IBM WebSEAL
Implementation Guide
(Version 5.7)

Copyright 2013
Deepnet Security Limited
Trademarks

Deepnet Unified Authentication, MobileID, QuickID, PocketID, SafeID, GridID, FlashID, SmartID, TypeSense, VoiceSense, MobilePass, DevicePass, RemotePass and Site Stamp are trademarks of Deepnet Security Limited. All other brand names and product names are trademarks or registered trademarks of their respective owners.

Copyrights

Under the international copyright law, neither the Deepnet Security software or documentation may be copied, reproduced, translated or reduced to any electronic medium or machine readable form, in whole or in part, without the prior written consent of Deepnet Security.

Licence Conditions

Please read your licence agreement with Deepnet carefully and make sure you understand the exact terms of usage. In particular, for which projects, on which platforms and at which sites, you are allowed to use the product. You are not allowed to make any modifications to the product. If you feel the need for any modifications, please contact Deepnet Security.

Disclaimer

This document is provided “as is” without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

This document could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the document. Deepnet Security may make improvements of and/or changes to the product described in this document at any time.

Contact

If you wish to obtain further information on this product or any other Deepnet Security products, you are always welcome to contact us.

Deepnet Security Limited
Comer Business Innovation Centres
North London Business Park
Oakleigh Road South
London N11 1GN
United Kingdom

Tel: +44(0)20 3668 1580
Fax: +44(0)20 8446 3182
Web: www.deepnetsecurity.com
Email: support@deepnetsecurity.com
# Table of Contents

1. Overview ...................................................................................................................... 4
2. Prerequisites ................................................................................................................. 4
3. Configuration ................................................................................................................ 5
   3.1 DualShield Configuration ......................................................................................... 5
      3.1.1 Logon Procedure ................................................................................................. 5
      3.1.2 Application .......................................................................................................... 5
      3.1.3 Service Provider ................................................................................................. 6
   3.2 WebSEAL Configuration ......................................................................................... 7
      3.2.1 Create Junctions ................................................................................................. 7
      3.2.2 Modify Configuration File .................................................................................. 8
      3.2.3 Edit Login File ..................................................................................................... 9
4. Authentication ............................................................................................................... 10
1. Overview

IBM Tivoli Access Manager (TAM) WebSEAL is a high-performance, multi-threaded Web server that applies fine-grained security policy to Access Manager’s protected Web object space. With the IBM TAM WebSEAL, e-business can leverage on the fine-grained access control and seamless web single sign-on to operate a multi-server, multi-domain web infrastructure for their employees, partners and customers.

By default, IBM TAM WebSEAL authenticates users with one-factor only, e.g. username and password. This document describes how to enhance the IBM Tivoli Access Manager for e-Business web infrastructure with two-factor authentication using the WebSEAL External Authentication Interface (EAI) and the DualShield SSO service provided by the DualShield Unified Authentication Platform.

2. Prerequisites

You must have the DualShield Authentication Server and DualShield SSO Server installed and operating (both are installed by default in the installation of the platform). For the installation, configuration and administration of DualShield Authentication and SSO servers please refer to the following documents:

- DualShield Authentication Platform – Installation Guide
- DualShield Authentication Platform – Quick Start Guide
- DualShield Authentication Platform – Administration Guide

It is expected that IBM Tivoli Access Manager WebSEAL has already been setup and operating.
3. Configuration

3.1 DualShield Configuration

In the DualShield authentication server we need to create an application which will be used for the two-factor authentication in Tivoli WebSeal. An application in DualShield needs a logon procedure which defines how users will be authenticated when they attempt to logon to the application.

3.1.1 Logon Procedure

Firstly, create a Web SSO logon procedure:

![Logon Procedure -- New](image1)

Then, modify its logon steps and add two logon steps:

![Logon Steps -- View](image2)

3.1.2 Application

The next step is to create an application in DualShield for the Web application in your WebSeal, and publish the application on the DualShield SSO server.
Use the Self-Test function to verify that the application is ready.

3.1.3 Service Provider

We need to also create a SSO Service Provider for your WebSeal.

The “Type” of the Service Provider must be set to “Generic”.

You need to enter a text string in the “EntityID” field that is used to uniquely identify the Service Provider. The EntityID should only contain alphanumeric letters.

Now, click the “Edit” button next to the “Attributes” label.
You must add the attribute named "am-eai-user-id" and maps its value to the user’s "loginName" identity attribute, as shown above.

You can add other attributes as desired.

3.2 WebSEAL Configuration

In WebSeal, we will need to create two junctions, one connecting to the backend web application that will be protected with two-factor authentication using DualShield, and the other connecting to your DualShield SSO server. As an example in this document, the backend web application to be protected is www.thesafebank.com and the junction we’ll create for the application is called “/thesafebank”. The DualShield SSO server used in the example is dualshield.deepnetsecurity.com and the junction we’ll create for the DualShield SSO server is called “/dualshield”.

3.2.1 Create Junctions

First, create a junction, “/thesafebank”, from the WebSEAL Servers for the protected backend web applications as shown in Listing 1.

Listing 1. Create a junction for the backend web application

```
# pdadmin -a sec_master -p password
pdadmin sec_master> server task default-webseald-Win2003-32-177 create -t tcp -h www.thesafebank.com -p 80 -f /thesafebank
```

Note that default-webseald-Win2003-32-177 is the server name of the WebSeal.

Second, create a junction, “/dualshield”, from the WebSEAL server to the DualShield SSO server as shown in Listing 2.

Listing 2. Create an EAI junction for the DualShield SSO

```
# pdadmin -a sec_master -p password
pdadmin sec_master> server task default-webseald-Win2003-32-177 create -t tcp -h dualshield.deepnetsecurity.com -p 8074 -c iv_user -f /dualshield
```

Note the -c iv_user junction option.

Last, the junction for the DualShield SSO server should not be in the protected domain, because unauthenticated users need to access these URLs. In Listing 3, it shows how to create the unauthenticated ACL for allowing unauthenticated access to the junction.

Listing 3. Create the ACL for the unauthenticated users

```
pdadmin sec_master> acl create unauth
pdadmin sec_master> acl modify unauth set group iv-admin TcmdbsvaBRrxl
pdadmin sec_master> acl modify unauth set group webseal-servers Tgmdbsrxl
pdadmin sec_master> acl modify unauth set any-other Trx
pdadmin sec_master> acl modify unauth set unauthenticated Trx
```

Then, attach the above ACL to the DualShield junction as shown in Listing 4.

Listing 4. Attach the ACL to the junction for the DualShield SSO server

```
pdadmin sec_master> acl attach /WebSEAL/Win2003-32-177-default/dualshield unauth
```
Note that /WebSEAL/Win2003-32-177-default is the object name in the WebSeal server.

3.2.2 Modify Configuration File

Now, we are ready to start the last part of configuring the application the DualShield SSO server with WebSEAL. It is configured through the WebSEAL configuration file, webseald-default.conf. On the Windows OS platforms, this file is located in C:\Program Files\Tivoli\PDWeb\etc\webseald-default.conf

First, enable the EAI authentication for HTTP and HTTPS sessions as shown in Listing 5.

**Listing 5. Enable the EAI authentication**

```
[eai]
eai-auth = both
```

Next, configure the authentication levels for step-up authentication. All levels of authentication will map to the EAI application:

**Listing 6. Enable the EAI authentication**

```
[authentication-levels]
level = unauthenticated
#level = password
level = ext-auth-interface
```

The next step is to specify the EAI authentication interface library in WebSEAL configuration file. Table 1 shows the EAI authentication module of different operating system.

**Table 1. EAI authentication module**

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solaris™</td>
<td>libeiauthn.so</td>
</tr>
<tr>
<td>AIX®</td>
<td>libeiauthn.a</td>
</tr>
<tr>
<td>HPUX</td>
<td>libeiauthn.sl</td>
</tr>
<tr>
<td>Linux</td>
<td>libeiauthn.so</td>
</tr>
<tr>
<td>Windows®</td>
<td>eaiauthn.dll</td>
</tr>
</tbody>
</table>

As seen in Listing 7, this document chooses eaiauthn.dll as the authentication module because it uses WebSEAL on the Windows platform.

**Listing 7. Configure the EAI authentication module**

```
[authentication-mechanisms]
# EXTERNAL AUTHENTICATION INTERFACE
ext-auth-interface = C:\Program Files\Tivoli\PDWebRTE\bin\eaiauthn.dll
```
The next step is to configure the trigger URL so that WebSEAL knows if the URL from the browser is for EAI authentication. Set the trigger URL as shown in Listing 8.

Listing 8. Configure the trigger URL

```
[eai-trigger-urls]
trigger = /dualshield/appsso/defaultauthnresponsepage*
```

Where /dualshield is the name of the junction for the DualShield SSO server.

3.2.3 Edit Login File

The next step is to edit the WebSeal’s default login file, login.html. Table 2 shows the location of login.html in different operating system.

Table 2. login.html

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX®</td>
<td>/opt/PDWeb/www-default/lib/html/C/</td>
</tr>
<tr>
<td>Linux</td>
<td>/opt/PDWeb/www-default/lib/html/C/</td>
</tr>
<tr>
<td>Windows®</td>
<td>\Program Files\Tivoli\PDWeb\www-default\lib\html\C\</td>
</tr>
</tbody>
</table>

Open login.html file in a text editor. Look for the </BODY> tag near the end of the file, and add the following lines before it:

Listing 9. Login.html

```
%EAIAUTHN%
<script type="text/javascript" language="javascript">
if((%AUTHNLEVEL% == '1') || (%AUTHNLEVEL% == ' '))
window.location.href='"/dualshield/appsso/login?authLevel=%AUTHNLEVEL%&pctx=/dualshield&dasapplicationname=webseal&entityID=WEBSEAL'"
</script>
```

Note that /dualshield is the name of the EAI junction that you created for the DualShield SSO server in Listing 2, webseal is the name of the application that you created in your DualShield server in section 3.1.2, and WEBSEAL is the EntityID of the Service Provider that you created in section 3.1.3.

Finally, restart the WebSEAL server in order to make the changes effective.
4. Authentication

Once fully configured, the backend web application /thesafebank will be protected with two-factor authentication by the EAI application /dualshield, i.e. the DualShield SSO server. To make a quick test, visit the /thesafebank application in a web browser by URL:

http://webseal.deepnetsecurity.com/thesafebank

you’ll be redirected to the DualShield SSO server to be authenticated with TFA:

http://webseal.deepnetsecurity.com/dualshield/

Once you have been successfully authenticated, you’ll be redirected back to the backend web application, /thesafebank