



APPLICATION ACCESS MANAGER

(Agent Based Credential Provider CP)

SECRETS MANAGER INTEGRATION - TECHNICAL DOCUMENTATION

Name of Company: Boardwalktech Inc.

Website: www.boardwalktech.com

Name of Product: Boardwalk Velocity / Boardwalk Digital Ledger

Version:

Date: December, 2021

PARTNER SOLUTION OVERVIEW

The Boardwalktech Velocity Platform is a low code platform that rapidly transforms Excel-based EUCs into Technology Managed Applications and is designed to mitigate the risk of Excel-based applications.

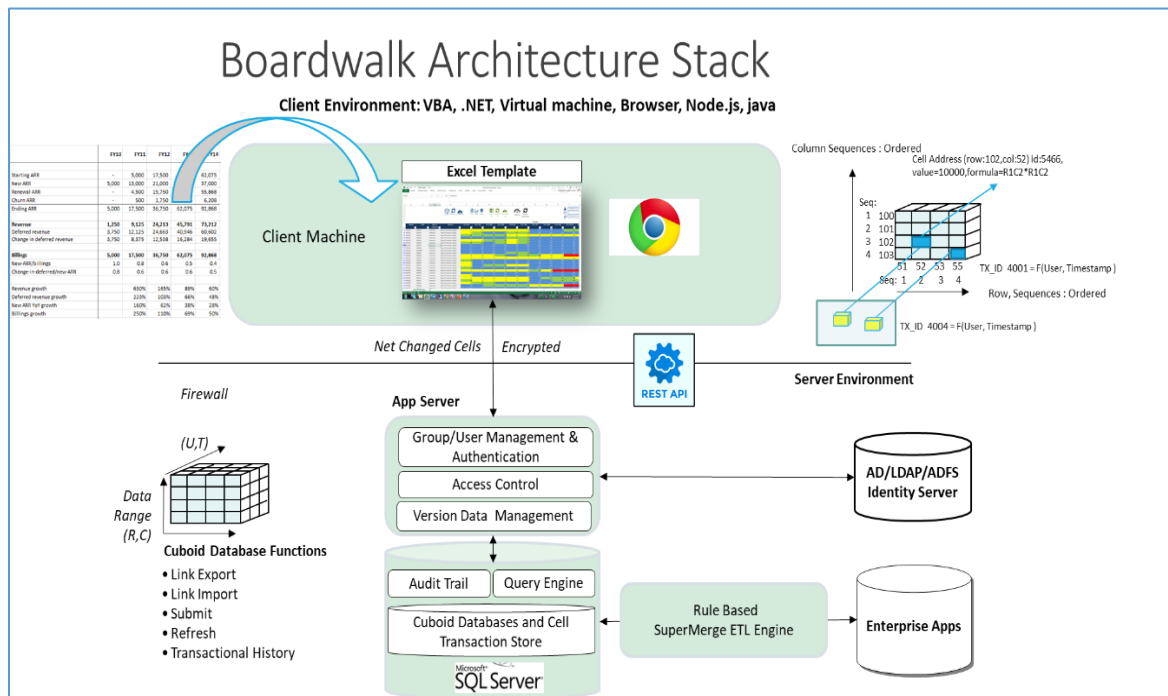
By implementing the Boardwalktech Velocity platform, Banking and financial organizations will rectify current EUC-related deficiencies and improve both ongoing data governance and effectiveness of Excel-based applications across all business units. In addition, the Boardwalktech Velocity Automation Platform digitally remediates Excel-based applications and associated business processes with the benefits of rapid re-platforming, non-intrusive impact on the user, and up to 70% less time to remediation than other re-platforming approaches.

Boardwalk Velocity integrated with CyberArk Secrets Manager to store and manage privileged accounts credentials.

Platform components:

1. App Server
2. Database Server
3. Web Browser
4. Microsoft Excel
5. SSIS

Below is the high-level architecture of platform.



KEY BENEFITS

Benefits of Boardwalk Velocity

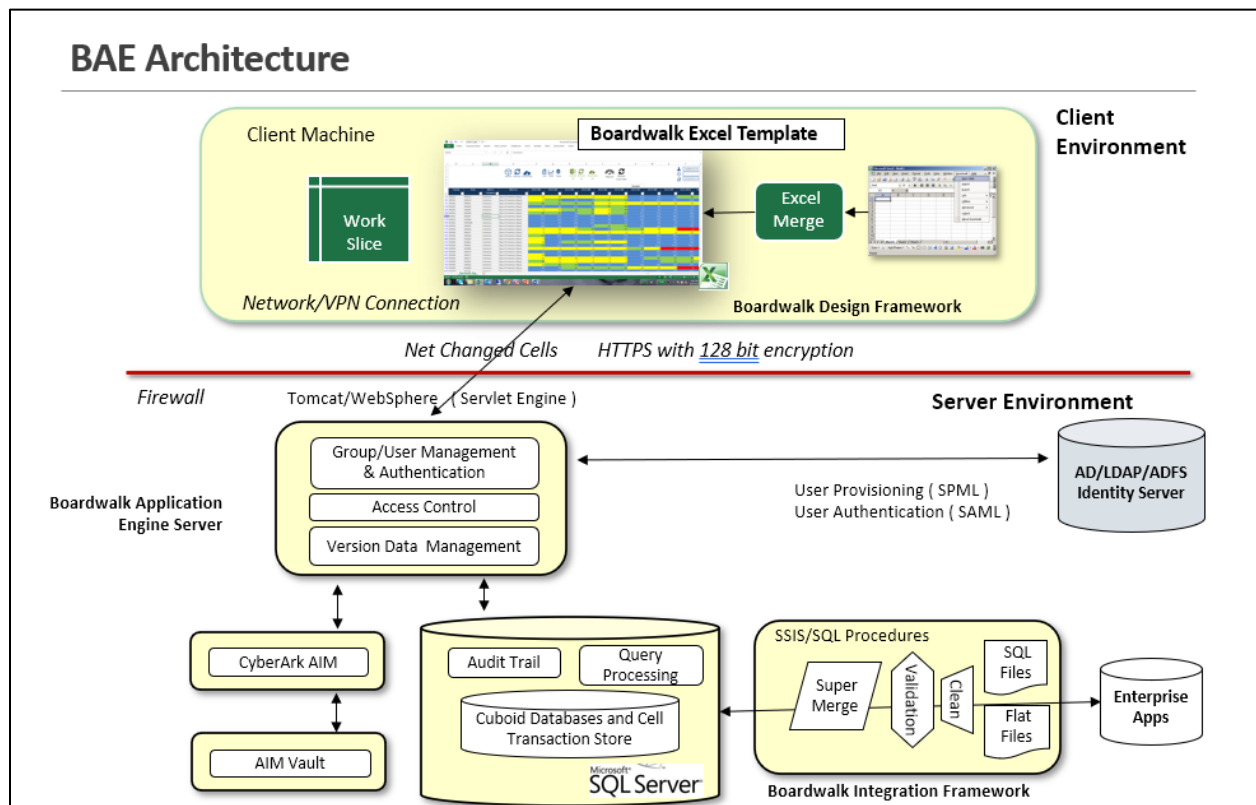
- Enable automated governance, compliance, control testing, monitoring, and reporting on EUC data previously unavailable.
- Transform the EUC without re-platforming – preserving business intellectual property (IP) and preserving existing Excel user experience.
- Mitigate EUC risks by centralizing the EUC data on the Velocity platform – delivering superior data quality, data provenance, and data governance; and eliminating data-at-rest risk on the desktop.
- Effect the EUC transformation to a technology solution with very low impact on the user and low change management – avoids unraveling formulas/macros and associated time by users re-specifying requirements.
- Rapidly transform EUCs to a "tech managed solutions" with SDLC controls at up to a 70% cost advantage compared to other low code tools.
- Improve EUC process efficiency by up to 70% by eliminating manual data handling, errors, re-work, copy/pasting, file management, and version management.
- Secure Models, Formulas, Macros, and Excel Templates with SDLC controls.
- Detect patterns across EUC applications for rationalization, further mitigating risk and improving efficiency.

Benefits of integrating Boardwalk Velocity with CyberArk Secrets Manager:

- Boardwalk Velocity integration with CyberArk Secrets Manager, credentials management is simplified as customers will not be required to store and manage their privileged accounts and passwords anymore. This significantly reduces the overhead of credential management because credentials are centrally managed in the CyberArk solution. Organizations can automatically rotate passwords, private keys and certificates based on their security policy, eliminating the need to manually update credentials within the Boardwalk platform.
- Integrating Boardwalk Velocity with CyberArk Secrets Manager will improve compliance and regulatory requirements.

PRODUCT DIAGRAM & DESCRIPTION OF PRODUCT INTEGRATION

Boardwalk Architecture depicting the CyberArk Integration



SECRETS MANAGERT INTEGRATION

The Boardwalk Application will query the Secrets Manager integration to acquire the credentials stored in the Safe. Once the secret credentials are acquired successfully, they will be used by the application program to establish the connection to Application DB.

The Secrets Manager integration resides in Java middleware on application server and typically there will be one Integration per Boardwalk Application/ EUC Process.

CREDENTIAL RETRIEVAL

The current application setup stores the DB connection parameters like database UserName and Password in **.xml** file in encrypted format. Application on startup will read the parameters from **.xml** file and use them for establishing the database connection.

Later, whenever there is need to fetch/update the data on the database server the connection is established which will depend on the user interaction with the application.

REQUIREMENTS

Please go to this [link](#) for the Boardwalk Document Library and reference the Boardwalk Server Setup Guide.

SECRETS MANAGER INSTALLATION

Refer to the “[Secrets Manager Installation Documentation](#)” for CyberArk Credential Provider installation and configuration.

SECRETS MANAGER CONFIGURATION

The configuration of CyberArk Secrets Manager will be done in **web.xml** file which contains all the configuration parameters for the application.

There are two possible ways in which database password can be retrieved from CyberArk Vault

- Object ID and related parameters
- Query and related parameters

Object Id based configuration

Following is the Object Id based configuration for retrieving the password from CyberArk

```
<init-param>
    <param-name>UseAIM</param-name>
    <param-value><1></param-value>
</init-param>
```

The above param will **enable/disable** the CyberArk Secrets Manager Integration. Default value is “0”. We will need to set this parameter to “1” to enable/start integration with CyberArk.

```
<init-param>
    <param-name>aimAPPID</param-name>
    <param-value><BWT></param-value>
</init-param>
```

The above param will define the **APPID** to be used for establishing the CyberArk Secrets Manager Integration. The corresponding App will be created in CyberArk Vault.

```
<init-param>  
  <param-name>aimSAFE</param-name>  
  <param-value><Test></param-value>  
</init-param>
```

The above param will define the name/id of the **Safe** to be used for establishing the CyberArk Secrets Manager Integration. The configured safe needs to be created in CyberArk Vault.

```
<init-param>
  <param-name>aimFOLDER</param-name>
  <param-value><root></param-value>
</init-param>
```

The above param will define the **folder** to be used for establishing the CyberArk Aim Integration. Default value root will be used if left blank.

```
<init-param>
  <param-name>aimOBJECT</param-name>
  <param-value>
    <Database-MSSql-192.168.250.207-boardwalk_application_user>
  </param-value>
</init-param>
```

The above param will define the **Object** to be used for establishing the CyberArk Secrets Manager Integration. This Object ID is generated automatically by CyberArk.

```
<init-param>
  <param-name>aimREASON</param-name>
  <param-value><getPwd></param-value>
</init-param>
```

The above param will define the **reason** to be used for establishing the CyberArk Secrets Manager Integration.

```
<init-param>
  <param-name>aimQUERY</param-name>
  <param-value>NA</param-value>
</init-param>
```

The above param will define the **Query** to be used for establishing the CyberArk Secrets Manager Integration.

This param is mutually exclusive with the combination of Appld/Safe/Object parameters and needs to be set to **NA**.

Query based configuration

Following is the Query based configuration for retrieving the password from CyberArk

```
<init-param>
  <param-name>UseAIM</param-name>
  <param-value><1></param-value>
</init-param>

<init-param>
  <param-name>aimAPPID</param-name>
  <param-value><BWT></param-value>
</init-param>
```

The above param will define the **APPID** to be used for establishing the CyberArk Secrets Manager Integration. The corresponding App will be created in CyberArk Vault.

```
<init-param>
    <param-name>aimSAFE</param-name>
    <param-value></param-value>
</init-param>
```

This parameter needs to be left blank for Query based configuration.

```
<init-param>
    <param-name>aimFOLDER</param-name>
    <param-value></param-value>
</init-param>
```

This parameter needs to be left blank for Query based configuration.

```
<init-param>
    <param-name>aimOBJECT</param-name>
    <param-value></param-value>
</init-param>
```

This parameter needs to be left blank for Query based configuration.

```
<init-param>
    <param-name>aimREASON</param-name>
    <param-value><getPwd></param-value>
</init-param>
```

The above parameter will be used for establishing the CyberArk Secrets Manager Integration.

```
<init-param>
    <param-name>aimQUERY</param-name>
    <param-value>Safe=<Test>;
        Folder=<root>;
        Username=<boardwalk_application_user>;
        Address=<10.134.85.170>;
        Database=<BAE_4_6>
    </param-value>
</init-param>
```

The above param will define the **query** to be used for establishing the CyberArk Secrets Manager Integration.

In the above query we will be using the following parameters with values separated by “;”

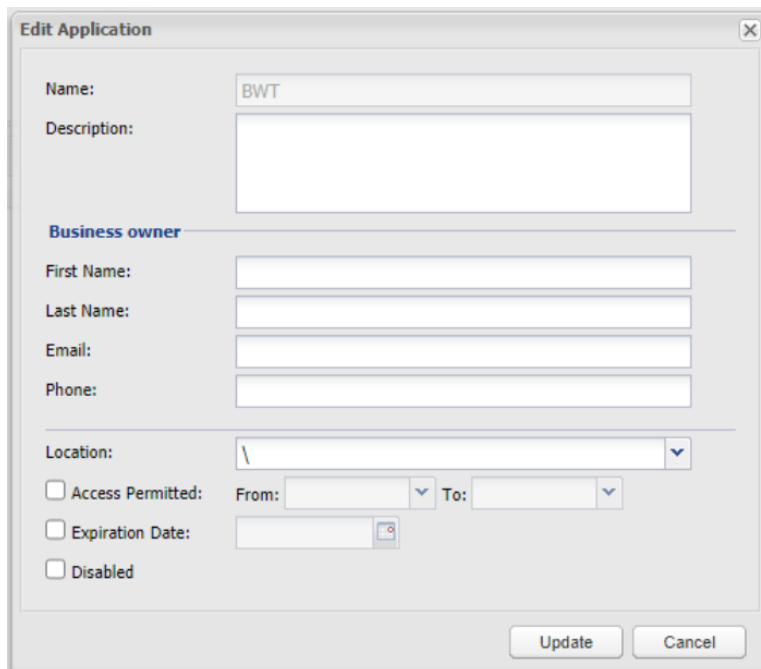
1. Safe – The name/id of safe created on CyberArk Vault.
2. Folder – The folder name, default value will be taken as root if left blank.
3. Username – Name of database user whose password needs to be retrieved.
4. Address - IP Address/DNS of the Cyber Ark Vault.
5. Database- Database Name for which the password needs to be retrieved.

Note: With this approach we will not be using the object id to retrieve the password from the vault

DEFINING THE APPLICATION ID (APPID) AND AUTHENTICATION DETAILS

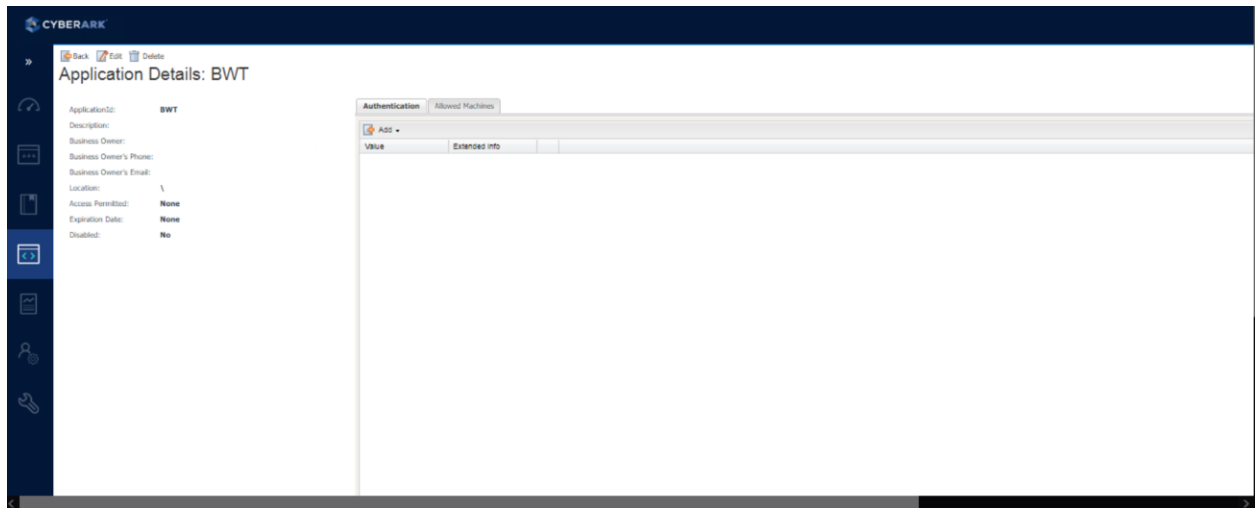
To define the application, define it manually through the CyberArk Password Vault Web Access (PVWA) interface:

- ☐ Log in as user allowed to manage applications (it requires Manage Users authorization)
- ☐ In the Applications tab, click **Add Application**. The Add Application page appears.

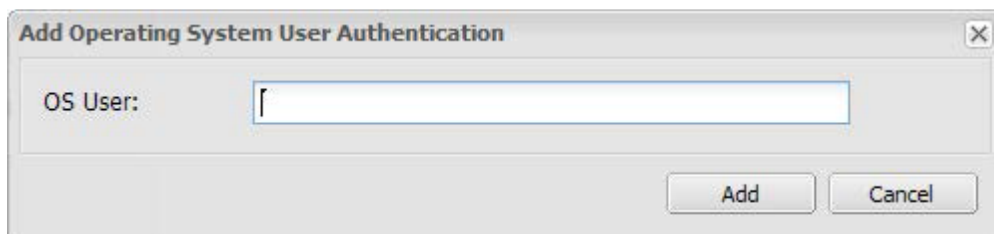


- ☐ Specify the following information:
 - In the **Name** box, specify the unique name (ID) of the application. The recommended Application ID for this integration is: APPID = BWT
 - In the **Description** box, specify a short description of the application that will help you identify it.
 - In the **Business owner** section, specify contact information about the application's business owner.
 - In the **Location** box, specify the location of the application in the Vault hierarchy. If a location is not selected, the application will be added in the same location as the user who is creating this application.

- ☐ Click **Add**. The application is added and is displayed in the Application Details page.

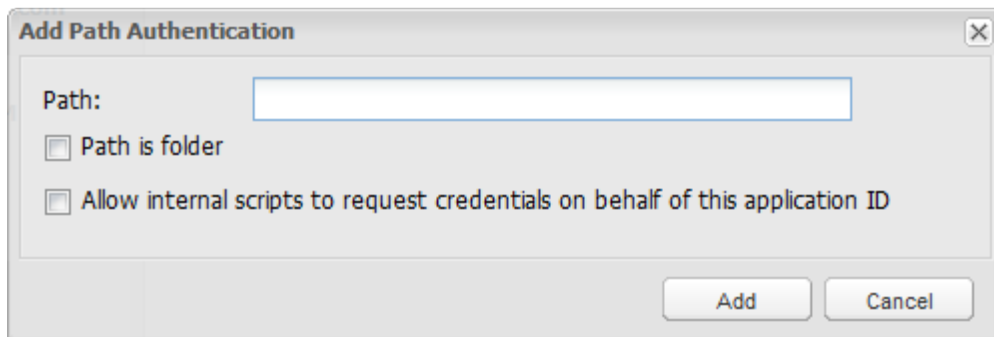


- ☐ Check the **Allowing extended authentication restrictions** box. This enables you to specify an unlimited number of machines and Windows domain OS users for a single application.
- ☐ Specify the application's **Authentication** details. This information enables the Credential Provider to check certain application characteristics before retrieving the application password.
- ☐ In the Authentication tab, click **Add**. A drop-down list of authentication characteristics is displayed.
- ☐ Select the authentication characteristic to specify.
- ☐ Select **OS user**. The Add Operating System User Authentication window appears.



- ☐ Specify the name of the OS user who will run the application, then click **Add**. The OS user is listed in the Authentication tab.

- ☐ Select **Path**. The Add Path Authentication window appears.

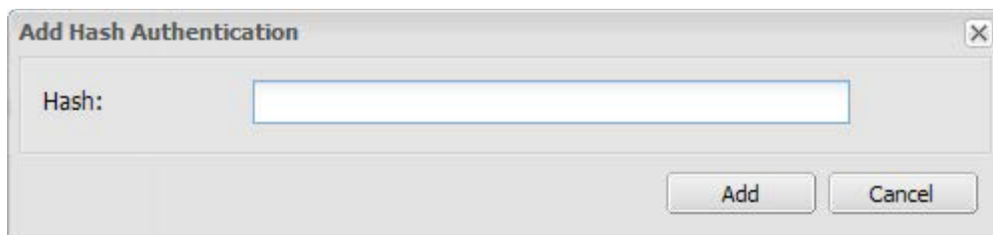
The 'Add Path Authentication' dialog box has a title bar with a close button. It contains a 'Path:' label followed by a text input field. Below the input field are two checkboxes: 'Path is folder' and 'Allow internal scripts to request credentials on behalf of this application ID'. At the bottom right are 'Add' and 'Cancel' buttons.

- ☐ In the **Path** box, specify the path where the application will run. To indicate that the specified path is a folder, select **Path is folder**. To allow internal scripts to retrieve the application password for this application, select **Allow internal scripts to request credentials on behalf of this application ID**.
- ☐ Click **Add**. The Path is added as an authentication characteristic with the information that you specified.
- ☐ Specify a hash:

- Run the AIMGetAppInfo utility to calculate the application's unique hash. Copy the hash value that is returned by the utility. Exact command is as follows

```
java -jar javaaimgetappinfo.jar GetHash/OnlyExecutablesWithAIMAnnotation=yes  
/AppExecutablesPattern="file path of BAE.war"
```

- In the PVWA, select **Hash**. The Add Hash window appears.

The 'Add Hash Authentication' dialog box has a title bar with a close button. It contains a 'Hash:' label followed by a text input field. At the bottom right are 'Add' and 'Cancel' buttons.

- In the Hash edit box, paste the application's unique hash value, or specify multiple hash values with a semi-colon. You can add additional information in a comment after each hash value specified for an application by specifying '#' after the hash value, followed by the comment.
For example, OE883B70D5B6E3EE37D37198049C9507C8383DB6 #app2
Note: The comment must not include a colon or a semicolon.
 - Click **Add**. The Hash is added as an authentication characteristic with the information that you specified.
- ☐ Specify the application's Allowed Machines. This information enables the Credential Provider to make sure that only applications that run from specified machines can access their passwords.
 - In the Allowed Machines tab, click **Add**. The Add allowed machine window is displayed.

- In the **Address** box, specify the IP/hostname/DNS of the machine where the application will run and will request passwords, then click **Add**. The IP address is listed in the Allowed Machines tab. Make sure the servers allowed include all mid-tier servers or all endpoints where the Secrets Manager Credential Providers were installed.

PROVISIONING ACCOUNTS AND SETTING PERMISSIONS FOR APPLICATION ACCESS

For the application to perform its functionality or tasks, the application must have access to existing accounts, or new accounts to be provisioned in CyberArk Vault.

In the Password Safe, provision the privileged accounts that will be required by the application. You can do this in either of the following ways:

- **Manually** – Add accounts manually one at a time, and specify all the account details.
- **Automatically** – Add multiple accounts automatically using the Password Upload feature.

For this step, you require the **Add accounts** authorization in the Password Safe.

For more information about adding and managing privileged accounts, refer to **the Privileged Access Security Implementation Guide**.

Once the accounts are managed by CyberArk, make sure to setup the access to both the application and CyberArk Application Password Providers serving the Application.

Add the Credential Provider and application users as members of the Password Safes where the application passwords are stored. This can either be done manually in the Safes tab, or by specifying the Safe names in the CSV file for adding multiple applications.

☐ Add the Provider user as a Safe Member with the following authorizations:

- List accounts
- Retrieve accounts
- View Safe Members

Note: When installing multiple Providers for this integration, it is recommended to create a group for them, and add the group to the Safe once with the above authorization.

Add Safe Member

Search:

Search In:

Vault

Search

Selected Search: Vault

Name	Business Email	Full Name
------	----------------	-----------

☐ Access

☐ Use accounts

☒ Retrieve accounts

☒ List accounts

☐ Account Management

☐ Safe Management

☐ Monitor

☐ View Audit log

☒ View Safe Members

Add

Close

- ☐ Add the application (the APPID) as a Safe Member with the following authorizations:
- Retrieve accounts

Partner Application with Object based configuration:

```
web.xml
343
344     <init-param>
345         <param-name>UseAIM</param-name>
346         <param-value>0</param-value>
347     </init-param>
348
349     <init-param>
350         <param-name>aimAPPID</param-name>
351         <param-value>BWT</param-value>
352     </init-param>
353
354     <init-param>
355         <param-name>aimSAFE</param-name>
356         <param-value>Test</param-value>
357     </init-param>
358
359     <init-param>
360         <param-name>aimFOLDER</param-name>
361         <param-value>root</param-value>
362     </init-param>
363
364     <init-param>
365         <param-name>aimOBJECT</param-name>
366         <param-value>Database-MSSql-192.168.250.207-boardwalk_application_user</param-value>
367     </init-param>
368
369     <init-param>
370         <param-name>aimREASON</param-name>
371         <param-value>getPwd</param-value>
372     </init-param>
373
374     <init-param>
375         <param-name>aimQUERY</param-name>
376         <param-value>NA</param-value>
377     </init-param>
378
379
```

Note: The Query parameter **aimQUERY** set to NA when not in use

Partner Application with Query based configuration:

```
web.xml
345
346     <init-param>
347         <param-name>UseAIM</param-name>
348         <param-value>1</param-value>
349     </init-param>
350
351     <init-param>
352         <param-name>aimAPPID</param-name>
353         <param-value>BWT</param-value>
354     </init-param>
355
356     <init-param>
357         <param-name>aimSAFE</param-name>
358         <param-value></param-value>
359     </init-param>
360
361     <init-param>
362         <param-name>aimFOLDER</param-name>
363         <param-value></param-value>
364     </init-param>
365
366     <init-param>
367         <param-name>aimOBJECT</param-name>
368         <param-value></param-value>
369     </init-param>
370
371     <init-param>
372         <param-name>aimREASON</param-name>
373         <param-value>getPwd</param-value>
374     </init-param>
375
376     <init-param>
377         <param-name>aimQUERY</param-name>
378         <param-value>Safe=Test;Folder=root;Username=boardwalk_application_user;Address=10.134.85.170;Database=BAE_4_6</param-value>
379     </init-param>
380
```

Note: Query parameter has set value, the **Safe/Folder/Object** params are now set to blank

PARTNER CONTACT INFO

Business Contact	Name	Charlie Glavin
	Email	charlie.glavin@boardwalktech.com
	Tel	650-618-6135
Technical Contact	Name	Sanjeev Patil
	Email	Sanjeev.patil@boardwalktech.com
	Tel	+91-9920017270
Support Contact	Name	Edward Madrak
	Email	edward.madrak@boardwalktech.com
	Tel	650-618-6135