protecting the TCP/IP message exchange between these components. AES Key Wrap is used by the Policy Server to create Session Ticket Keys and Agent Keys. The Policy Server will then place these keys through the FIPS-140 Key Expansion Algorithm to generate an AES Key which is utilized by the TOE to encrypt TOE data. The Web Agent utilizes the AES Key derived from the Agent Key to encrypt Single-sign on (SSO) cookies, which are used by the Web Agent to identify and authorize users which are trying to access protected resources. The Policy Server utilizes the AES Key derived from the Session Ticket Key to encrypt Password Services data in the user stores, and sensitive data (keys, shared secrets, passwords) in the Policy Stores.

5. Assumptions

5.1. Personnel Assumptions

Table 1 – Personnel Assumptions

One or more authorized administrators will be assigned to install, configure and manage the TOE and the security of the information it contains in order to maintains its security objectives.
System Administrators exercise due diligence to update the TOE with the latest patches and patch the Operational Environment (e.g., OS and database) so they are not susceptible to network attacks.
Users and administrators of the TOE are not careless, wilfully negligent, or hostile and will follow and abide by the instructions provided by the organization's guidance documentation.

5.2. Physical Assumptions

Table 2 – Physical Assumptions

The TOE will be located within controlled access facilities that will prevent unauthorized physical access.

5.3. Logical Assumptions

Table 3 – Logical Assumptions

There are no logical assumptions for operation of the TOE.

6. Clarification of Scope

The TOE includes all the code that enforces the policies identified (see section 4).

The evaluated configuration of the TOE includes the CA SiteMinder r12 SP1-CR3 application that is comprised of the following:

Policy Server/WAM Admin UI system:

	Policy Server / WAM Administrative UI			
#	OS Version	LDAP Server	Relational Database	Application Server
1	Windows Server 2003 SP2	Active Directory win2k3	Oracle 10g R2	Jboss 4.0.5
2	Red Hat Advanced Server 4.0	iPlanet 5.2	Oracle 10g R2	Jboss 4.0.5
3	Solaris 10	iPlanet 5.2	Oracle 10g R2	WebLogic 9.2

Web Agent system:

	Web Agent	
#	OS Version	Web Server
1	Windows Server 2003 SP2	IIS 6.0 on Win2k3 SP2

		ASF Apache 2.2 on Win2k3 SP2
2	Red Hat Advanced Server 4.0	Sunone 6.1 SP2 on RHAS 4.0
		ASF Apache 2.2 on RHAS 4.0
3	Solaris 10	Sunone 6.1 SP2 on Solaris 10
		ASF Apache 2.2 on Solaris 10

The evaluated configuration does not include the following features described in the user manual:

- Local configuration files
- Non-persistent sessions (see section 9.1.3.1 Session Ticket Management)
- Use of the Static Agent Key for cookie encryption
- The native Operating System
- Support for RADIUS
- User tokens (e.g. smartcards)
- Multiple Policy Stores
- Virtual Servers
- Proxy Servers (including Reverse Proxy Servers)
- Domino Application Servers
- Security Zones
- Administrative Journal and Event Handler
- Nested Security
- OneView Monitor
- Simple Network Management Protocol (SNMP) Module
- Event Manager Application
- Directory Mapping
- The following Authentication Schemes - CRYPTO Card RB-1, HTML forms, MS Passport, RADIUS CHAP/PAP, RADIUS Server, Safeword Server, Safeword Server and SecurID, TeleID, Anonymous, Custom, Impersonation, Certificate Mapping
- Credentials Collector
- Variables (Static, Request Context, User Context, Form Post)
- Impersonation

The scope and requirements for the evaluated configuration are summarized as follows:

1. The SiteMinder r12 SP1-CR3 software (i.e., the TOE) will be installed as three components: the Policy Server, WAM Admin UI, and Web Agent.

2. The TOE requires certain environmental components to be present prior to installation. The Web Agent requires a configured web server and the Policy Server requires a configured LDAP server, relational database, and application server for WAM Admin UI.

6.1. System Requirements

This section identifies the hardware and software requirements for the platforms described in the evaluated configuration. The TOE was evaluated in Windows Server 2003 SP2, Red Hat Advanced Server 4.0, and Solaris 10.

6.1.1. Policy Server and WAM Admin UI

For the evaluated configuration, the Policy Server and WAM Admin UI require the following hardware:

Component	Windows or Linux	Solaris Unix
CPU	Single or Dual-processor, Intel Pentium III (or compatible), 700-900 MHZ	Sparc Workstation 440 MHz
Memory	512 MB system RAM. 1 GB is recommended	512 MB system RAM. 1 GB is recommended
Available Disk Space	540 MB for the AdminUI	540 MB for the AdminUI
	270 MB for the Policy Server	300 MB for the Policy Server
Temp Directory Space	450 MB for the AdminUI	450 MB for the AdminUI
	180 MB for the Policy Server	200 MB for the Policy Server

6.1.2. Web Agent

For the evaluated configuration, the Web Agent requires the following hardware:

Component	Windows or Linux	Solaris Unix
CPU	Single or Dual-processor, Intel Pentium III (or compatible), 700-900 MHZ	Sparc Workstation 440 MHz
Memory	512 MB system RAM	512 MB system RAM
Available Disk Space	1 GB	1 GB

6.2. Physical Boundary Components

Section 6.2.1 (Hardware Components) and section 6.2.2 (Software Components) denote the components that are in the TOE and that are in the environment.

6.2.1. Hardware Components

Table 6 identifies hardware components and indicates whether or not each component is in the TOE.

Table 6 – Hardware Specifications

Environment	Policy Server and WAM Admin UI Platform	See section 6.1.1

Environment	Web Agent Platform	See section 6.1.2

6.2.2. Software Components

Table 7 identifies software components and indicates whether or not each component is in the TOE.

Table 7 – Software Specifications

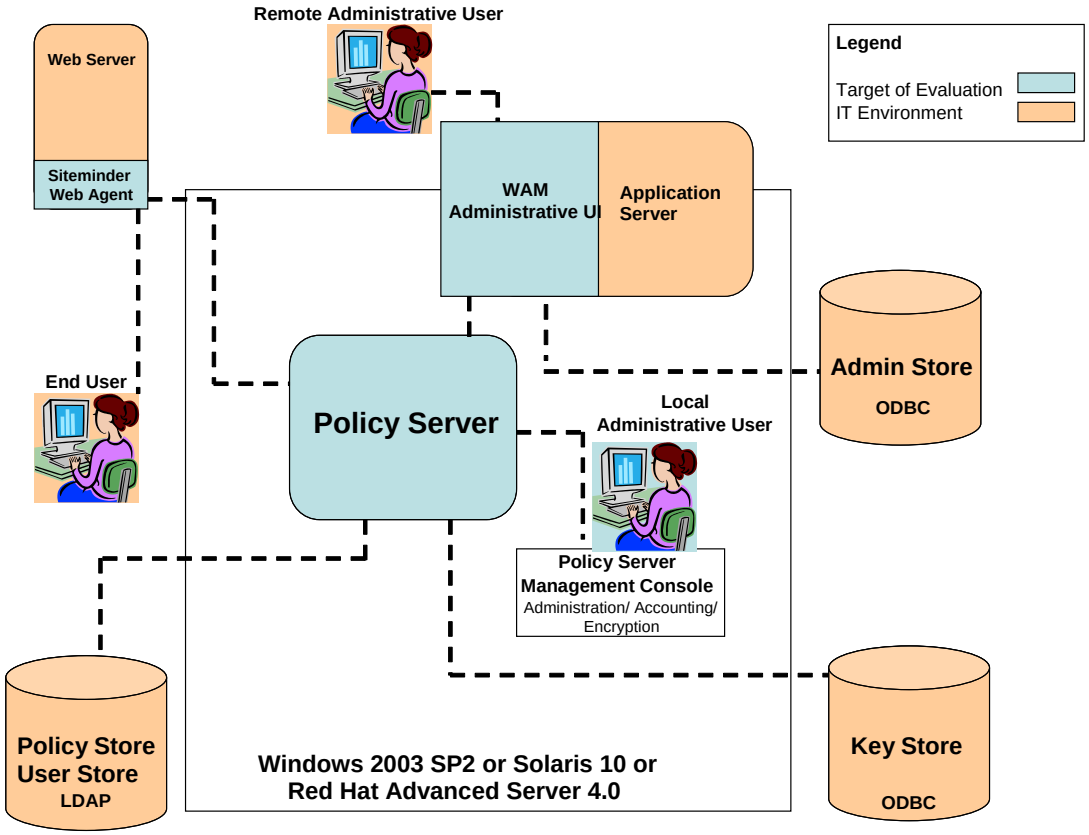
TOE	SiteMinder Web Access Manager r12 SP1-CR3	Software package installed includes all TOE items listed below: Policy Server Web Agent WAM Admin UI
Environment	Windows Server 2003 SP2, Red Hat AS 4.0, or Solaris 10	SiteMinder with Operating System
Environment	Oracle Database	Oracle Database, version 10g r2
Environment	Web Server	IIS 6.0, ASF Apache 2.2, or SunOne 6.1 SP2
Environment	LDAP Server	Active Directory Windows 2003 or iPlanet 5.2
Environment	Application Server	Jboss 4.0.5 or WebLogic 9.2

7. Architectural Information

The Policy Server and Web Agent provide partial protection of TSF data. The TOE maintains and controls individual sessions for Administrators and End Users. The TSF, when invoked by the underlying host OS, ensures that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed. The TSF maintains a security domain for its own execution that protects it from interference and tampering by untrusted subjects initiating actions through its own TSFI.

7.1. Logical Boundary

Figure 2 – TOE Logical Boundary



The logical boundary of the TOE includes the SiteMinder Web Access Manager r12 SP1-CR3 Policy Server, Web Agent, and WAM Administrative UI software. These components are described in the following subsections:

7.1.1. Policy Server

The Policy Server provides access control and single sign-on. It allows an authorized administrator to configure policies via the Siteminder WAM Administrative UI. It also allows administrators to override the default values provided for the TOE. The policies govern the level of access to a resource granted to an end user. It typically runs on a separate Windows or UNIX system and offers the following security operations:

- Authentication — Authenticates end users via a range of authentication methods including usernames and passwords, and public key certificates.
- Authorization — Manages and enforces access control rules established by administrators.
- Administration — Enables the Siteminder WAM Administrative UI to record configuration information in the Policy Store.
- Accounting service — Generates log files that contain auditing information.

- Trusted path — Uses an encrypted tunnel and encrypted keys to pass sensitive data between separate parts of the TOE.

7.1.2. Web Agent

A SiteMinder Web Agent is a software component that controls end user access to a protected resource (any URL protected by the TOE). The Web Agent grants or denies access by enforcing policies defined through the Policy Server. Web Agents work with the Policy Server to authenticate and authorize end users for access to web server resources. The Web Agent enables Web applications to personalize content. The network path between the Web Agent and the Policy Server is secured by AES encryption over a standard TCP/IP connection. The Web Agent is integrated with a Web server. The Web Agent intercepts requests for a resource and determines whether or not the resource is protected by the TOE.

Web Agents perform the following tasks:

- Intercept access requests for protected resources and work with the Policy Server to determine whether or not an end user should have access.
- Provide information to a Web application that dictates how content is presented to the end user (policy-based personalization) and how to deliver access privileges.
- Ensure an end user's ability to securely access information. Web Agents store contextual information about end user access privileges in a session cache. Performance can be optimized by modifying the cache settings.
- Enable single sign-on across web servers in a single cookie domain or across multiple cookie domains without requiring end users to re-authenticate.

7.1.3. WAM Administrative UI

The WAM Administrative UI lets administrators view, modify, and delete Policy Server objects. All of the capabilities discussed in the Policy Server section above are implemented through use of the WAM Administrative UI. Although the details of each task differ by object, the general methods are similar. For example, the procedure for deleting an Agent is similar to the procedure for deleting a response. Policy Server objects include but are not limited to end users, policies, rules, realms and agents. The ability for administrators to perform actions on Policy Server objects is what allows them to define what resources are protected by the TOE from end user access.

8. Documentation

The following publicly available documents were evaluated per assurance requirement.

Table 8 – Assurance Documents Evidence

Component	Document(s)	Rationale
AGD_OPE.1 Operational User Guidance	**1. CA SiteMinder Web Access Manager R12 SP1-CR3 Admin Supplemental Guidance, 1.1, May 29 2009.** **2. SM r12 SP1 Configuration Guide, December 2008** **3. CA SiteMinder Web Access Manager Policy Server Administration Guide r12** **4. CA SiteMinder Web Access Manager Policy Server Configuration Guide r12** **5. CA SiteMinder Web Access Manager Web Agent Configuration Guide r12**	These documents describe the operational user guidance for CA SiteMinder r12 SP1-CR3.
AGD_PRE.1 Preparative Procedures	**Evaluated Configuration for CA Siteminder Web Access Manager R12 SP1, March 2009**	This document describes the preparative procedures that need to be done prior to installing CA SiteMinder r12 SP1-CR3.
ALC_DEL.1 Delivery Procedures	**SiteMinder r12 SP1 – NIAP Download/Installation Instruction, October 2008**	This document describes product delivery for CA SiteMinder r12 SP1-CR3 and a description of all procedures used to ensure objectives are not compromised in the delivery process.

9. TOE Acquisition

The NIAP-certified SiteMinder product is acquired via normal sales channels, and digital delivery of the TOE is coordinated with the end customer by CA.

10. IT Product Testing

The test team's test approach is to test the security mechanisms of the CA Siteminder Web Access Manager by exercising the external interfaces to the TOE and viewing the TOE behavior either remotely, or on the platform. Each TOE external interface is described in the appropriate design documentation (e.g., FSP) in terms of the relevant claims on the TOE that can be tested through the external interface. The ST, TOE Design Specification (TDS), Architecture (ARC), Functional Specification (FSP), and the vendor's test plans were used to demonstrate test coverage of all appropriate EAL3 requirements for all security relevant TOE external interfaces. TOE external interfaces that were determined to be security relevant are interfaces that

- change the security state of the product,
- permit an object access or information flow that is regulated by the security policy,

- are restricted to subjects with privilege or behave differently when executed by subjects with privilege, or
- invoke or configure a security mechanism.

Security functional requirements were determined to be appropriate to a particular interface if the behavior of the TOE that supported the requirement could be invoked or observed through that interface.

The evaluation team created a test plan that contained a sample of the vendor functional test suite, and supplemental functional testing of the vendors' tests. Booz Allen also performed vulnerability assessment and penetration testing.

10.1. VULNERABILITY TESTING

The evaluation team executed the following vulnerability tests against CA Siteminder Web Access Manager R12 SP1-CR3:

- Eavesdropping on Communications
  [arpspoof (version 2.4), wireshark (version 1.0)]
  In this test, the evaluators manually inspected network traffic to and from the TOE in order to ensure that no useful or confidential information could be obtained by a malicious user on the network.
- Port Scanning
  [nmap (version 4.60)]
  Remote access to the TOE should be limited to the standard TOE interfaces and procedures. This test attempted to find ways to bypass these standard interfaces of the TOE and open any other vectors of attack.
- Buffer Overflow / Format String / Unexpected Input Attack
  [CIRT.dk fuzzer (version 1.0)]
  In this attack, the evaluators attempted to discover and exploit any software errors that do not appropriately handle various non standard inputs. The evaluators attempted to inject known malicious inputs into the various TOE interfaces. These malicious inputs form 3 categories.
  - Buffer Overflows: In this case, larger and larger inputs are injected to try to overflow a buffer and corrupt the program stack.
  - Format Strings: In this case, format strings are injected to attempt to see if they are not handled correctly by the program.
  - Special Characters: In this case, unexpected special characters are injected in an attempt to induce non standard behavior.
- ICMP Blind Connection Reset
  [icmp-reset (version 1.0)]
  This test attempted to exploit a known vulnerability using ICMP connection reset packets. If effective, this test would prevent the normal functionality of the TOE and invoke a denial of service against it.
- Generic Vulnerability Scanner
  [Nessus (version 3.2.1.1)]

This test used the Nessus Vulnerability scanner to test any and all open interfaces on any applicable systems of the TOE. The scanner probes a wide range of vulnerabilities that include the following:

Backdoors	Gain root remotely	RPC
CGI abuses	General	Settings
Denial of Service	Miscellaneous	SMTP Problems
Finger abuses	Netware	SNMP
Firewalls	NIS	Untested
FTP	Port scanners	Useless services
Gain shell remotely	Remote file access	

- Unauthenticated Access / Directory Traversal Attack
  [manual browser methods]
  This test used "URL hacking" to attempt to access protected TOE resources by injecting unexpected input into requests that were sent to the TOE. This was done using two different approaches to URL exploitation.
  - The first part attempted to access protected TOE resources as an unauthenticated outsider.
  - The second part attempted to access local TOE resources that should be protected from any remote access (unauthenticated and authenticated).
- SQL Injection / Cross Site Scripting Attack
  [Paros (version 3.2.13)]
  This test executed automated SQL Injection and Cross Site Scripting attacks against the TOE. The evaluators determined any fields or variables that could be prone to attack. They then used a scanner, which contained a large database of standard strings that are used for testing SQL Injection and Cross Site Scripting issues. These strings were input into the various fields and variables and the output was analyzed for inconsistencies.
- Direct Database Access Attack
  [tnscmd (revision 1.3)]
  The TOE uses a database to store all of its security related data. The way it is designed, the TOE should perform all direct interaction to and from the backend database and no user should have any access to it. This test attempted to access the database directly and bypass these normal access procedures.
- Custom Buffer Overflow / Denial of Service
  [custom perl (version 5.10.0) and shell scripts (using netcat version 0.7.1)]
  These tests attempted to exploit specific Buffer Overflow vulnerabilities that were found via public resources. A successful exploit would have resulted in a denial of service attack by crashing the server, or it could have resulted in remote code injection.
- Remote Code Injection
  This test attempted to use a known vulnerability in an application server that is used by the TOE to inject arbitrary code to be executed.
- Documented Server Vulnerabilities
  [custom perl (version 5.10.0) and shell scripts (using netcat version 0.7.1)]

This test attempted to exploit publicly known vulnerabilities that potentially exist in the web servers of the product.

- Web Server Vulnerability Scanner
  [Nikto (version 2.02)]
  This test used the Nikto web server vulnerability scanner to test for any known vulnerabilities that could be present in the TOE's web interfaces. This scanner probed a wide range of vulnerabilities that included the following:

File Upload. Interesting File / Seen in logs. Misconfiguration / Default File. Information Disclosure. Injection (XSS/Script/HTML). Remote File Retrieval	Denial of Service. Command Execution / Remote Shell. SQL Injection. Authentication Bypass. Software Identification Remote source inclusion.

11. Results of the Evaluation

The Evaluation Team conducted the evaluation in accordance with the CC and the CEM. A Pass, Fail, or Inconclusive verdict was assigned to each work unit of assurance component. For Fail or Inconclusive work unit verdicts, the Evaluation Team advised the developer of the issue that needed to be resolved or the clarification that needed to be made to the particular evaluation evidence.

The Evaluation Team accomplished this by providing notes, comments, or vendor actions in the draft ETR sections for an evaluation activity (e.g., ASE, ADV) that recorded the Evaluation Team's evaluation results and that the Evaluation Team provided to the developer. The Evaluation Team also communicated with the developer by telephone and electronic mail. If applicable, the Evaluation Team re-performed the work unit or units affected. In this way, the Evaluation Team assigned an overall Pass verdict to the assurance component only when all of the work units for that component had been assigned a Pass verdict.

The criteria and evaluation methodology against which the CA SiteMinder Web Access Manager r12 SP1-CR3 TOE was judged are described in the CC and CEM, Version 3.1 revision 2, dated September 2007. The Booz Allen Hamilton CCTL determined that the evaluation assurance level (EAL) for the CA SiteMinder Web Access Manager r12 SP1-CR3 TOE is EAL 3 and that the TOE, configured as specified in the installation guide, satisfies all of the security functional requirements stated in the Security Target. The results of the evaluation and the rationale supporting each CEM work unit verdict are recorded in the *CA Siteminder Web Access Manager R12 SP1 Evaluation Technical Report* which is considered proprietary.

12. Validator Comments/Recommendations

The following vulnerabilities were discovered in the vulnerability analysis process and not mitigated. They are considered acceptable residual vulnerabilities because they require a higher attack potential to exploit than is required for EAL3 or because successfully executing them would require a violation of the assumptions defined by the ST for the TOE's intended environment. For all other vulnerabilities discovered during the vulnerability analysis process, either the evaluated version of the TOE is resistant to them or they can be mitigated by ensuring that the TOE systems are diligently patched. For specific patches and fixes that mitigate vulnerabilities which were discovered, refer to the supplemental administrative guidance. This guidance must be followed in order to ensure a secure configuration.

CVE-2008-2168 - Cross-site scripting (XSS) vulnerability in Apache 2.2.6 and earlier allows remote attackers to inject arbitrary web script or HTML via UTF-7 encoded URLs that are not properly handled when displaying the 403 Forbidden error page.

CVE-2008-0455, CVE-2008-0456 - Cross-site scripting (XSS) vulnerability in the mod_negotiation module in the Apache HTTP Server 2.2.6 and earlier in the 2.2.x series, 2.0.61 and earlier in the 2.0.x series, and 1.3.39 and earlier in the 1.3.x series allows remote authenticated users to inject arbitrary web script or HTML by uploading a file with a name containing XSS sequences and a file extension, which leads to injection within a (1) "406 Not Acceptable" or (2) "300 Multiple Choices" HTTP response when the extension is omitted in a request for the file.

CVE-2007-6203 - Apache HTTP Server 2.0.x and 2.2.x does not sanitize the HTTP Method specifier header from an HTTP request when it is reflected back in a "413 Request Entity Too Large" error message, which might allow cross-site scripting (XSS) style attacks using web client components that can send arbitrary headers in requests, as demonstrated via an HTTP request containing an invalid Content-length value, a similar issue to CVE-2006-3918.

CVE-2007-4465 - Cross-site scripting (XSS) vulnerability in mod_autoindex.c in the Apache HTTP Server before 2.2.6, when the charset on a server-generated page is not defined, allows remote attackers to inject arbitrary web script or HTML via the P parameter using the UTF-7 charset. NOTE: it could be argued that this issue is due to a design limitation of browsers that attempt to perform automatic content type detection.

CVE-2005-2089 - Microsoft IIS 5.0 and 6.0 allows remote attackers to poison the web cache, bypass web application firewall protection, and conduct XSS attacks via an HTTP request with both a "Transfer-Encoding: chunked" header and a Content-Length header, which causes IIS to incorrectly handle and forward the body of the request in a way that causes the receiving server to process it as a separate HTTP request, aka "HTTP Request Smuggling."

CVE-2007-1157 - Cross-site request forgery (CSRF) vulnerability in jmx-console/HtmlAdaptor in JBoss allows remote attackers to perform privileged actions as administrators via certain MBean operations, a different vulnerability than CVE-2006-3733.

CVE-2007-5923 - Cross-site scripting (XSS) vulnerability in forms/smpwservices.fcc in CA (formerly Computer Associates) eTrust SiteMinder Agent allows remote attackers to inject

arbitrary web script or HTML via the SMAUTHREASON parameter, a different vector than CVE-2005-2204.

CVE-2003-1312 - siteminderagent/SmMakeCookie.ccc in Netegrity SiteMinder places a session ID string in the value of the SMSESSION parameter in a URL, which might allow remote attackers to obtain the ID by sniffing, reading Referrer logs, or other methods.

The following easily exploitable vulnerabilities can be mitigated by appropriate patches as discussed in the supplemental administrative guidance:

CVE-2007-1354 - The Access Control functionality (JMXOpsAccessControlFilter) in JMX Console in JBoss Application Server 4.0.2 and 4.0.5 before 20070416 uses a member variable to store the roles of the current user, which allows remote authenticated administrators to trigger a race condition and gain privileges by logging in during a session by a more privileged administrator, as demonstrated by privilege escalation from Read Mode to Write Mode.

CVE-2008-2518 - Cross-site scripting (XSS) vulnerability in the advanced search mechanism (webapps/search/advanced.jsp) in Sun Java System Web Server 6.1 before SP9 and 7.0 before Update 3 allows remote attackers to inject arbitrary web script or HTML via unspecified vectors, probably related to the next parameter.

CVE-2008-2166 - Cross-site scripting (XSS) vulnerability in the search module in Sun Java System Web Server 6.1 before SP9 and 7.0 before Update 2 allows remote attackers to inject arbitrary web script or HTML via unknown parameters in index.jsp.

CVE-2007-6572 - Cross-site scripting (XSS) vulnerability in Sun Java System Web Server 6.1 before SP8 and 7.0 before Update 1 allows remote attackers to inject arbitrary web script or HTML via unspecified vectors, aka BugID 6566204.

CVE-2007-1526 - Sun Java System Web Server 6.1 before 20070314 allows remote authenticated users with revoked client certificates to bypass the Certificate Revocation List (CRL) authorization control and access secure web server instances running under an account different from that used for the admin server via unspecified vectors.

CVE-2006-2501 - Cross-site scripting (XSS) vulnerability in Sun ONE Web Server 6.0 SP9 and earlier, Java System Web Server 6.1 SP4 and earlier, Sun ONE Application Server 7 Platform and Standard Edition Update 6 and earlier, and Java System Application Server 7 2004Q2 Standard and Enterprise Edition Update 2 and earlier, allows remote attackers to inject arbitrary web script or HTML via unknown attack vectors, possibly involving error messages.

CVE-2008-1446 - Integer overflow in the Internet Printing Protocol (IPP) ISAPI extension in Microsoft Internet Information Services (IIS) 5.0 through 7.0 on Windows 2000 SP4, XP SP2 and SP3, Server 2003 SP1 and SP2, and Server 2008 allows remote authenticated users to execute arbitrary code via an HTTP POST request that triggers an outbound IPP connection from a web server to a machine operated by the attacker, aka "Integer Overflow in IPP Service Vulnerability."

CVE-2008-0075 - Unspecified vulnerability in Microsoft Internet Information Services (IIS) 5.1 through 6.0 allows remote attackers to execute arbitrary code via crafted inputs to ASP pages.

CVE-2003-0718 - The WebDAV Message Handler for Internet Information Services (IIS) 5.0, 5.1, and 6.0 allows remote attackers to cause a denial of service (memory and CPU exhaustion, application crash) via a PROPFIND request with an XML message containing XML elements with a large number of attributes.

13. Annexes

Not applicable.

14. Security Target

The security target for this product's evaluation is CA SiteMinder Web Access Manager r12 SP1-CR3, Version 0.8, 2009-05-29

15. List of Acronyms

Acronym	Definition
ACM	Configuration Management
ADO	Delivery and Operation
ADV	Development
AGD	Guidance Documents
ALC	Life cycle support
ATE	Tests
AVA	Vulnerability assessment
CC	Common Criteria [for IT Security Evaluation]
EAL	Evaluation Assurance Level
FAU	Security Audit
FCO	Communication
FCS	Cryptographic Support
FDP	User Data Protection
FIA	Identification and Authentication
FMT	Security Management
FPT	Protection of the TSF
FTA	TOE Access
FTP	Trusted Channels/Path
GUI	Graphical User Interface
ICMP	Internet Control Message Protocol
ID	Identifier
IP	Internet Protocol
IT	Information Technology
SF	Security Function
SFP	Security Function Policy
ST	Security Target
TCP	Transmission Control Protocol
TOE	Target of Evaluation

Acronym	Definition
TSC	TSF Scope of Control
TSF	TOE Security Functions
TSFI	TOE Security Functions Interface
TSP	TOE Security Policy

16. Terminology

Term	Definition
Administrator	A trusted user who has privileges to administer the TOE. The privileges and activities of an administrator account vary by administrative scope and tasks. Types of administrators include domain, system, and super user admins.
Administrator Policy	A policy that sets restrictions on an administrator's ability to manage the TOE.
Agent	An Agent is installed on Web servers, or application servers to secure access to resources.
Agent Group	An Agent group is a Policy Server object that points to a group of Agents. The Agents in the group can be installed on different servers, but all of the Agents protect the same resources. Typically Agent groups are configured in SiteMinder for groups of servers that distribute the workload for access to a popular set of resources.
Agent Configuration Object	An Agent Configuration Object holds configuration parameters for one or more Web Agents.
Agent Key	Used by Web Agents to encrypt cookies.
Authentication Level	Each authentication method is associated with a particular level, ranging from a top priority of 1 to a lowest priority of 1000. End users that authenticate with a low level must re-authenticate when trying to access a resource with a higher authentication level.
Authentication Scheme	An authentication scheme is a Policy Server object that determines the credentials an end user will need to access a protected resource. Authentication schemes are assigned to realms. When an end user tries to access a resource in a realm, the authentication scheme of the realm determines the credentials that an end user must supply in order to access the resource.
Authorized	An administrator or workstation end user that has been identified and authenticated by the TOE.
Cluster	A set of Policy Servers that are grouped to improve system availability and response time by dynamically balancing load among the servers in the cluster and failing over to other clusters based on customer-defined failover thresholds. Clusters are typically grouped by data centers located in different geographic locations.
Cluster failover	Switching to another cluster when the number of servers available in a cluster falls below a configurable threshold. The priority of the clusters is defined in the Host Configuration Object. Requests will fail-back to a higher-priority cluster as soon as the threshold requirement for that cluster is met.
End User	An authorized user of the TOE without privileges who tries to gain access to a protected resource.
Get/Put/Post	An HTTP operation known as an end user's request. It is received by the Web Agent and forwarded to the Policy Server.
Global Objects	Objects that apply to all resources (global rules, global responses, global policies).
Global Rules	A global rule is a Policy Server object that specifies a filter used to apply a global policy to a large group of resources.
Global Responses	A global response is a Policy Server object that determines a reaction to a global rule.

Term	Definition
	Global responses are included in global policies, and take place when a global rule is triggered.
Global Policies	A global policy is a Policy Server object that binds end users, global rules, global responses, and optionally, time restrictions and IP address restrictions together.
Groups	A group (agent group, rule group, response group) can contain individual items or groups of its own type. For example, a rule group can contain rules and/or groups of rules.
Host Configuration Objects	A Host Configuration Object holds configuration parameters for the Trusted host.
Key Store	Entity used by the SiteMinder Policy Server to store encryption keys used by the Policy Server when communicating with SiteMinder Web Agents.
Nested Groups	Groups that contain other groups. Also known as sub-groups. When nested groups are allowed, each user group and each sub-group is searched when the policy is processed. When they are not allowed, each user group is searched, but sub-groups are not searched, when the policy is processed.
Nested Realms	Realms created within other realms to better represent the grouping of resources in a corporate network. Deeper levels of nested realms typically correspond to heightened security requirements in a directory tree. An administrator can achieve this by assigning a stronger authentication scheme with a higher protection level to the nested realm.
Policy	A policy is a Policy Server object that binds users, rules, responses, and optionally, time restrictions and IP address restrictions together. Policies establish entitlements for a SiteMinder protected entity. When a user attempts to access a resource, the policy is what SiteMinder ultimately uses to resolve the request.
Policy Domains	A policy domain is a logical grouping of one or more user directories, administrators, and realms. This Policy Server object is the basis for entitlement data. By creating policy domains, an administrator creates a container for entitlements that surround a particular group of resources (realm), as well as the end users who may access the resources, and the administrator who sets up entitlements.
Policy Domain Object	A SiteMinder object within a domain (policy, realm, response, response groups, response attributes, rules and rule groups, rule policies).
Policy Server	CA SiteMinder software component that provides a platform for managed key operations, authentication, authorization, and security management.
Policy Server Object	An object that the Policy Server uses (System objects, Policy Domain objects, Global objects).
Policy Store	Collection of CA SiteMinder Policy Server objects. Policy stores can reside in an ODBC (see page 19)-enabled database or an LDAP (see page 17) directory.
Policy Store Key	A key used to encrypt data that is sent between the Policy Server and the Policy Store. The key can be from 6 to 24 characters in length. All Policy Servers that share a SiteMinder Policy Store (a database containing policy information) must be configured using the same Policy Store Key.
Protected Resource	Any URL under SiteMinder protection.
Protection Level	A number between 0 and 1000 that is given to authentication schemes. A higher number indicates a higher level of protection.
Realm	A realm is a Policy Server object that identifies a group of resources. Realms typically define a directory or folder and possibly its subdirectories.

Term	Definition
Realm Resource Filter	A string, such as a relative path to a directory that specifies the resources covered by the realm. If the realm is a top-level realm, specify the resources relative to the server that serves up the files or application. If the realm is nested, specify the resources relative to the parent realm.
Remote Server	The workstation used by the end user to gain access to the TOE.
Resource	Any URL to which an end user attempts to gain access.
Response	A response is a Policy Server object that determines a reaction to a rule. Responses are included in policies, and take place when a rule is triggered.
Response Groups	A response group is a Policy Server object that contains a logical grouping of responses. Response groups are most often used when many responses will be included in a policy.
Rules	A rule is a Policy Server object that identifies a resource and the actions that will be allowed or denied for the resource. Rules can also include actions associated with specific events, such as what to do if an end user fails to authenticate correctly when asked for their credentials.
Rule Groups	A rule group is a Policy Server object that contains multiple rules. Rule groups are used to tie together different rules that will be used in a single policy.
Rule Resource	A string or regular expression that specifies the resources to which the rule applies. Specify the resources relative to the realm containing the resource.
Scope	Indicates whether the administrator's privileges extend to all domains and applications or to only specific domains and applications. Included in the Administrator Policy.
Session Key	The Policy Server creates Session Keys using AES with HMAC-SHA256. The Session Keys are utilized by the Policy Server and SiteMinder agents for protecting the TCP/IP message exchange between these components.
Session Ticket	Also known as session specification. Session tickets contain credentials and other information relating to an end user's session.
Session Ticket Key	The Policy Server utilizes the Session Ticket Key to encrypt Password Services data in the user stores, and sensitive data (keys, shared secrets, passwords) in the Policy Stores.
Super User Administrator	The default administrator account with full privileges that is set up during installation of the TOE. There are two Super User accounts created during the installation of the TOE. A local Super User account which is used during installation and configuration and a remote Super User account which is used in the evaluated configuration of the TOE. The remote Super User is the administrator that can access the TOE via the WAM Administrative UI.
System Objects	Objects used throughout a SiteMinder deployment (agents, agent groups, agent configuration objects, host configuration objects, user directories, policy domains, administrators, authentication schemes, registration schemes, agent types, password policies, trusted hosts).
Target network	The domain of workstations that have the TOE installed on them.
Task	Determines the privileges an administrator is allowed to perform within their scope. Included in the Administrator Policy.
Time Restrictions	A time restriction indicates when a rule fires. For example, if an administrator creates an Allow Access rule with a time restriction that limits access to a resource to 9am - 5 pm, Monday - Friday, the rule will only fire and allow end users to access the resource during the specified time. The resource will not be available outside the times indicated.

Term	Definition
Trusted Hosts	A Trusted Host object represents the client component that connects to the Policy Server.
User	Defined as an administrator, domain administrator, system administrator, or end user of the TOE.
User Authorization Cache (memory)	A configurable cache inside the Policy Server that stores user information after the login step.
User Directories/ User Store	A user directory in SiteMinder is an object that contains details for connecting to an existing user directory that resides outside of SiteMinder. This allows an administrator to configure a simple connection to an existing user directory, instead of replicating end user information within SiteMinder. The username space is an LDAP directory server.

17. Bibliography

1. Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, Version 3.1r2.
2. Common Criteria for Information Technology Security Evaluation – Part 2: Security functional requirements, Version 3.1r2.
3. Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance requirements, Version 3.1r2.
4. Common Evaluation Methodology for Information Technology Security Evaluation, Version 3.1r2
5. CA Siteminder Web Access Manager R12 SP1-CR3 Security Target, Version 0.8, 2009-05-29
6. CA Siteminder Web Access Manager R12 SP1 Evaluation Technical Report (ETR), Version 1.2, 2009-06-09