



WEBADE

Security Configuration Definitions

Client: BC Provincial Government
Date: Dec 01, 2008
Revision: 1

Vivid Solutions Inc.
Suite #1A, 2328 Government St.
Victoria, BC V8T 5G5
Phone: (250) 385-6040
Fax: (250) 385-6046
Website: www.vividsolutions.com

Document Change Control

REVISION NUMBER	DATE OF ISSUE	AUTHOR(S)	DESCRIPTION
1	June 22, 2007	Jason Ross	Original draft
2	Dec 01, 2008	Paul Hill	Release Update

Table of Contents

1. INTRODUCTION.....	4
2. WEBSITE IS COMPLETELY SITEMINDER PROTECTED.....	5
2.1 SITEMINDER CONFIGURATION	5
2.2 WEBADE EXPECTATIONS.....	5
2.3 ERROR CONDITIONS.....	6
2.4 CONFIGURATION	6
3. TWO-URL CONFIGURATION FOR AUTHENTICATED AND ANONYMOUS ACCESS.....	7
3.1 SITEMINDER CONFIGURATION	7
3.2 WEBADE EXPECTATIONS.....	7
3.3 ERROR CONDITIONS.....	8
3.4 CONFIGURATION	9
4. PARTIALLY-SECURED WEB APPLCIATION	10
4.1 SITEMINDER CONFIGURATION	10
4.2 WEBADE EXPECTATIONS.....	10
4.3 ERROR CONDITIONS.....	11
4.4 CONFIGURATION	11
5. WEBSITE WITH SITEMINDER LOG-IN ONLY	13
5.1 SITEMINDER CONFIGURATION	13
5.2 WEBADE EXPECTATIONS.....	13
5.3 ERROR CONDITIONS.....	14
6. OTHER CONFIGURATIONS	15
6.1 COMPLETELY UNSECURED WEBSITE	15
6.2 WEB SERVICES	15
6.3 SINGLE-SIGN-ON AND MULTIPLE APPLICATIONS.....	15
7. OUTSTANDING ANALYSIS.....	17

1. INTRODUCTION

The WebADE User API provides user information and authentication services to WebADE applications.

WebADE applications can require one of several different security configurations. This document describes each supported security configuration, specifying what information and behaviour WebADE expects from the container's security architecture and how WebADE responds to changes in the information passed to it from the container.

It is important to describe not only what information WebADE requires to perform properly, but to describe the situations that can arise that would cause an error in the WebADE API due to improper or unexpected behaviour on the container's part.

2. WEBSITE IS COMPLETELY SITEMINDER PROTECTED

This configuration describes a web application that has SiteMinder providing authentication services for all requests to the application. When a user accesses any part of the web application for the first time, they will be redirected to the BC Government Common Log-in Page and are forced to log-in before accessing the application directly. A user will be logged in to SiteMinder for the duration of their session with the web application.

In this configuration setup, no anonymous users can access the application.

2.1 SITEMINDER CONFIGURATION

For this configuration setup, SiteMinder must be setup to secure all requests to the target web application. SiteMinder must not allow unsecured requests through to the web application. As this is the only avenue for accessing the application, SiteMinder will ensure all requests (SM headers and encrypted cookies) are valid. SiteMinder's session timeout must also be set to be equal to or greater than the session timeout of the web application it is securing.

2.2 WEBADE EXPECTATIONS

For all requests to a WebADE application, WebADE expects the following request headers to be set by SiteMinder before the container hands the request to WebADE to process:

REQUEST HEADERS	EXPECTED VALUES	DESCRIPTION
SM_UNIVERSALID	<i>User-account-name</i>	The logged-in user's account name (not including the domain, such as IDIR, BCEID, or MYID).
SMGOV_USERTYPE	Internal, Business, or Individual	The user type code indicating whether the user is an IDIR (Internal), BCEID(Business), or MYID(Individual) domain user.
SMGOV_USERGUID	A 32 character length Hexadecimal number.	A generated ID that is unique across domains.

2.3 ERROR CONDITIONS

WebADE interprets the following situations as session-level errors and will prevent the request from being processed by the web application.

NOTE: Other requests to the web application will not be affected by these error conditions, unless the same errors conditions are present for those requests as well. In other words, WebADE will resume processing requests once the error conditions have been resolved.

ERROR CONDITION	DESCRIPTION
Missing Request Headers	<p>If a request is received by the WebADE with all of the SiteMinder request headers missing, WebADE will interpret this as a processing error. WebADE will prevent the request from being processed and will present the user with an error page.</p> <p>The most likely reason for this situation is a misconfiguration of the WebADE. This is probably due to the web application actually being setup as a two-URL-configuration allowing anonymous user access or a partially secured website, but the WebADE has been configured to always expect authenticated users.</p> <p>Updating the WebADE security configuration to the correct configuration should fix the issue.</p>
Partial Request Headers	<p>If a request is received by the WebADE with some of the request headers missing or set with invalid values, WebADE will interpret this as a processing error. WebADE will prevent the request from being processed and will present the user with an error page.</p> <p>This error is different from the previous one, in that the error most likely lies with SiteMinder. In this situation, the user has been authenticated by SiteMinder, but SiteMinder is not setting all of the required headers properly, or is setting the headers with improper values (such as a SMGOV_USERTYPE value not matching one of those specified above).</p> <p>This can be resolved by reading the WebADE error message on the error page and in the logs, and working with CITS to identify and fix the SiteMinder issue causing the problem.</p>
Conflicting Request Headers	<p>security configurations), update the WebADE security configuration to match the proper configuration.</p>

2.4 CONFIGURATION

Preference setting required for this configuration:

SUB_TYPE	NAME	VALUE
webade-security-provider	webade-security-provider-type	authenticated-only-siteminder

3. TWO-URL CONFIGURATION FOR AUTHENTICATED AND ANONYMOUS ACCESS

This configuration describes a web application that has SiteMinder managing security for all requests to the application via one URL, while a second URL is configured without SiteMinder security to allow anonymous access. When a user accesses any part of the web application via the SiteMinder-controlled URL for the first time, they will be redirected to the BC Government Common Log-in Page and are forced to log-in before accessing the application directly. A user will be logged in to SiteMinder for the duration of their session with the web application.

3.1 SITEMINDER CONFIGURATION

For this configuration setup, SiteMinder must be setup to secure all requests to the target web application via the "secured" URL. SiteMinder must not allow unsecured requests through to the web application via this URL. SiteMinder's session timeout must also be set to be equal to or greater than the session timeout of the web application it is securing.

For the anonymous URL, no SiteMinder configuration should be configured, allowing anonymous users to access the application.

3.2 WEBADE EXPECTATIONS

For all requests to a WebADE application via the secured URL, WebADE expects the following request headers to be set by SiteMinder before the container hands the request to WebADE to process:

REQUEST HEADERS	EXPECTED VALUES	DESCRIPTION
SM_UNIVERSALID	<i>User-account-name</i>	The logged-in user's account name (not including the domain, such as IDIR, BCEID, or MYID).
SMGOV_USERTYPE	Internal, Business, or Individual	The user type code indicating whether the user is an IDIR (Internal), BCEID(Business), or MYID(Individual) domain user.
SMGOV_USERGUID	A 32 character length Hexadecimal number.	A generated ID that is unique across domains.

For all requests to a WebADE application via the unsecured URL, WebADE expects no SiteMinder headers to be present for all requests through this URL. At this point the user is treated as Public and will be authorized to any actions granted to the "Public Users" group for the application.

A session in the web application will either be authenticated or unauthenticated. In this configuration, a session cannot switch between authenticated and unauthenticated access, as the user's web browser maintains separate sessions for each URL.

3.3 ERROR CONDITIONS

WebADE interprets the following situations as session-level errors and will prevent the request from being processed by the web application.

NOTE: Other requests to the web application will not be affected by these error conditions, unless the same errors conditions are present for those requests as well. In other words, WebADE will resume processing requests once the error conditions have been resolved.

ERROR CONDITION	DESCRIPTION
Partial Request Headers	<p>If a request is received by the WebADE with some of the request headers missing or set with invalid values, WebADE will interpret this as a processing error. WebADE will prevent the request from being processed and will present the user with an error page.</p> <p>This error most likely lies with SiteMinder. In this situation, the user has been authenticated by SiteMinder, but SiteMinder is not setting all of the required headers properly, or is setting the headers with improper values (such as a SMGOV_USERTYPE value not matching one of those specified above).</p> <p>This can be resolved by reading the WebADE error message on the error page and in the logs, and working with CITS to identify and fix the SiteMinder issue causing the problem.</p>
Conflicting Request Headers	<p>If a request is received by the WebADE with request headers with values that do not match those stored in the session, WebADE will interpret this as a processing error. WebADE will prevent the request from being processed and will present the user with an error page.</p> <p>This error is most likely due to a user's session with SiteMinder having timed out (or having been logged off manually) before the web application's session time out.</p> <p>If this is unintended, the SiteMinder timeout configuration should be changed to be equal to or greater than the web application session timeout.</p> <p>If this session timeout is intentional (the application is actually designed to use either the or website with SiteMinder log-in only security configurations), update the WebADE security configuration to match the proper configuration.</p> <p>NOTE: When faced with conflicting request headers, WebADE could simply invalidate the session and reload the user's information and permission from the new request headers. However, this has the potential to cause security or logic errors for the application, as WebADE would be taking the control of logging a user out of the application out of the application's hand. As it is unlikely, in the first place, that this particular session-managing behaviour is required by any application using this security configuration, it seems the safe choice to simply throw up an error page, preventing the user from continuing to access the application with this web session. This can always be changed in a later release, and it would be harder to put the genie back in the bottle, so to speak, after the fact.</p>

Unauthenticated/Authenticated Headers Switch	<p>If a request is received by the WebADE with either request headers set for an unauthenticated session, or with no headers set for an authenticated session, WebADE will interpret this as a processing error. WebADE will prevent the request from being processed and will present the user with an error page.</p> <p>This error should not occur in a two-URL security configuration, due to the user's web browser maintaining separate sessions for each URL. If this does occur, this should be resolved by reading the WebADE error message on the error page and in the logs, and working with CITS to identify and fix any SiteMinder issues that may be causing the problem.</p>
--	---

3.4 CONFIGURATION

Preference setting required for this configuration:

SUB_TYPE	NAME	VALUE
webade-security-provider	<i>webade-security-provider-type</i>	authenticated-and-unauthenticated-siteminder

4. PARTIALLY-SECURED WEB APPLICATION

This configuration describes a web application that has SiteMinder requiring authentication for some, but not all, requests to the application. Users accessing only the unsecured part of the application will initially be treated as unauthenticated users. When a user accesses the secured part of the web application for the first time, they will be redirected to the BC Government Common Log-in Page and are forced to log-in before accessing the secured part of the application directly. A user will then be logged in to SiteMinder for the duration of their session with the web application.

NOTE: Developers of applications that are to be deployed using this security configuration should be made aware that the user's information and permissions can change during the lifetime of the session, as a user moves from an unauthenticated user to authenticated. Applications using this security configuration must be able to handle this change in context.

4.1 SITEMINDER CONFIGURATION

For this configuration setup, SiteMinder is setup to secure all requests to a specific section of the target web application (usually a subdirectory in the application URL such as "/secure/"). SiteMinder must not allow unsecured requests through to this section of the web application, but should allow unauthenticated access to the rest of the application. SiteMinder's session timeout must also be set to be equal to or greater than the session timeout of the web application it is securing.

4.2 WEBADE EXPECTATIONS

For users accessing the unauthenticated section of the web application, WebADE will treat these users as anonymous users, until (and if) they access the secured section of the application, at which time their session credentials will be switched to match the headers returned by SiteMinder.

For all requests to the secured section of a WebADE application, WebADE expects the following request headers to be set by SiteMinder before the container hands the request to WebADE to process:

REQUEST HEADERS	EXPECTED VALUES	DESCRIPTION
SM_UNIVERSALID	<i>User-account-name</i>	The logged-in user's account name (not including the domain, such as IDIR, BCEID, or MYID).
SMGOV_USERTYPE	Internal, Business, or Individual	The user type code indicating whether the user is an IDIR (Internal), BCEID(Business), or MYID(Individual) domain user.
SMGOV_USERGUID	A 32 character length Hexadecimal number.	A generated ID that is unique across domains.

For all requests to the unsecured section of a WebADE application, WebADE expects none of the above SiteMinder headers to be present.

4.3 ERROR CONDITIONS

WebADE interprets the following situations as session-level errors and will prevent the request from being processed by the web application.

NOTE: Other requests to the web application will not be affected by these error conditions, unless the same errors conditions are present for those requests as well. In other words, WebADE will resume processing requests once the error conditions have been resolved.

ERROR CONDITION	DESCRIPTION
Partial Request Headers	<p>If a secured-section request is received by the WebADE with some of the request headers missing or set with invalid values, WebADE will interpret this as a processing error. WebADE will prevent the request from being processed and will present the user with an error page.</p> <p>This error most likely lies with SiteMinder. In this situation, the user has been authenticated by SiteMinder, but SiteMinder is not setting all of the required headers properly, or is setting the headers with improper values (such as a SMGOV_USERTYPE value not matching one of those specified above).</p> <p>This can be resolved by reading the WebADE error message on the error page and in the logs, and working with CITS to identify and fix the SiteMinder issue causing the problem.</p> <p>NOTE: The other possibility for this error is if the WebADE is not properly instructed by the application which sections of the application are secured and which are not secured by SiteMinder. In this configuration, the WebADE must be explicitly told when to expect SiteMinder headers and when to expect no headers, for security reasons. It is up to the application developer to make sure this is properly configured in the WebADE settings for their application.</p>
Conflicting Request Headers	<p>If a secured-section request is received by the WebADE with request headers with values that do not match those stored in the session of an authenticated user, WebADE will interpret this as a processing error. WebADE will prevent the request from being processed and will present the user with an error page.</p> <p>NOTE: In this situation, unauthenticated sessions will be switched to an authenticated session, instead of raising an error, as this is expected security behaviour for this configuration.</p> <p>This error is most likely due to a user's session with SiteMinder having timed out (or having been logged off manually) before the web application's session time out. The SiteMinder timeout configuration should be changed to be equal to or greater than the web application session timeout.</p>

4.4 CONFIGURATION

Preference setting required for this configuration:

SUB_TYPE	NAME	VALUE
----------	------	-------

webade-security-provider	<i>webade-security-provider-type</i>	partially-authenticated-siteminder
--------------------------	--------------------------------------	------------------------------------

5. WEBSITE WITH SITEMINDER LOG-IN ONLY

This configuration describes a web application that has SiteMinder managing only security for log-in requests to the application. Users accessing only the unsecured part of the application will initially be treated as unauthenticated users. When a user accesses the secured “log-in” part of the web application for the first time, they will be redirected to the BC Government Common Log-in Page and are forced to log-in before accessing the secured part of the application directly.

The difference between this configuration and the previous one is that a user will not be logged in to SiteMinder for the duration of their session with the web application. Instead, the user’s SiteMinder session will timeout after a short period of time (around 30 seconds to a minute).

5.1 SITEMINDER CONFIGURATION

For this configuration setup, SiteMinder is setup to secure all requests to a specific section of the target web application (usually a subdirectory in the application URL for log-in purposes only). SiteMinder must not allow unsecured requests through to this section of the web application, but should allow unauthenticated access to the rest of the application. SiteMinder will timeout its session for the user after a short period of time.

5.2 WEBADE EXPECTATIONS

For users accessing the unauthenticated section of the web application, WebADE will treat these users as anonymous users, until (and if) they access the secured section of the application, at which time their session credentials will be switched to match the headers returned by SiteMinder.

For all requests to the secured section of a WebADE application, WebADE expects the following request headers to be set by SiteMinder before the container hands the request to WebADE to process:

REQUEST HEADERS	EXPECTED VALUES	DESCRIPTION
SM_UNIVERSALID	<i>User-account-name</i>	The logged-in user’s account name (not including the domain, such as IDIR, BCEID, or MYID).
SMGOV_USERTYPE	Internal, Business, or Individual	The user type code indicating whether the user is an IDIR (Internal), BCEID(Business), or MYID(Individual) domain user.
SMGOV_USERGUID	A 32 character length Hexadecimal number.	A generated ID that is unique across domains.

For all requests to the unsecured section of a WebADE application, WebADE expects none of the above SiteMinder headers to be present.

5.3 ERROR CONDITIONS

WebADE interprets the following situations as session-level errors and will prevent the request from being processed by the web application.

NOTE: Other requests to the web application will not be affected by these error conditions, unless the same errors conditions are present for those requests as well. In other words, WebADE will resume processing requests once the error conditions have been resolved.

ERROR CONDITION	DESCRIPTION
Partial Request Headers	<p>If a secured-section request is received by the WebADE with some of the request headers missing or set with invalid values, WebADE will interpret this as a processing error. WebADE will prevent the request from being processed and will present the user with an error page.</p> <p>This error most likely lies with SiteMinder. In this situation, the user has been authenticated by SiteMinder, but SiteMinder is not setting all of the required headers properly, or is setting the headers with improper values (such as a SMGOV_USERTYPE value not matching one of those specified above).</p> <p>This can be resolved by reading the WebADE error message on the error page and in the logs, and working with CITS to identify and fix the SiteMinder issue causing the problem.</p>
Conflicting Request Headers	<p>If a secured-section request is received by the WebADE with request headers with values that do not match those stored in the session of an authenticated user, WebADE will interpret this as a processing error. WebADE will prevent the request from being processed and will present the user with an error page.</p> <p>NOTE: In this situation, unauthenticated sessions will be switched to an authenticated session, instead of raising an error, as this is expected security behaviour for this configuration.</p> <p>This error is most likely due to a user's session with SiteMinder having timed out (or having been logged off manually) and another user has logged in through SiteMinder before the web application's session time out.</p> <p>There is no automatic solution for this error, as it is up to the web application to invalidate the user's session when the user wishes to log out of the application. Modify the application code to prevent this issue.</p>

6. OTHER CONFIGURATIONS

The following configurations are possible, but require more discussion to determine if these are to be supported security configurations, and what sort of behaviour is to be expected for these configurations.

6.1 COMPLETELY UNSECURED WEBSITE

This configuration describes a web application that no managed security for all requests to the application. All access to this application is unauthenticated.

In this configuration setup, no authenticated users can access the application.

6.2 WEB SERVICES

This configuration describes a web service application that is managed security via basic authentication. Users will not generally directly access this type of application, but use some other application to make web service calls to the application, using either the user's credentials or a service account to authenticate.

For a web service application using IDIR service accounts for connections, SiteMinder may be used to handle the authentication. In this case, the security configuration should be similar in behaviour to the [completely SiteMinder-managed](#) configuration. See that configuration for expected behaviour and error cases.

For a web service application using user accounts for connections, SiteMinder may not be the optimum framework for handling user authentication, due to the poor way SiteMinder handles multiple domains (IDIR, BCEID, MYID, etc.) with basic authentication (it uses a domain-priority list, where it tries each domain in order to locate the user's ID and stops on the first occurrence, whether it is the user's account or not). In this case, another security framework, tied into the container via JAAS, may be used to authenticate the user. In this configuration, WebADE would only look for the "remote user" parameter in the request's session to determine the current user, ignoring any SiteMinder headers.

In this configuration setup, no unauthenticated users can access the application.

A SAML service would be the optimum solution for this situation

6.3 SINGLE-SIGN-ON AND MULTIPLE APPLICATIONS

Some applications will want to use SiteMinder's single-sign-on functionality to log in once, and then be able to access a number of applications. If all applications that are to share the single-sign-on are using the [completely SiteMinder-managed](#) configuration, this setup should work without any extra effort.

However, for situations where one or more of these application use the [partially secured website](#) configuration, some discussion is needed to address how to obtain the logged-in user credentials from SiteMinder. This is because the WebADE may

not be able to determine the logged in user as the SiteMinder headers are not set for application requests in the unsecured section of these applications.

7. OUTSTANDING ANALYSIS

We still need to determine what options are available, if any, for the WebADE code within a web application to communicate directly to SiteMinder to verify the user's credentials being passed in via the request headers. If there is a web service or some other sort of web call that we can use to directly confirm that the headers are valid, it would help to address a couple of the issues, relating to single-sign on across multiple apps and request header spoofing. This will probably involve some discussions with CITS to see what the options are.