Fischer International Identity
Fischer Identity Suite 4.2

RSA SecurID Ready Implementation Guide

Last Modified: June 16, 2010

Partner Information

Product Information	
Partner Name	Fischer International Identity
Web Site	www.fischerinternational.com
Product Name	Fischer Identity Suite
Version & Platform	V4.2 – Cross Platform (windows/unix/linux)
Product Description	Fischer Identity suite is an Identity management product that enables organizations to automate their provisioning processes. It also provides a comprehensive list of self-service password management and provisioning features that allow users to securely request resources, claim accounts, reset passwords etc.
Product Category	Provisioning

Solution Summary

The integration described in this guide enables Fischer Identity Suite to automate the provisioning and de-provisioning of RSA SecurID tokens, users, groups and agents. In addition, it allows Fischer Identity to offer RSA SecurID two-factor authentication to users who access the Identity Self-Service, where they can perform various password management and provisioning functions. The following list explains these features in more detail:

1. **Provisioning of RSA SecurID tokens, users and groups**: As a provisioning solution, Fischer Identity Suite provides administrators with the ability to export, add, modify and delete RSA users, groups and SecurID tokens. These functions apply to both standard and custom attributes. Administrators can enable and disable tokens and reset token PIN numbers and user passwords. In addition, companies can automate the token provisioning/de-provisioning processes to/from employees.
2. **Reconciling data between Fischer Identity and RSA Authentication Manager repositories**: Administrators can create and schedule Fischer Identity Suite workflows to synchronize user, group and token data between the two systems. This ensures that any changes made to RSA Authentication Manager data outside of Fischer Identity Admin Console will be reconciled.
3. **RSA SecurID Authentication for the Self-Service Portal**: End users can log in to Fischer Identity Suite's self-service portal with RSA SecurID tokens to perform various functions like requesting resources, resetting SecurID token PINs and passwords.

Note: A full list and description of provisioning features as well as instructions on how to use those features are well outside of this document's scope. In addition, the installation and configuration instructions in this guide are limited to those necessary for a basic deployment. There are many non-standard configuration options that are also out of the scope of the document. Please visit http://fischerinternational.com/ for additional documentation.

Partner Integration Overview	
Support for Standard Card, Key Fob, PIPAD	Yes
Export/Add/Modify/Delete User (principal)	Yes
Export/Modify/Delete Token	Yes
Export/Add/Modify/Delete Group	Yes
Export/Add/Modify/Delete Agent Host	Yes
Assign & un-assign a token to/from user	Yes
Assign & un-assign a user to/from group	Yes
Assign & un-Assign a Agent Host to/from Group	Yes
Set token in New PIN Mode	Yes
Reset a token's PIN	Yes
Enable & Disable a Token	Yes
Next Tokencode Mode support	Yes
Reset a user's password	Yes

Product Requirements

Fischer Identity suite is a web application that supports Windows 2003/2008 Server, Unix and Linux platforms.

Operating System Support:

Partner Product Requirements: Fischer Identity Suite	
Version	
Server - Identity Suite v4.2	All Patch Levels Supported
Client – Internet Explorer 7 or 8	Latest patch as recommended by Microsoft

Operating System	
Platform	**Required Patches**
Windows 2003 Server	As recommended by Microsoft
Windows 2008 Server	As recommended by Microsoft

Additional Software Requirements:

Additional Software Requirements	
Application	**Additional Patches**
Application Server: Apache Tomcat 6.0.18	
Java : Sun Java Development Kit 1.6	Update 18 or later
Data base (Any one)MSSQL / Oracle / PostgreSQL	
LDAP - One of SunOne / Active Directory 2003/2008	

Partner Product Configuration

Before You Begin

This section provides instructions for integrating RSA Authentication Manager resources into the Fischer Identity Suite enterprise identity management system. This document is not intended to suggest optimum installations or configurations.

It is assumed that the reader has both working knowledge of Fischer Identity Suite, RSA Authentication Manager and their respective components, as well as the ability to perform the tasks outlined in this section. Administrators should have access to the relevant product documentation in order to install, configure and use these products.

Fischer Identity Suite and RSA Authentication Manager must be installed and working prior to the integration. Perform the necessary tests to confirm that this is true before proceeding.

Configuration

This section contains instructions to configure the Fischer Identity Suite - RSA Authentication Manager integration. It is divided into the following three sections:

- Configure the Fischer Identity Server Environment
- Configure Fischer Identity Suite Provisioning and Authentication
- Configure Two-Factor Authentication for Self-Service Users

Configure the Fischer Identity Server Environment

The following steps must be performed to set up Fischer Identity Suite's environment.

- Copy RSA Authentication Manager JAR File to the Fischer Identity Server
- Enable Secure RSA Authentication Manager API Communication
- Set the *allowArraySyntax* JAVA_OPTS Parameter

Note: The instructions in this section use the variable names listed in the table below. Please replace each name with its corresponding value.

Variable Name	Description
%CACERTS_PASSWORD%	The password for Java's system-wide keystore, *cacerts*; its default value is "***changeit***"
%DATAFORUM_HOME%	The Fischer Identity Suite provisioning component's home directory
%IDENTITY_HOME%	The Fischer Identity Suite identity component's home directory
%JAVA_HOME%/	The JDK's root directory on the Fischer Identity Suite server host
%RSA_AM_HOME%	RSA Authentication Manager's installation directory
%SDK_HOME%	The RSA Authentication Manager SDK's root directory
%SERVER_NAME%	The RSA Authentication Manager host name
%TRUSTSTORE_PATH%	The Fischer Identity Server's cacerts keystore. This value is specified in Fischer Identity's **prio.properties** file

Copy RSA Authentication Manager jar files to the Fischer Identity Server

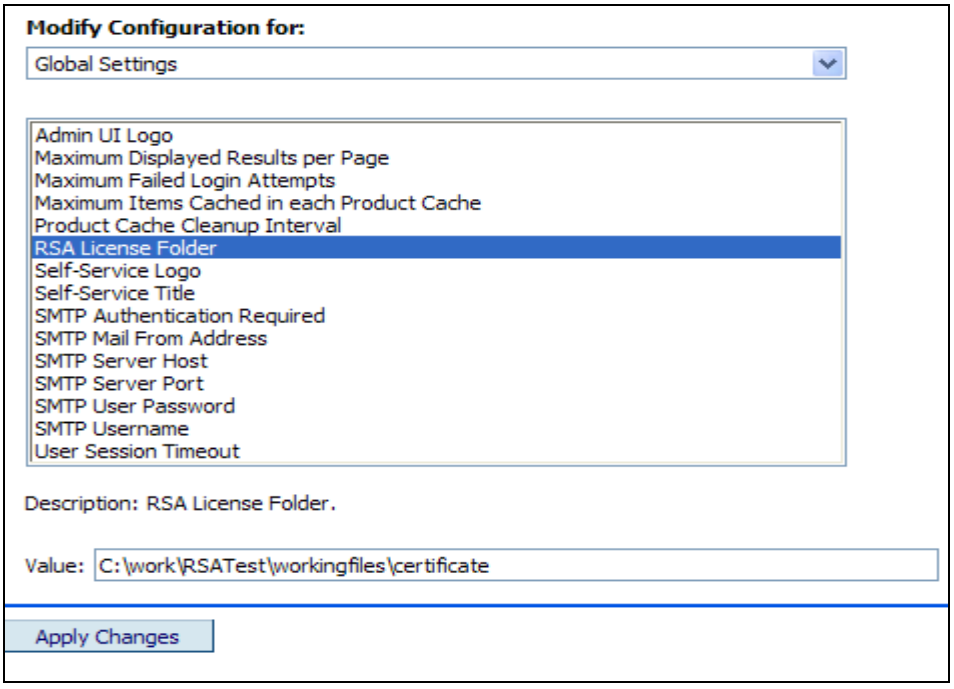
1. Open a command prompt on the Authentication Manager host, change directories to %RSA_AM_HOME%/*appserver/weblogic/server/lib/* and enter the following command:

```
java -jar ../../../modules/com.bea.core.jarbuilder_1.0.0.0.jar -profile
wlfullclient
```

2. Locate the following files in the RSA Authentication Manager server installation directories and copy them to the Fischer Identity Server under JAVA_HOME\jre\lib\ext:

   - %RSA_AM_HOME%/appserver/license.bea
   - %RSA_AM_HOME%/appserver/modules/com.bea.core.process_5.3.0.0.jar
   - %RSA_AM_HOME%/appserver/weblogic/server/lib/wlfullclient.jar
   - %RSA_AM_HOME%/appserver/weblogic/server/lib/wlcipher.jar
   - %RSA_AM_HOME%/appserver/weblogic/server/lib/EccpressoAsn1.jar
   - %RSA_AM_HOME%/appserver/weblogic/server/lib/EccpressoCore.jar
   - %RSA_AM_HOME%/appserver/weblogic/server/lib/EccpressoJcae.jar

3. Copy the *license.bea* to a share folder accessible by the Identity and Provisioning Servers
4. Login to the **Fischer Identity Admin UI**, go to **Configuration → Configuration → Global Settings**, select ***RSA License Folder*** and specify the path to the shared folder that contains the *license.bea* file from step 3.



Enable Secure RSA Authentication Manager API Communication

The RSA Authentication Manager installation process creates a self-signed root certificate and stores it in %RSA_AM_HOME%/*server/security/*%SERVER_NAME%.*jks*. This certificate must be imported into a keystore on the Fischer Identity Server to enable a secure communication channel.

The installer also creates a username and password to secure API connections to the command server. The Fischer Identity Server will use these credentials to establish an API connection.

Please perform the following procedures to enable secure communication:

- **Export the Server Root Certificate**
- **Import the Server Root Certificate**
- **Set the Command Client's Username and Password**

Export the Server Root Certificate

Run the **keytool** utility to export the RSA Authentication Manager server's root certificate as follows:

1. Open a command prompt on the server's host, go to the %RSA_AM_HOME%/appserver/ directory and enter the following command:

```
jdk/jre/bin/keytool -export -keystore
%%RSA_AM_HOME%/server/security/%SERVER_NAME%.jks -file
am_root.cer -alias rsa_am_ca
```

2. When prompted for the keystore password, leave the password field empty and press ***Enter***. Ignore the warning message displayed by the **keytool**.

Import the Server Root Certificate

To import the RSA Authentication Server's root certificate into Fischer Identity Server's trusted keystore:

1. Open the Fischer Identity Server's *prio.properties* file and find the *TruststorePath* variable's value. This value is the path to the Fischer Identity Server's trusted keystore. The *prio.properties* file can be found at the following locations:
    - *%IDENTITY_HOME%\config*
    - *%DATAFORUM_HOME%\config*
2. Locate the root certificate file that was exported in the previous section and copy it to the Fischer Identity Server at the location specified by %TRUSTSTORE_PATH%.
3. Open a command prompt on the Fischer Identity Server host, change directories to the one specified by *TruststorePath* and enter the following command:

```
keytool -import -keystore %TRUSTSTORE_PATH%/lib/java/trust.jks -storepass
%CACERTS_PASSWORD%[1] -file am_root.cer -alias rsa_am_ca -trustcacerts
```

[1]The default cacerts keystore password is "***changeit***".

Set the Command Client User Name and Password

During the RSA Authentication Manager installation process, the system creates a command client user name and password to secure connections to the command server. These credentials are randomly generated and unique for each deployment. As such, each client must have its corresponding set of credentials to establish a command server connection.

To obtain the command client user name and password from RSA Authentication Manager:

1. Open a command prompt on the RSA Authentication Manager host, change directories to *%RSA_AM_HOME%/utils* and enter the following command:

```
rsautil manage-secrets --action list
```

2. When prompted, type the master password, and the system will display the list of internal system passwords.
3. Locate the command client user name and password. For example:

```
Command Client User Name ..................: CmdClient_vKr9aLK9
Command Client User Password ..............: e9SHbK0W4i
```

Each username and password pair will be needed for the associated client to connect to the command server. See the Configure Fischer Identity Suite Provisioning and Authentication section for more details.

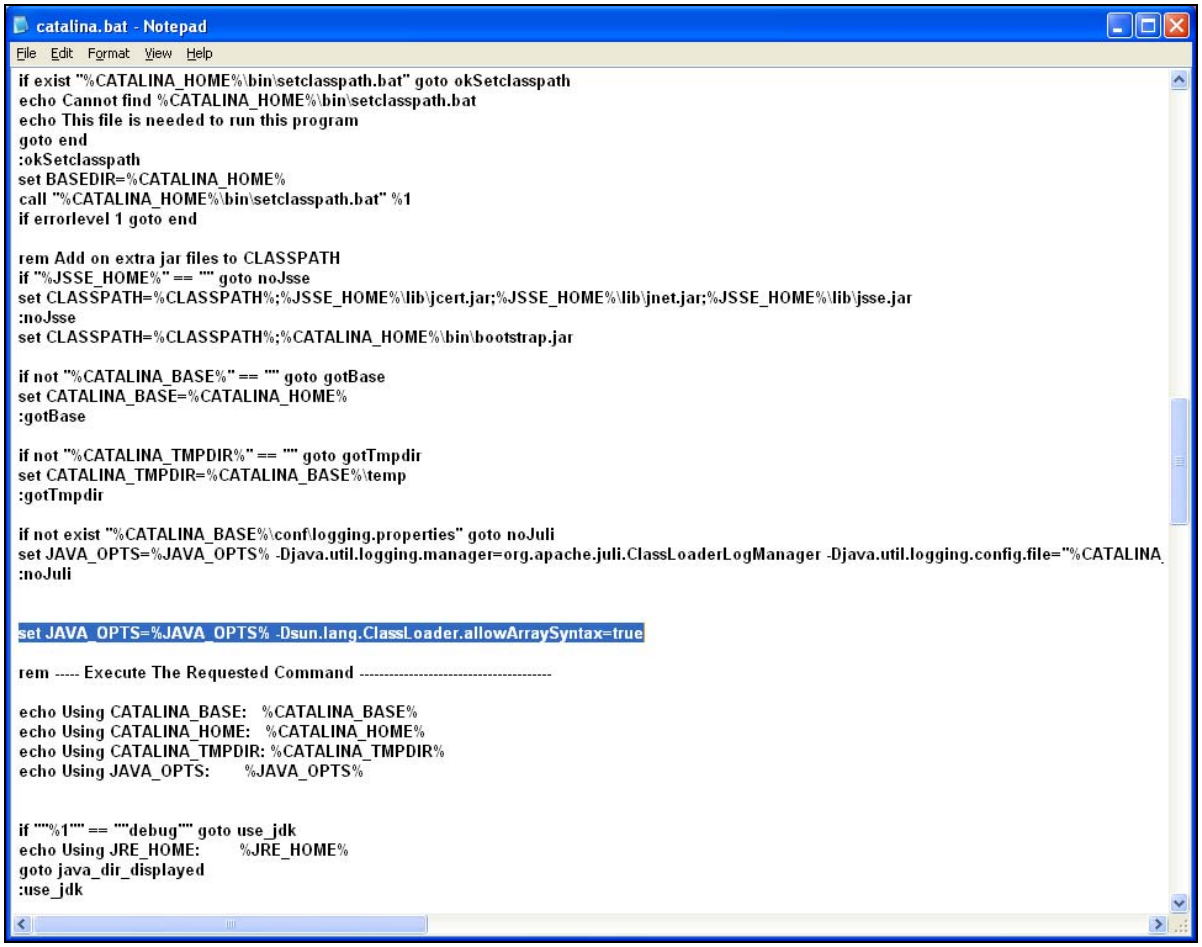
Important: Do not change the command client user name and password. Any change to these values can cause serious issues in the operation of RSA Authentication Manager.

Set the *allowArraySyntax* JAVA_OPTS Parameter

Application server startup scripts use a variable called either *JAVA_OPTIONS* or *JAVA_OPTS* to pass command line parameter to the Java virtual machine during the startup process. For application servers that are using JDK 1.6, the *allowArraySyntax* parameter must be set to ***true***. The following is an example of how to set this *JAVA_OPTS* parameter on Tomcat. Please consult the appropriate documentation for instructions to set this parameter on other supported application servers.

1. Open *catalina.bat* in a text editor, and add the following line:

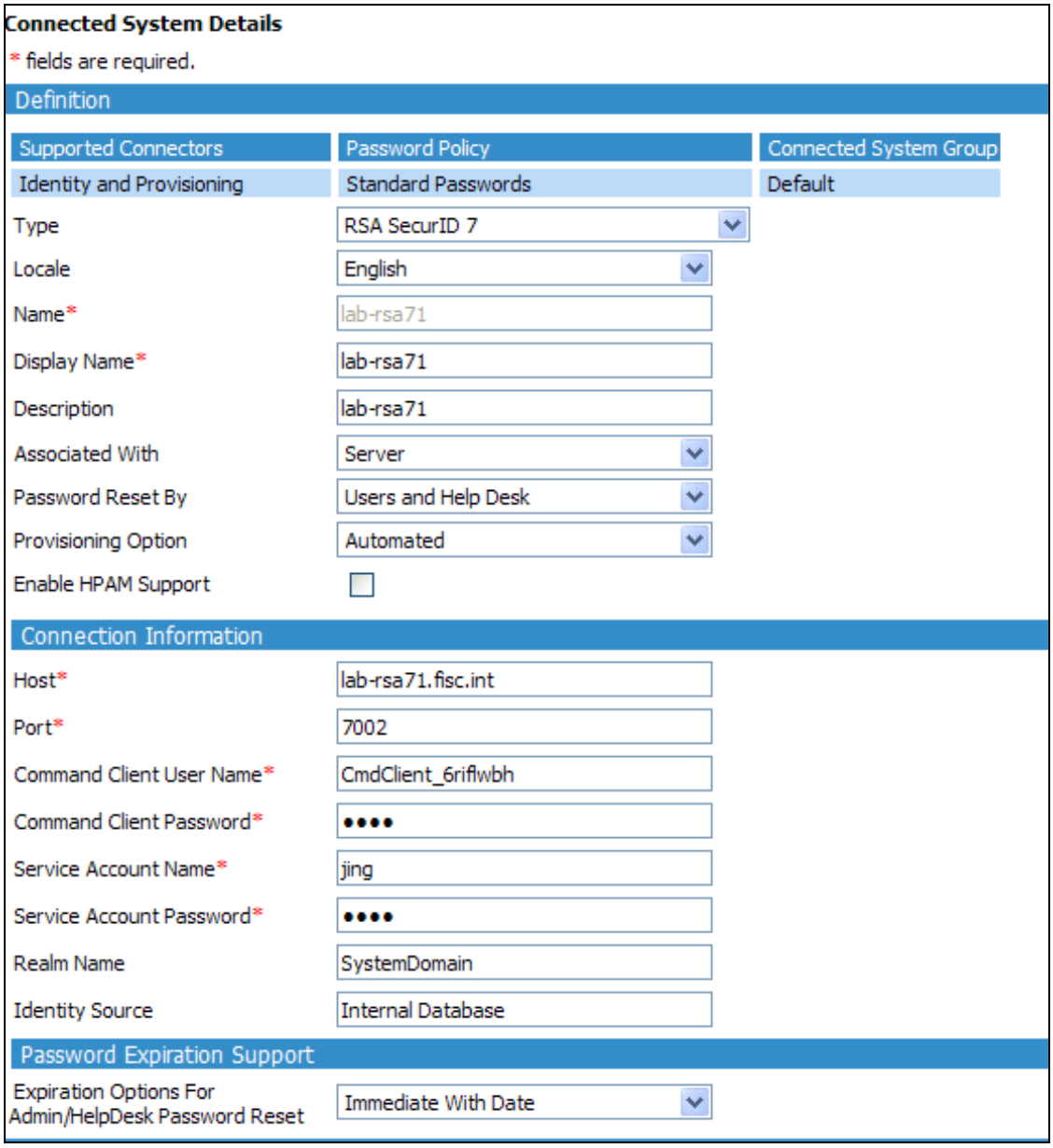
   ```
   set JAVA_OPTS=%JAVA_OPTS% -Dsun.lang.ClassLoader.allowArraySyntax= "true"
   ```



2. Save the changes, close the file and restart the server.

Configure Fischer Identity Suite Provisioning and Authentication

1. Log in to the **Fischer Identity Admin UI** as a user with rights to manage connected systems and define a new connected system of type ***RSA SecurID 7***.
2. Enter a meaningful name in the **Name** and **Display Name** fields.
3. Enter the RSA Authentication Manager server's host name and port in the in the **Host** and **Port** fields respectively.
4. Enter the user name for the RSA command server in the **Command Client User Name** field.
5. Enter the RSA command server user's password in the **Command Client Password** field.



6. Enter the user ID for the account that will be used to administer provisioning tasks in the **Service Account Name** field. This user account must have the following permissions:
    - Add/modify/delete user (principal)
    - Modify/delete token
    - Add/modify/delete group
    - Add/modify/delete agent host
    - Assign/Unassign token to user
    - Assign/Unassign user to group
    - Reset PIN
    - Set token to New PIN Mode
7. Enter the password for the account mentioned in Step 6 in the **Service Account Password** field.
8. Optionally, enter an RSA Authentication Realm name in the **Realm Name** field. If this field is left blank, the Fischer Identity provisioning process will use the *TaskRealm* specified in the workflow. Please consult the *RSA Authentication Manager Administration Guide* for more information about RSA Authentication Manager Realms.
9. Optionally, enter the name of the identity source in the **Identity Source** field. If this field is left blank, the Fischer Identity provisioning process will use the *TaskIdentitySource* specified in the workflow definition. Please consult the *RSA Authentication Manager Administration Guide* for more information about Identity Sources.

> **NOTE: If the realm name and identity source are not provided, the Fischer Identity authentication and password management functionality will use "SystemDomain" as the default realm, and "Internal Database" as the default identity source.**

10. Optionally, select a password expiration preference from the **Expiration Options for Admin/HelpDesk Password Reset** dropdown list. The following password expiration options are available:
    - Set a user's password to expire immediately
    - Set a user's password to expire on a given date
    - Set an RSA SecurID token's PIN to expire immediately (New PIN mode)
11. Click the **Test Connection** button to verify the RSA Authentication Manager server connection.
12. Click the **Add** button to save the connected system information.

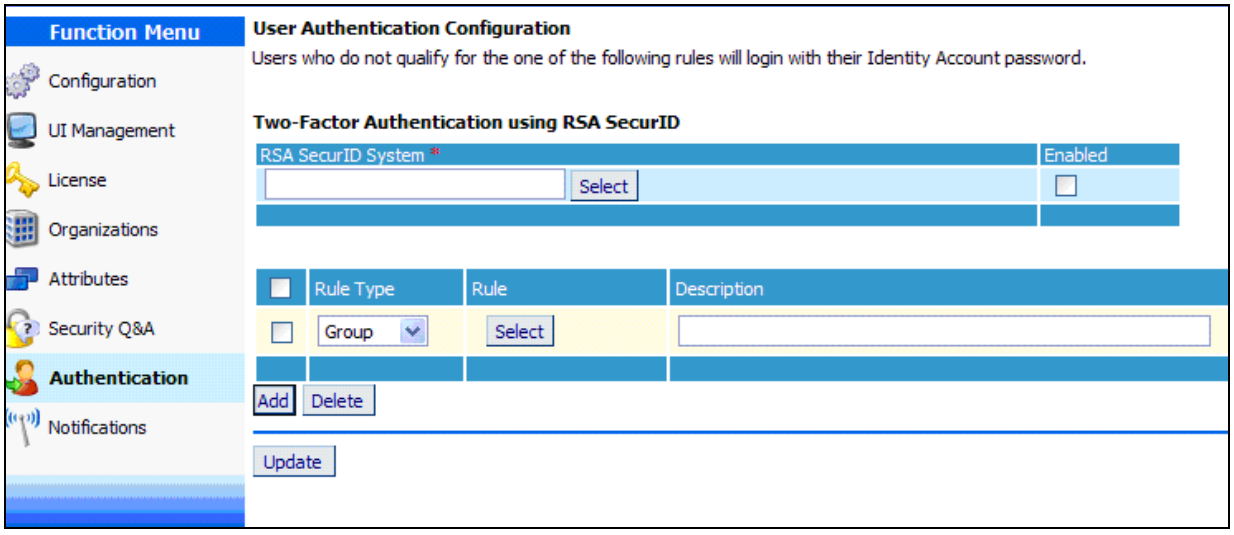
Configure Two-Factor Authentication for Self-Service Users

This section contains instructions for enabling RSA SecurID two-factor authentication for Fischer Identity Self-Service users. It is divided into the following to subsections:

- Define an RSA SecurID Authentication Rule
- Associate an Identity User with the RSA User and RSA SecurID Token(s)

Define an RSA SecurID Authentication Rule

1. Log in to the Fischer Identity Suite Admin UI and navigate to **User Authentication Configuration Page by selecting Configuration → Authentication**:



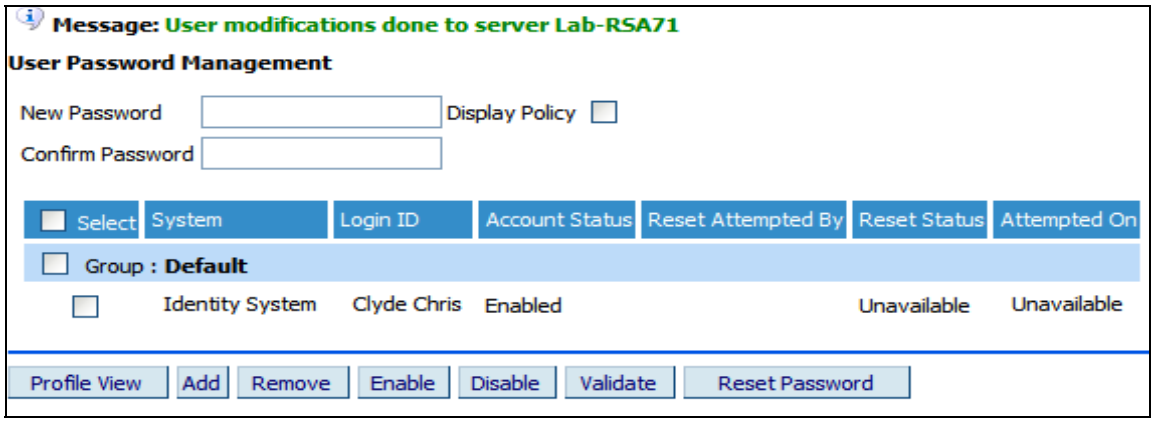
2. Define rules for enforcing RSA SecurID two-factor authentication to log in to the **Fischer Identity Self-Service Portal** based on the current business requirements. For more information about creating Fischer Identity Authentication rules, please consult the appropriate Fischer Identity Suite administration documentation.
3. Click on the **Select** button and pick the RSA SecurID system that was defined in the first step in the Configure Fischer Identity Suite Provisioning and Authentication section.
4. Check the **Enabled** check box and click the **Add** button to add a new rule.
5. Select one of the following rule types from the **Rule Type** dropdown list:
    - **User** (a rule that will apply to a single user)
    - **Group** (a rule that will apply to many, pre-assigned users)
    - **Dynamic Filter** (a rule that will apply to a group of users who are filtered at runtime based on a certain criteria, such as: ***department = sales***)
6. Click **Update** to save the changes.

Associate an Identity User with the RSA User and RSA SecurID Token(s)

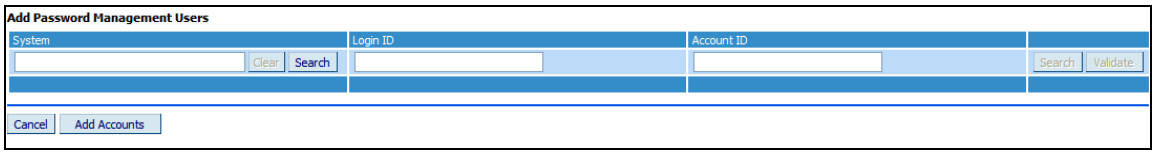
This section contains instructions for mapping Fischer Identity users to RSA users and their associated RSA SecurID tokens.

> **Note: This step can be done either through the provisioning process or manually through the Admin UI.**

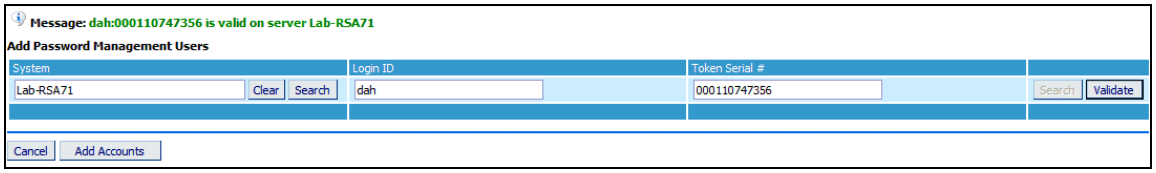
1. Select a user profile in the **Fischer Identity Suite Admin UI**, and click the **Password View** button to view all accounts associated with the user.
2. Click the **Add** button to add a new account.



3. Click the **Search** button to select an RSA system.

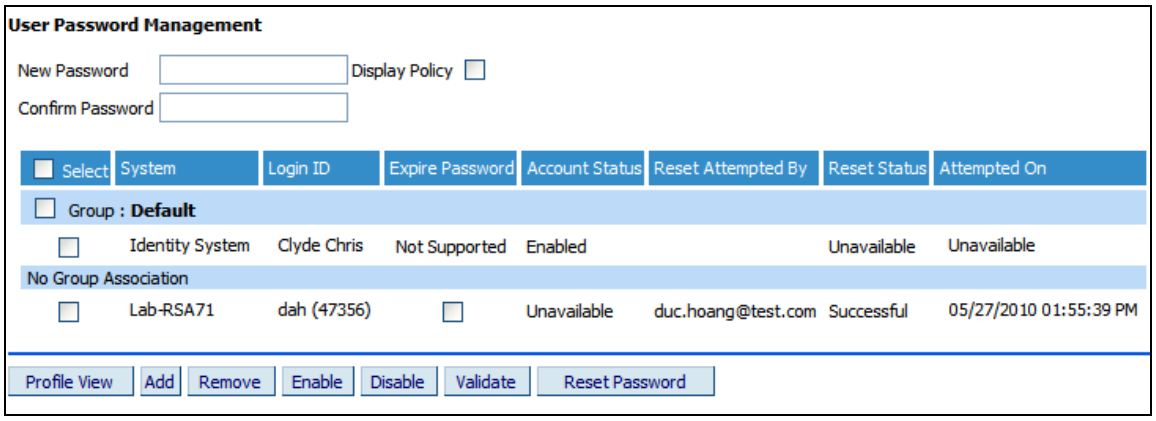


4. Enter the appropriate RSA Authentication Manager user name and a corresponding RSA SecurID token serial number (if associating the user with a RSA SecurID token) into the **Login ID** and **Token Serial Number** fields. If the user won't be authenticating with RSA SecurID, enter the word "***PASSWORD***" into the **Login ID** field.



5. Optionally, click the **Validate** button to validate the account with RSA Authentication Manager.

6. Click **Add Accounts** to associate this account with the user

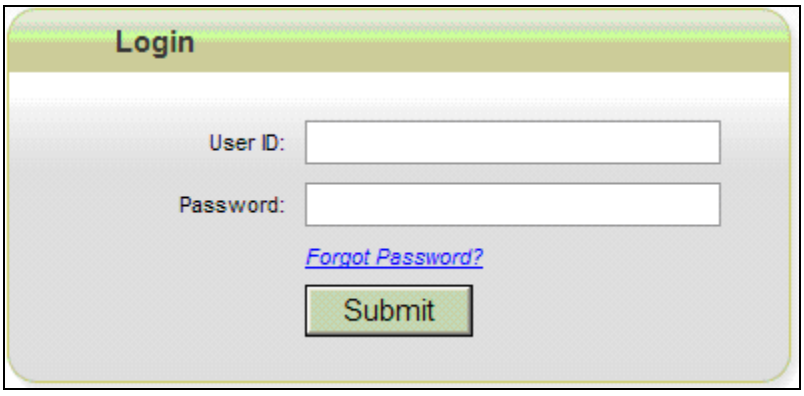


User Experience

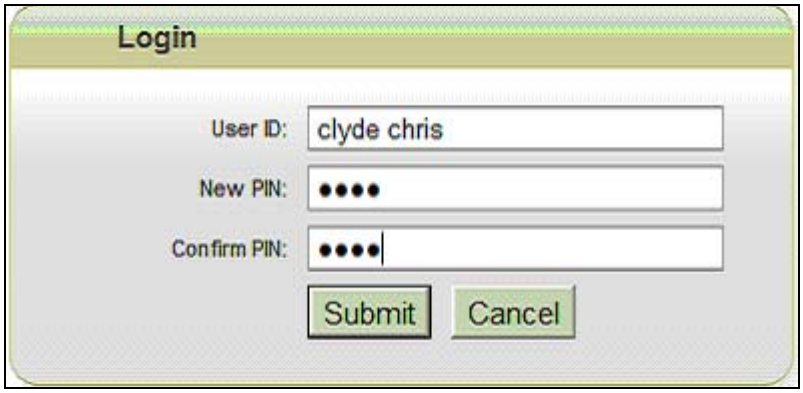
Once RSA SecurID rules have been defined and assigned to **Fischer Identity Self-Service Portal** users, these users will be prompted to log in to the portal with their SecurID passcodes.

1. The selected **Self-Service Portal** users will be prompted for their Fischer Identity user names and RSA SecurID passcodes.

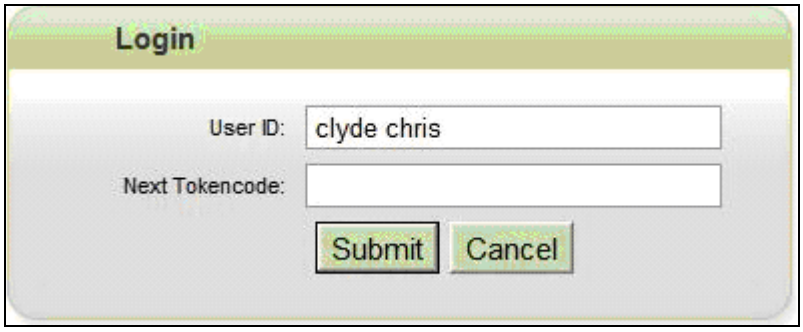
> **Note: RSA SecurID token users will enter their passcodes in the Password field.**



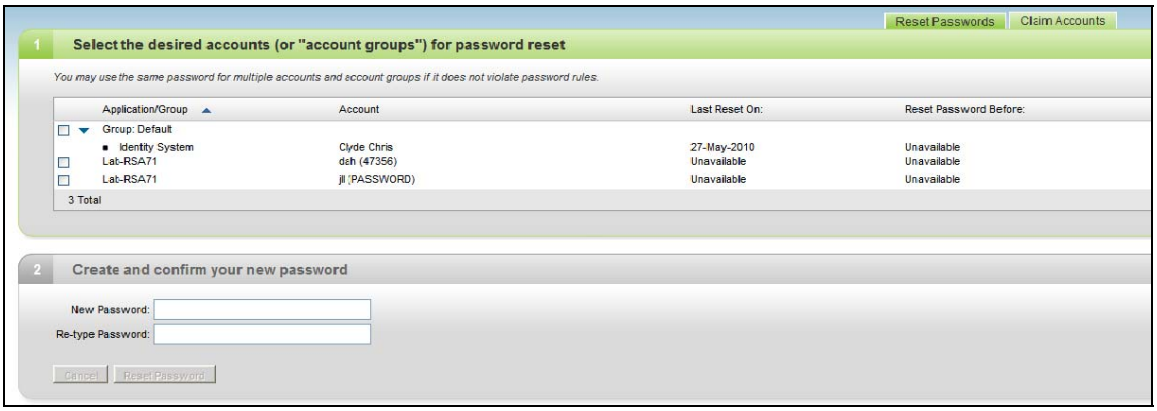
2. The following screen is displayed when a token is in **New PIN Mode**.



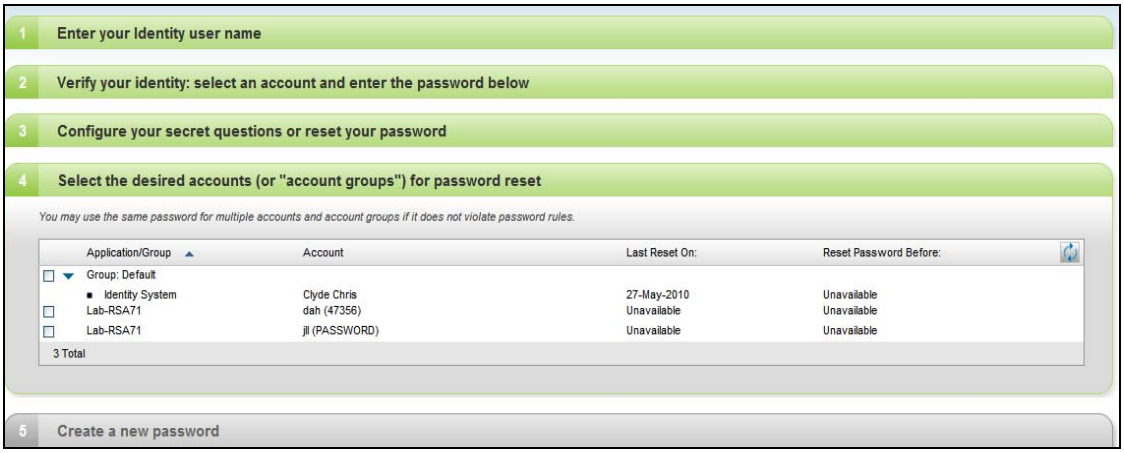
3. The following screen is displayed when a token is in **Next Tokencode Mode**.



4. Once authenticated by RSA Authentication Manager, these users can now reset their RSA SecurID token PINs or RSA Authentication Manager static passwords via **Self-Service.**



They may also perform these actions from the **Kiosk.**



Certification Checklist For RSA Authentication Manager v7.1

Date Tested: 06/15/2010

Certification Environment		
Product Name	**Version Information**	**Operating System**
RSA Authentication Manager	7.1 SP3	Windows 2003 Server
Fischer Identity Suite	4.2	Windows 2008 Server

Mandatory Functionality			
RSA Native Protocol		**RADIUS Protocol**	
Export[2]/Add/Modify/Delete User (Principal)	✓		N/A
Export/Modify/Delete Token	✓		N/A
Export/Add/Modify/Delete Group	✓		N/A
Export/Add/Modify/Delete Agent Host	✓		N/A
Assign or un-assign a token to/from user	✓		N/A
Assign or un-assign a user to/from group	✓		N/A
Assign or un-assign a AgentHost to/from Group	✓		
Enable or disable a token	✓		N/A
Set a token to new PIN mode	✓		N/A
Set a token to Next Tokencode mode	✓		N/A
Reset a token's PIN	✓		N/A
Reset a user's password	✓		N/A

JGS / PAR

✓ = Pass ✗ = Fail N/A = Non-Available Function

[2] Note that "Export" functionality for users, tokens, groups and hosts refers to the product's ability to export data associated with these objects from Authentication Manager. This functionality is used, for instance, during data reconciliation.

Certification Checklist For RSA Authentication Manager 7.1‡

Date Tested: June 15, 2010

Certification Environment		
Product Name	**Version Information**	**Operating System**
RSA Authentication Manager	7.1	Windows 2003 Server
Fischer Identity Suite	4.2	Windows 2008 Server

Mandatory Functionality			
RSA Native Protocol		**RADIUS Protocol**	
New PIN Mode			
Force Authentication After New PIN	✓	Force Authentication After New PIN	N/A
System Generated PIN	✓	System Generated PIN	N/A
User Defined (4-8 Alphanumeric)	✓	User Defined (4-8 Alphanumeric)	N/A
User Defined (5-7 Numeric)	✓	User Defined (5-7 Numeric)	N/A
Deny 4 and 8 Digit PIN	✓	Deny 4 and 8 Digit PIN	N/A
Deny Alphanumeric PIN	✓	Deny Alphanumeric PIN	N/A
Deny Numeric PIN	✓	Deny Numeric PIN	N/A
PIN Reuse	✓	PIN Reuse	N/A
Passcode			
16 Digit Passcode	✓	16 Digit Passcode	N/A
4 Digit Fixed Passcode	✓	4 Digit Fixed Passcode	N/A
Next Tokencode Mode			
Next Tokencode Mode	✓	Next Tokencode Mode	N/A
Load Balancing / Reliability Testing			
Failover (3-10 Replicas)	N/A	Failover	N/A
No RSA Authentication Manager	✓	No RSA Authentication Manager	N/A
Additional Functionality			
RSA Software Token Automation			
System Generated PIN	N/A	System Generated PIN	N/A
User Defined (8 Digit Numeric)	N/A	User Defined (8 Digit Numeric)	N/A
Next Tokencode Mode	N/A	Next Tokencode Mode	N/A
RSA SecurID 800 Token Automation			
System Generated PIN	N/A	System Generated PIN	N/A
User Defined (8 Digit Numeric)	N/A	User Defined (8 Digit Numeric)	N/A
Next Tokencode Mode	N/A	Next Tokencode Mode	N/A

JGS / PAR

✓ = Pass ✗ = Fail N/A = Non-Available Function

‡ Note that this checklist contains the tests for SecurID authentication for the Fischer Identity Suite Self-Service Portal.

Known Issues

Fischer Identity Suite currently does not support RSA Authentication Manager replica servers and failover functionality. This applies to Self-Service Portal RSA SecurID authentication part of the integration.