



ACCELRY'S LAB NOTEBOOK
VERSION 6.8 SP1

Accelrys Deployment Manager for Notebook Administrators

LAST UPDATED: 24 OCTOBER 2013

Copyright Notice

Copyright © 2013 Accelrys Software Inc. All rights reserved.

This product (software and/or documentation) is furnished under a License Agreement and may be used only in accordance with the terms of such agreement.

Trademarks

The registered trademarks or trademarks of Accelrys Software Inc. include but are not limited to:

ACCELRYS®
ACCELRYS & Logo®
PIPELINE PILOT

All other trademarks are the property of their respective owners.

Acknowledgments and References

Accelrys may grant permission to republish or reprint its copyrighted materials. Requests should be submitted to Accelrys Scientific and Technical Support, either through email to support@accelrys.com or in writing to:

Accelrys Scientific and Technical Support
5005 Wateridge Vista Drive
San Diego, CA 92121

Contents

Overview

Mapping Rules, Profiles, and Packages.....	4
Components	4
Release Profiles and Release Packages.....	5
Release Profiles that come with Notebook	6
Release Profile with mapping rules	6
Workflow for Package Customization.....	8
ADM Store Commands – high level view.....	9

Example Tasks

Getting ADM Store Manager command syntax	10
Assigning Users to the Roaming Profile	11
Deploying a Custom Profile.....	13
Creating a New Client Profile	13

Creating the Working and Test Environment.....	13
Creating a New Client Package	16
Get Namespace (one time task)	16
Create the Working and Test Environment.....	17
Create and Test the Custom Package.....	17
Release the Custom Package.....	19
Install the Custom Package to Clients	20
Creating a Login Domain List.....	21

Glossary

Definitions of Key Terms	23
--------------------------------	----

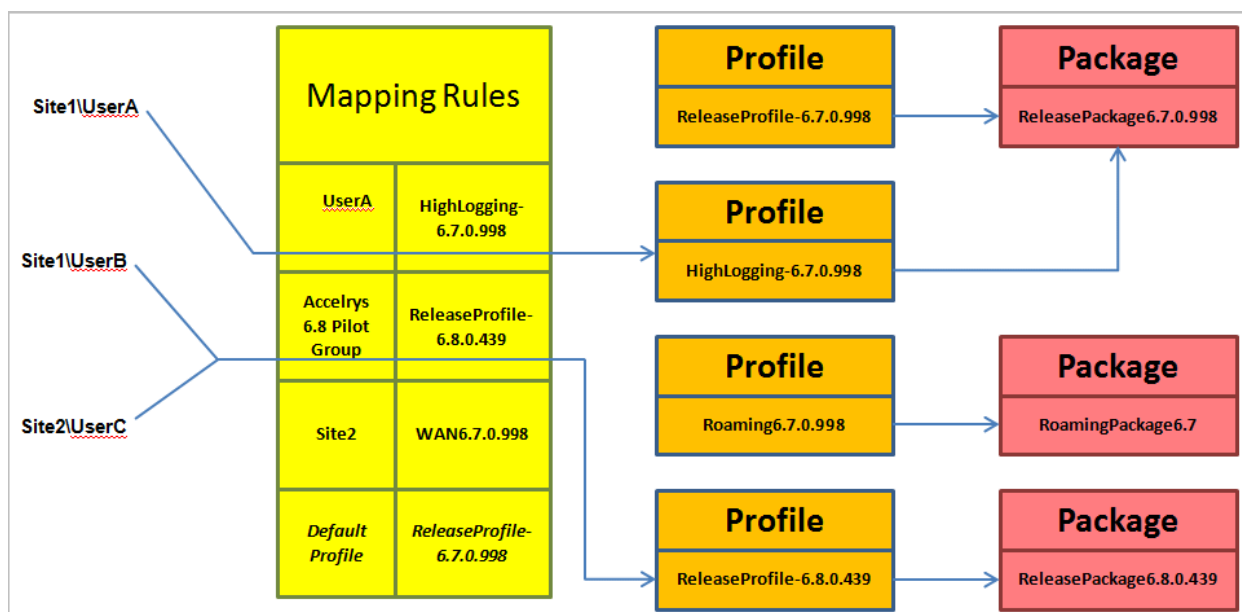
Overview

Accelrys Deployment Manager (ADM) is a feature of Accelrys Lab Notebook (Symyx Notebook) that allows you, the Notebook Administrator, to manage user configurations and deliver customized versions of Notebook to your scientist end-users.

Mapping Rules, Profiles, and Packages

You can use ADM to:

- Assign users to different configurations (Profiles) at the group and/or individual level through Mapping Rules. You can use the default profiles, or create custom profiles.
- Push to each end-user the appropriate Profile, such as whether storage is Roaming, Online, or Offline.
- Push to each end-user, if appropriate, a new version of Notebook binaries, that is, a new release Package.



ADM does not affect the client machine's registry or modify the file system outside the Symyx Technologies application data folder.

Components

ADM Component	Description
ADM Service	<ul style="list-style-type: none">■ A web service that provides the correct configuration (Profile) and the correct set of binaries for a Notebook version (Package) to the users of the ADM Client, that is, the scientist end-users.■ The ADM service URL is http://your-server/accelrys/deployment-manager
ADM Client	The ADM Client is the Symyx Notebook Client Manager "shell" that resides on the scientist end-user client machine. The scientist logs in to the shell, the shell contacts the ADM service, the shell downloads the correct configuration (Profile) and, if appropriate, downloads and unpacks a new set of binaries (Package). Finally, the shell launches Notebook.

You, as the administrator, have command line options to:

- Customize the login screen

- Download a Package.
For example, to avoid making end-users wait for a download to complete when they login, download the Package at night or on a weekend.
- Delete the locally stored copy of a Package.
This cleans up the hard drive, and is the same as deleting Packages from the deployments folder.
- **NotebookClientManager.exe** resides by default in C:\Program Files (x86)\Symyx\Symyx Notebook\bin\

Note: To delete the Notebook client, an end-user can uninstall Symyx Notebook Client Manager using the Windows Control Panel.

ADM Store Manager

A **server-side** command-line tool that enables administrators to:

- Create profiles and packages
- Change the Mapping Rules that associate Profiles to users or groups of users.

The easiest way to control which users get a new release Package is by using the Symyx Vault Administration Console to assign users to the Group that the Profile points to.

- **VaultAdmStoreManager.exe** resides in a subdirectory of the Vault Deployment Utility, and can also reside on the administrator's computer

Release Profiles and Release Packages

ADM administration involves creating Release Profiles and Release Packages.

Profile	Package
<ul style="list-style-type: none"> ■ <i>Configuration</i>, such as the setting for the object cache and enabling or disabling features described in the Notebook Administration Guide, Chapter 1. ■ Also contains a pointer to the Release Package. 	<ul style="list-style-type: none"> ■ <i>Deployment</i>, such as rolling forward to a new version of the Accelry ELN or installing an add-in.
Regarding the <code>Symyx.Notebook.Application.exe.config</code> file ...	
<ul style="list-style-type: none"> ■ Settings within this configuration file but not inside its <runtime> node 	<ul style="list-style-type: none"> ■ Settings inside the <runtime> node of this configuration file
	<ul style="list-style-type: none"> ■ Other files

To work with the `Symyx.Notebook.Application.exe.config` file, see [Creating the Working and Test Environment](#).

Release Profiles that come with Notebook

An installation of Accelrys Vault Server has a set of release profiles created by the Vault Deployment Utility (VDU). You can use these profiles or modify them to suit your needs. For examples of edits to the configuration file that correspond to the features in these profiles, see the Notebook Administration Guide, Chapter 1.

- All profiles will show log in times and will create a `Symyx.Notebook.log` file.
 - For performance logging, use a “PerfLog” Profile, which logs `CheckOutAndOpen`, `CheckIn`, and `ScriptExecution`.
 - Debug profiles log a `OneTimeInitializationTasks` entry that shows initialization of notebook components and any errors, such as a script on a form that is missing an argument.
 - Roaming and Offline profiles show log in times in two files: `Symyx.Notebook.log` and `Login.log`
- For low-level development troubleshooting, use a “Debug” Profile.
- The default, Offline, stores the client’s private repository on the client’s local drive.
- Roaming stores the private repository in Oracle.
- Citrix uses Roaming with the object cache disabled.

Name	User Repository Storage Location	Local Object Caching Enabled	Additional Information
Offline	Client machine	Yes	Default profile
Offline-Debug	Client machine	Yes	Extensive framework level logging
Offline-PerfLog	Client machine	Yes	Performance counters
Roaming	Vault server	Yes	
Roaming-Debug	Vault server	Yes	Extensive framework level logging
Roaming-PerfLog	Vault server	Yes	Performance counters
Citrix	Vault server	No	
Citrix-Debug	Vault server	No	Extensive framework level logging
Citrix-PerfLog	Vault server	No	Performance counters

Release Profile with mapping rules

Example Rule Set

```
<?xml version="1.0" encoding="utf-8"?>
<Rules xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xmlns:xsd=http://www.w3.org/2001/XMLSchema
defaultProfile="V6.7.0.744-Roaming">
<!-- SNIPPED: Help text -->
<Rule userSelector='Debug Logging' profile='V6.7.0.744-Citrix-Debug' />
<Rule userSelector='Citrix Users' profile='V6.7.0.744-Citrix' />
</Rules>
```

Note that `defaultProfile` specifies the profile to be used if no rule applies to the current user, and `userSelector` maps a user or group to a profile.

Mapping Rules associate users and groups to Release Profiles.

- It is possible to specify rules with group or individual scope
- The first rule that applies, regardless of how wide its scope, governs the object
- The *default* release Profile always has lower priority than all other rules

Example using group membership to control logging levels in a Citrix deployment

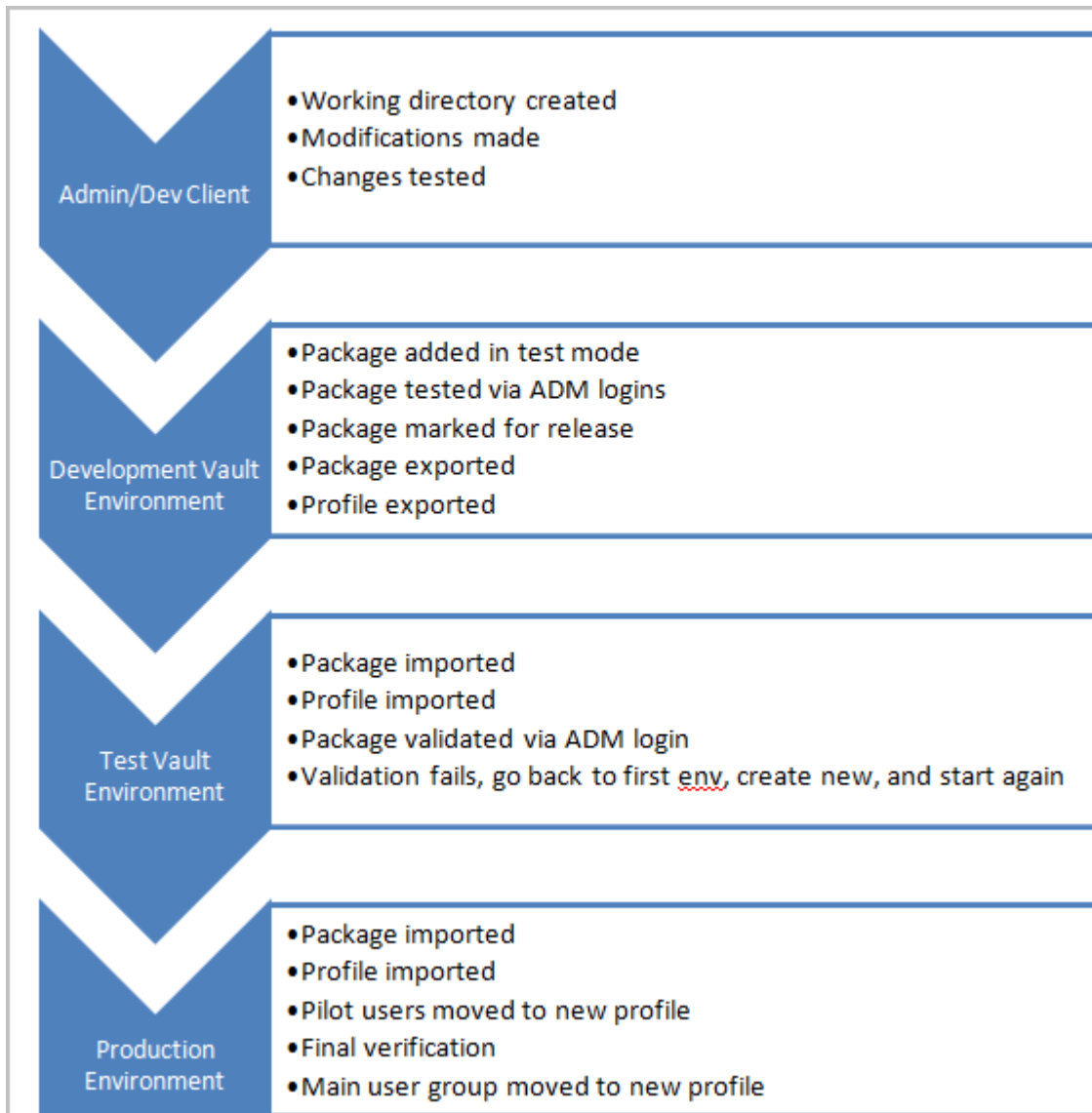
```
<?xml version="1.0" encoding="utf-8"?>
<Rules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
defaultProfile="V6.7.0.744-Citrix">
  <!-- SNIPPED: Help text -->
  <Rule userSelector='ADM Debug Profile' profile='V6.7.0.744-Citrix-Debug' />
  <Rule userSelector='ADM PerfLog Profile' profile='V6.7.0.744-Citrix-PerfLog' />
</Rules>
```

Notes on example

- The default profile is Citrix
- Members of “ADM Debug Profile” will get the debugging profile
- Members of “ADM PerfLog Profile” will get the performance logging profile
- Members of both groups will get the debugging profile because it is the first rule

Workflow for Package Customization

The following depicts the steps in the process of developing a custom release Package.



ADM Store Commands – high level view

- | | |
|--|--|
| <ul style="list-style-type: none">• Rule Set<ul style="list-style-type: none">– download-rules– upload-rules• Profiles<ul style="list-style-type: none">– list-profiles– add-profile– update-profile– change-profile-package• Package Management<ul style="list-style-type: none">– add-package– list-packages– release– update-package | <ul style="list-style-type: none">• Working Directories<ul style="list-style-type: none">– generate– get-folder-status• Package Security<ul style="list-style-type: none">– add-namespace– list-namespaces– create-key– get-public-key• Promotion<ul style="list-style-type: none">– export-profile– import-profile– export-package– import-package |
|--|--|

Example Tasks

Getting ADM Store Manager command syntax

1. On your Vault server, at a command prompt, navigate to the directory in which you installed the Vault Deployment Utility.
2. Navigate to the subdirectory for the ADM Store Manager.
3. Type `VaultADMStoreManager /help`
4. Read the online Help that shows the command syntax.

Usage:

```
VaultADMStoreManager [/help]
```

Get general help and a list of commands.

```
VaultADMStoreManager /help <command>
```

Get specific help on a particular command.

```
VaultADMStoreManager [/nostore | /local <store-folder> | /vault <server> <user>  
<password>] <command>
```

Execute a particular command

Storage Options:

If no option is selected, `/nostore` is assumed.

```
/nostore
```

Do not connect to a local or Vault store.

```
/local <store-folder>
```

Use or create a local store in `<store-folder>`

```
/vault <server> <user> <password>
```

Use the ADM store in the specified vault server and connect with the specified credentials. If `<password>` is an empty string (`""`), you will be prompted for your password on the command-line.

Commands:

<code>add-namespace</code>	Creates a strong namespace.
<code>add-package</code>	Creates a new package and profile.
<code>add-profile</code>	Creates a new release profile using the same package.
<code>change-profile-package</code>	Generates a working directory for a profile with a new package.
<code>create-key*</code>	Creates a key pair.
<code>download-rules</code>	Downloads the user to profile mapping rules into a file.
<code>export-package</code>	Creates a file containing a released package.
<code>export-profile</code>	Creates a file containing a release profile.
<code>generate</code>	Generates a working directory.
<code>get-folder-status*</code>	Outputs the state of a working directory.

<code>get-public-key*</code>	Gets the public key from key pair file and saves it to another file.
<code>help*</code>	Displays this message or describes a command.
<code>import-package</code>	Creates a package based on the contents of a package export file.
<code>import-profile</code>	Creates a profile based on the contents of a package export file.
<code>list-namespaces</code>	Outputs a line-delimited list of package namespaces.
<code>list-packages</code>	Outputs a line-delimited list of release packages.
<code>list-profiles</code>	Outputs a line-delimited list of release profiles.
<code>release</code>	Marks an unreleased package as released.
<code>update-package</code>	Updates a TEST package from a working directory. Only TEST packages can be updated, not release packages.
<code>update-profile</code>	Updates a release profile from a working directory.
<code>upload-rules</code>	Uploads the user to profile mapping rules from a file.

* = Does not require a release store to function.

To get help for a specific command, such as the `download-rules` command, do as follows:

```
VaultADMStoreManager /help download-rules
```

which displays the following details:

```
Command: download-rules
Copies the user to profile mapping rules out of the specified storage
mechanism and into the specified file. The target file is overwritten
without warning.

Usage:
... download-rules <file-name>

Examples:
... download-rules rules.xml
Gets mapping rules out of the specified store and saves them in rules.xml.
```

Assigning Users to the Roaming Profile

If your organization has a site preference for the Private Repository to use Roaming storage, modify the ADM default Rules as follows.

1. On your Vault server, navigate to the subdirectory with the Vault Store Manager. For example:

```
cd C:\VaultDeploymentUtility\VaultADMStoreManager
```

where `C:\VaultDeploymentUtility` represents the location to which the Vault Deployment Utility was unzipped.

2. Make sure that the users have checked in all their documents
3. Get a list of the available Profiles by running the following command with a Vault administrator account.

```
VaultADMStoreManager /vault server name domain\username "" list-profiles
```

Note: Using the double-quotes after the username causes the system to prompt for the password. The other option is to type the password instead of " "

Example:

```
VaultADMStoreManager /vault vm-avs67 vm-avs67\vault.admin "" list-profiles
```

```
Enter password: *****
```

Output:

```
V6.7.0.744-Offline  
V6.7.0.744-Offline-Debug  
V6.7.0.744-Offline-PerfLog  
V6.7.0.744-Roaming  
V6.7.0.744-Roaming-Debug  
V6.7.0.744-Roaming-PerfLog  
V6.7.0.744-Citrix  
V6.7.0.744-Citrix-Debug  
V6.7.0.744-Citrix-PerfLog
```

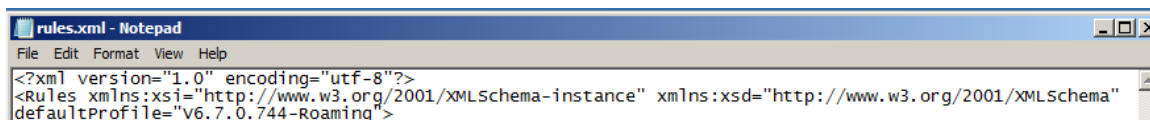
4. Download the mapping rules file from the server.

```
VaultADMStoreManager /vault server name domain\username "" download-rules  
C:\Temp\rules.xml
```

Tip: To minimize retyping, use the up arrow to cycle through the previously entered commands.

5. Edit the mapping rules file, C:\Temp\rules.xml, using Notepad to specify the default profile.

For example, if your Notebook version happened to be 6.7, you might change
defaultProfile="V6.7.0.744-Offline">
to
defaultProfile="V6.7.0.744-Roaming">



6. Upload the newly edited mapping rules file to your Vault server.

```
VaultADMStoreManager /vault server name domain\username ""  
upload-rules C:\Temp\rules.xml
```

7. Inform your end-users that they can log in to Symyx Notebook. When each end-user logs in, the Roaming profile automatically reconfigures the client.

Deploying a Custom Profile

Suppose that Accelrys has released a new version of Notebook in a zip file and you want to deploy it to at least one group of users. Here we assume that you want to use your existing customized profiles for the new release.

- 1 Publish the zip file with a **Release Package**. For example, `NewRelease.admpkg` to put the new package in your vault environment:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> import-package  
NewRelease.admpkg
```

- 2 Generate a *working directory* that represents the change, test it, and then update your profile.

To generate a working directory that represents a **Release Profile** having been assigned to a new **Release Package**, use the `change-profile-package` command.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> change-profile-package  
YourProfileName System ReleasePackage6.8.0.700 C:\Temp\MoveProfileTo68
```

- 3 In the newly created working folder, represented here as `C:\Temp\MoveProfileTo68`, modify the configuration files as needed.
- 4 Test your changes by double-clicking on the `Symyx.Notebook.Application.exe` file to start the Notebook Client.
- 5 When the configuration works the way you want, update the profile.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> update-profile  
C:\Temp\MoveProfileTo68
```

The result is that each user who is assigned to that `profileName` automatically downloads `ReleasePackage6.8.0.700` the next time they log in to Notebook.

Creating a New Client Profile

Suppose that your organization, which we will call `My Company Name`, wants to create a customized Notebook Profile to push out to its users. This would be accomplished by creating a **working directory** based on an existing package and profile, modifying its configuration files, then adding a new **Release Profile** based on that working directory.

Creating the Working and Test Environment

Build your new **Release Profile (or altering and existing Profile)**.

1. Create a working directory.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> generate <profile>  
<directory>
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 generate  
V6.7.0.788-Test C:\TEMP\Rules_package
```

- where `abc123` represents the password,
- `V6.7.0.788-Test` represents the name of an existing profile, and
- `C:\TEMP\Rules_package` represents the working directory.

ADM creates a working directory if it does not already exist.

1. In the working directory, modify the configuration file, `Symyx.Notebook.Application.exe.config`, as needed.

- For examples of configuration changes you might want to make, see the Notebook Administration Guide, Chapter 1.

2. Test your changes by double-clicking on `Symyx.Notebook.Application.exe` to start the Notebook Client.
3. When you are satisfied with your changes, add the new profile:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> add-profile  
NewProfileName <NewPackageDir>
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 add-profile  
MyPackage1.V6.7.0.788-Roaming C:\TEMP\Rules_package
```

Or update an existing profile

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> update-profile  
C:\TEMP\Rules_package
```

If you are altering an *existing* Profile, you are done.

4. Download the current mapping rules, if needed.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> download-rules  
<directory\filename>
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 download-rules  
C:\temp\rules.xml
```

5. List the available profiles.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> list-profiles
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 list-profiles
```

Example Results:

```
V6.7.0.744-Offline  
V6.7.0.744-Offline-Debug  
V6.7.0.744-Offline-PerfLog  
V6.7.0.744-Roaming  
V6.7.0.744-Roaming-Debug  
V6.7.0.744-Roaming-PerfLog  
V6.7.0.744-Citrix  
V6.7.0.744-Citrix-Debug  
V6.7.0.744-Citrix-PerfLog  
V6.7.0.788-Offline  
V6.7.0.788-Offline-Debug  
V6.7.0.788-Offline-PerfLog  
V6.7.0.788-Roaming  
V6.7.0.788-Roaming-Debug  
V6.7.0.788-Roaming-PerfLog  
V6.7.0.788-Citrix  
V6.7.0.788-Citrix-Debug  
V6.7.0.788-Citrix-PerfLog  
MyPackage1.V6.7.0.788-Roaming
```


6. Update the mapping rules.

Example:

```
<?xml version="1.0" encoding="utf-8"?>
  <Rules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema"
        defaultProfile="V6.7.0.788-Roaming">
  </Rules>
```

Example Results:

```
<?xml version="1.0" encoding="utf-8"?>
  <Rules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema"
        defaultProfile="MyPackage1.V6.7.0.788-Roaming">
  </Rules>
```

7. Upload the changed rules for testers.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> upload-rules
  <directory>\<filename>
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 upload-rules
  C:\temp\rules.xml
```

Creating a New Client Package

Suppose that your organization, which we will call `My Company Name`, wants to create a customized version of Notebook to push out to its users. This would be accomplished by creating a **working directory** based on an existing package and profile, modifying its contents, then creating a new **Release Package** and a new **Release Profile** based on that working directory.

Get Namespace (one time task)

1. If `My Company Name` does not already have a namespace for packages you create, you will need to create one. You only need to do this one time for your organization. You will need to pass the key files around to create one namespace. Create a new public/private key pair:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> create-key
MyOrganization.keypair
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 create-key
C:\Temp\MYCompany.keypair
```

The newly-created `MYCompany.keypair` file has a public and private key for signing packages, and a public key for validating them. Store the key pair in a secure location because you need to use it whenever you want to create a **Release Package**.

2. Generate a public-key-only file:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> get-public-key
MyOrganization.keypair MyOrganization.publickey
```


Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 get-public-key  
C:\Temp\MYCompany.keypair C:\Temp\MYCompany.publickey
```

3. Put the public key file where you can easily access it because you need it each time you create your organization's package namespace in new Vault environments.

The key-pair files are very important and must be reused for each usage of that namespace in the development process.

4. Set up a namespace on your test or development server:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> add-namespace  
MyOrganization MyOrganization.publickey
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 add-namespace  
MYCompany C:\Temp\My Company Name.publickey
```

This adds a new package namespace to your Vault server and associates it with the public key you provided. Packages not signed with the corresponding key pair will be rejected by Vault. This allows you to create as many packages as you want with no risk of colliding with packages released by Accelrys or other parties.

Create the Working and Test Environment

Build your new **Release Package**.

1. Create a working directory.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> generate <profile>  
<directory>
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 generate  
V6.7.0.788-Roaming C:\TEMP\Rules_package
```

ADM creates a working directory if it does not already exist.

2. Modify the working directory by adding DLLs, files, or other needed changes to it.
3. Test the new package by double-clicking the `Symyx.Notebook.Application.exe` file to start the Notebook Client.

Create and Test the Custom Package

1. When you are satisfied that it works the way you want it to work, add the new package and profile:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> add-package  
NewProfileName <namespace> <NewPackageName> <MyOrganization.keypair>  
<NewPackageDir>
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 add-package  
MyPackage1.V6.7.0.788-Roaming MYCompany MyPackage1.V6.7.0.788-Roaming  
C:\Temp\MYCompany.Keypair C:\TEMP\Rules_package
```

The newly created package is automatically in **test mode**.

Packages in test mode are refreshed by clients each time they log in to Notebook. This allows them to be updated and prevents them from being exported to another server. (Only TEST packages can be updated, not release packages.)

2. Download the current mapping rules.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> download-rules <directory>\filename>
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 download-rules  
C:\temp\rules.xml
```

3. List the available profiles.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> list-profiles
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 list-profiles
```

Example Results:

```
V6.7.0.744-Offline  
V6.7.0.744-Offline-Debug  
V6.7.0.744-Offline-PerfLog  
V6.7.0.744-Roaming  
V6.7.0.744-Roaming-Debug  
V6.7.0.744-Roaming-PerfLog  
V6.7.0.744-Citrix  
V6.7.0.744-Citrix-Debug  
V6.7.0.744-Citrix-PerfLog  
V6.7.0.788-Offline  
V6.7.0.788-Offline-Debug  
V6.7.0.788-Offline-PerfLog  
V6.7.0.788-Roaming  
V6.7.0.788-Roaming-Debug  
V6.7.0.788-Roaming-PerfLog  
V6.7.0.788-Citrix  
V6.7.0.788-Citrix-Debug  
V6.7.0.788-Citrix-PerfLog  
MyPackage1.V6.7.0.788-Roaming
```

4. Update the mapping rules.

Example:

```
<?xml version="1.0" encoding="utf-8"?>  
  
  <Rules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
  
    defaultProfile="V6.7.0.788-Roaming">  
  
  </Rules>
```

Example Results:

```
<?xml version="1.0" encoding="utf-8"?>  
  
  <Rules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
  
    defaultProfile="MyPackage1.V6.7.0.788-Roaming">  
  
  </Rules>
```

5. Upload the changed rules for testers.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> upload-rules  
    <directory\filename>
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 upload-rules  
    C:\temp\rules.xml
```

6. Ask the testers to be a pilot group to test the new **Release Profile** and **Release Package**.

Release the Custom Package

7. When you are ready to promote the Profile and Package to a validation environment, mark the package released. For example:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> release  
    MyOrganization NewPackageName
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 release MYCompany  
    MyPackage1.V6.7.0.788-Roaming
```

When a package is in release mode, it is read-only. Its content is safe to export and import into other Vault environments. A **Release Package** can be exported using the `export-package` command and a Release Profile can be exported with the `export-profile` command.

8. Create portable Package and Profile for installation on other machines

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> export-package  
    MyOrganization PackageName package-file
```

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> export-profile  
    NewProfileName profile-file
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 export-package  
    MYCompany MyPackage1.V6.7.0.788-Roaming C:\Temp\MYCompany
```

Those assets can then be promoted through your validation environments, and ultimately released to the production environment.

9. Create zip file for transport to machines

- a. C:\Temp\MYCompany.admprof
- b. C:\Temp\MYCompany.admpkg
- c. MYCompany.publickey

10.

Install the Custom Package to Clients

Suppose that My Company Name has released this new version of Notebook in a zip file and you want to use your customized profiles to deploy it to at least one group of users.

1. If the MYCompany namespace has not been initialized, set up a namespace on your test or development server:

To check, run the following command and look for MYCompany in the returned list:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> list-namespaces
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 list-namespaces
```

If the namespace is not found, add the namespace:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> add-namespace  
MyOrganization MyOrganization.publickey
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 add-namespace  
MYCompany C:\Temp\My Company Name.publickey
```

This adds a new package namespace to your Vault server and associates it with the public key you provided. Packages not signed with the corresponding key pair will be rejected by Vault. This allows you to create as many packages as you want with no risk of colliding with packages released by Accelrys or other parties.

2. Publish the new **Release Package** from the zip file:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> import-package  
<directory\filename>.admpkg
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 import-package  
C:\Temp\MYCompany.admpkg
```

3. Publish the new **Release Profile** from the zip file:

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> import-profile  
<directory\profilefilename>
```

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 import-profile  
C:\Temp\MYCompany.admprof
```

4. Download the current rules.

```
VaultADMStoreManager /vault <server> <domain>\<user> <password> download-rules  
<directory\filename>
```

Example:

```
VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 download-rules  
C:\temp\rules.xml
```

5. Edit the current rules.

Example:

```
<?xml version="1.0" encoding="utf-8"?>
```

```

    <Rules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"

    defaultProfile="V6.7.0.788-Roaming">

</Rules>

```

Example Results:

```

<?xml version="1.0" encoding="utf-8"?>

    <Rules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"

    defaultProfile="MyPackage1.V6.7.0.788-Roaming">

</Rules>

```

6. Upload the changed rules for users

```

VaultADMStoreManager /vault <server> <domain>\<user> <password> upload-rules
    <directory>\filename>

```

Example:

```

VaultADMStoreManager /vault vm-avs66 vm-avs66\vault.admin abc123 upload-rules
    C:\temp\rules.xml

```

The result is that each user who is assigned to the new profile automatically downloads the new package when logging in to Notebook.

Creating a Login Domain List

In this scenario, we create a drop-down list of connection machines and limit the end-user clients to those machines. The commands must be run on each machine separately.

1. Edit an XML file to list the possible Domains and servers.
2. In a command window, set the default directory to the folder where **NotebookClientManager.exe** resides

```

cd <directory>

```

Example:

```

C:\Program Files (x86)\Symyx\Symyx Notebook\bin\

```

3. Create the XML file, adding values to the username, domain, and server tags.

```

<?xml version="1.0" encoding="utf-16"?>
<MostRecentlyUsed xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <DisallowUserEnteredServer>true</DisallowUserEnteredServer>
    <IsLocked>true</IsLocked>
    <UserName>
        <string>SomeUser</string>
    </UserName>
    <Domains>
        <string>SomeDomain</string>
    </Domains>
    <Servers>
        <string>SomeServer</string>
    </Servers>
</MostRecentlyUsed>

```

4. Set MRUs for client. (See "MRU" in the [Glossary](#).)

```
NotebookClientManager.exe /install-mrus <Fully Qualified Path to the XML file>
```

Example:

```
NotebookClientManager.exe /install-mrus C:\Temp\MRUS.XML
```

Glossary

Definitions of Key Terms

ADM

Accelrys Deployment Manager, a configuration and deployment solution that includes three components: ADM Client, ADM Service, and ADM Store Manager.

ADM Client

Symyx Notebook Client Manager “shell” resides on the scientist end-user **client** machine and launches Notebook.

ADM Service

A web service that provides the correct configuration (profile) and the correct version binaries (package) to the user of the ADM Client, that is, the scientist end-user. The ADM service URL is <http://your-server/accelrys/deployment-manager>

ADM Store Manager

A **server-side** command-line tool that enables administrators to set up user configurations (profiles) and customized versions (packages) of Notebook.

keypair

If your organization decides to create custom packages, it will need to create a custom namespace. To create a custom namespace, your organization needs a public key and a private key. They are the keypair.

mapping rules

The profile mapping rules file associate (or map) a Vault user or group to a Release Profile. The RULES.XML file is created by the ADM Store Manager in the specified folder. The profiles are listed in descending order of precedence. For example, if the rules say:

```
<?xml version="1.0" encoding="utf-8" ?>
<Rules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" defaultProfile="V6.7.0.788-Offline-Draw41">
  <Rule userSelector='TestUser' profile='V6.7.0.788-Offline' />
  <Rule userSelector='SlowGroup' profile='V6.7.0.744-Offline' />
</Rules>
```

then TestUser uses 'V6.7.0.788-Offline' regardless of whether or not he is part of the SlowGroup.

“defaultProfile” rule has no user/group selector, no priority and must always point to a valid Release Profile.

MRU

Means “most recently used”. The MRUs represent login data, such as most recent used username, domain, server.

You can get the MRU for the

- **machine**
- **user**
- **both** machine and user
- **default**
- **example** of dummy values

namespace

A namespace provides a way to have a uniquely named set of release packages. The default package namespace for releases from Accelrys is named `System`.

A non-default namespace can be created by Accelrys Professional Services or if you want to create a custom package for Notebook.

package

A release package is a set of files for a specific release of Notebook, for example, `ReleasePackage6.7.0.744`

Multiple profiles can point to a single package.

Each time you install a service pack, hotfix, or update, you will receive a new package, and each package has a unique number. A machine might have many packages, but at any given time, it will use one package.

profile

A release profile associates a profile to a package and configuration files. For example, there might be a `BetaTester` profile for a new package, while a `Production` profile continues to point to an older package.