CyberArk + Talend

CyberArk Integration with Talend

Contents

Date	02/10/2017

CyberArk + Talend

Objective/Summary

CyberArk is an information security company offering Privileged Account Security, is designed to discover, secure, rotate and control access to privileged account passwords used to access systems throughout the enterprise IT environment. The company's technology is utilized primarily in the financial services, energy, retail and healthcare markets.

CyberArk eliminates hard-coded passwords and locally stored SSH keys from applications, service accounts and scripts with no impact on application performance. In order to setup/configure/retrieve credentials from CyberArk, there are a series of steps that needs to be followed to setup communication between CyberArk and application. CyberArk also provides a set of APIs to access the credentials from the CyberArk Vault.

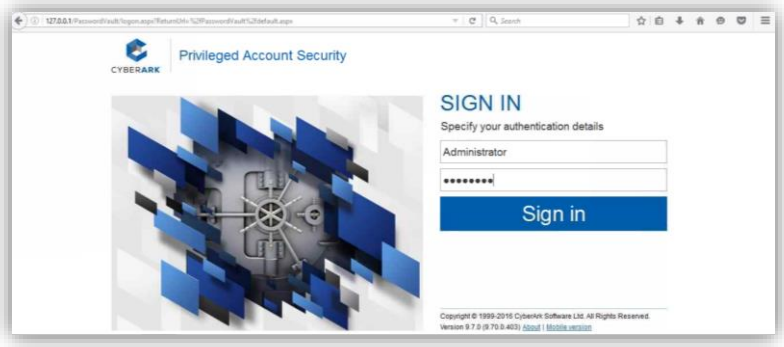
The aim of this document is to cover how do we setup credentials in credential manager at CyberArk, what are the steps to be performed, how it can be tested using a set of CyberArk commands and also above all how can we leverage Talend to communicate with CyberArk using their APIs – authenticate, retrieve credentials, get them and assign it to context variables at runtime during execution of job.

Creating a Safe

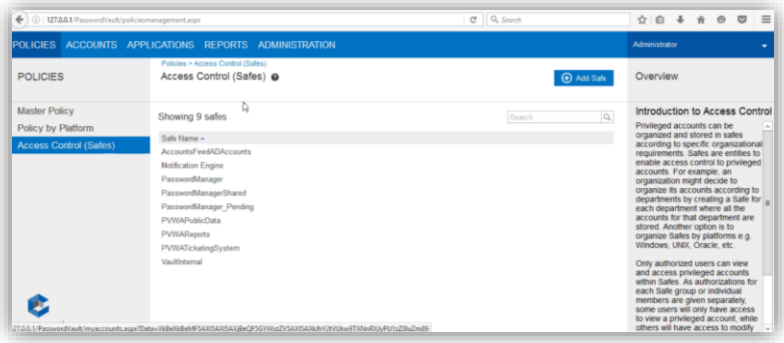
A safe is a container for storing passwords. Safes are typically created based on who will need access to the privileged accounts whose passwords will be stored within the safe. For instance, you might create a safe for a business unit or for a group of administrators. The safes are collectively referred to as the vault.

You will need a CyberArk safe to store objects. But at a customer site will have all their accounts spread to as many safes as they need to.

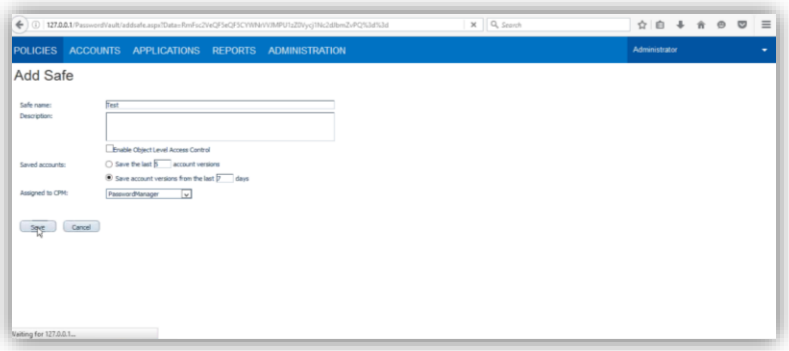
Login to CyberArk with your credentials.



Navigate to Policies -> Access Controls(Safes) -> Click on “Add Safe”.



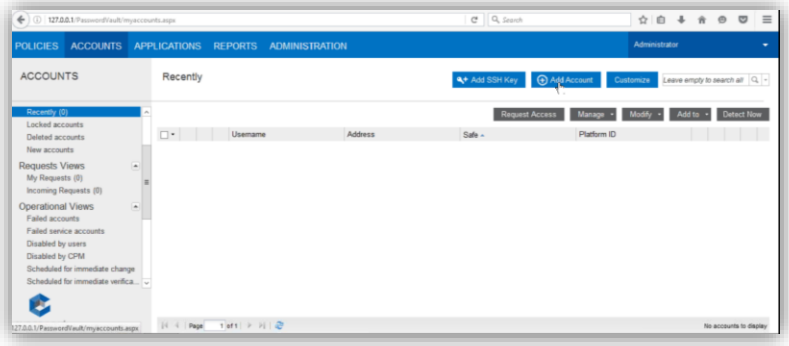
Create a safe.



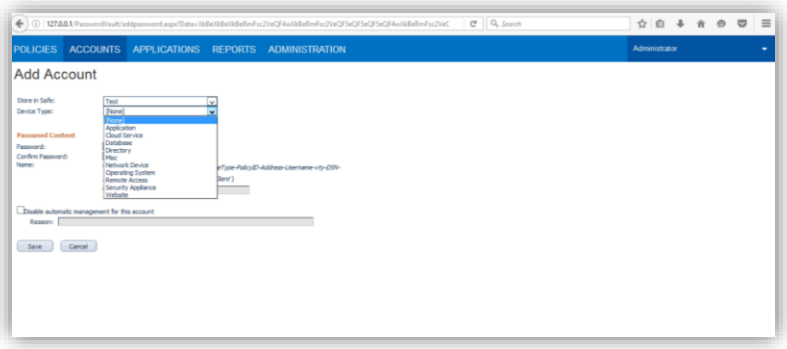
Since that safe is now created now we need a couple of accounts.

Creating an Account

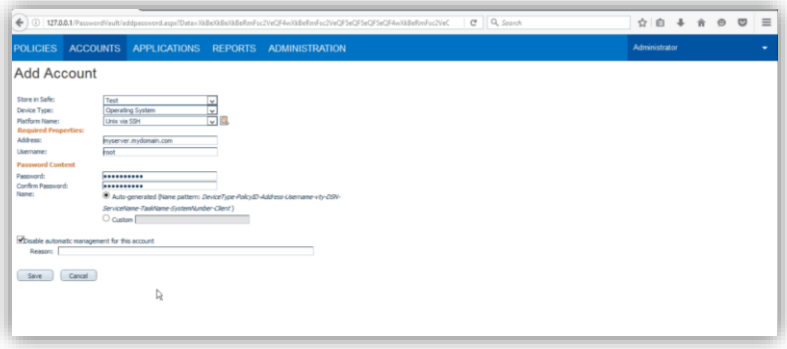
Navigate to Accounts and click on “Add Account”



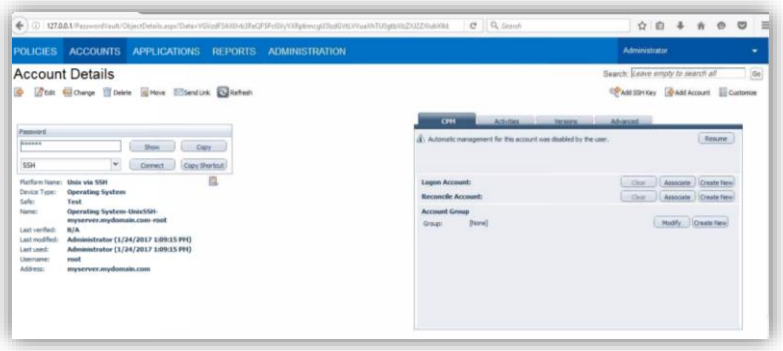
Typically, customer can create accounts of different types, list is in the screen here



Fill in the account details as below



Basically customer will give FQDN name (in this demo is dummy account)



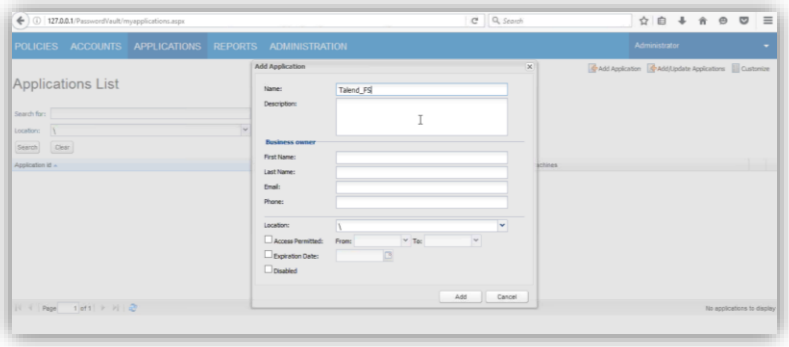
Now that account is created, we need to create an application.

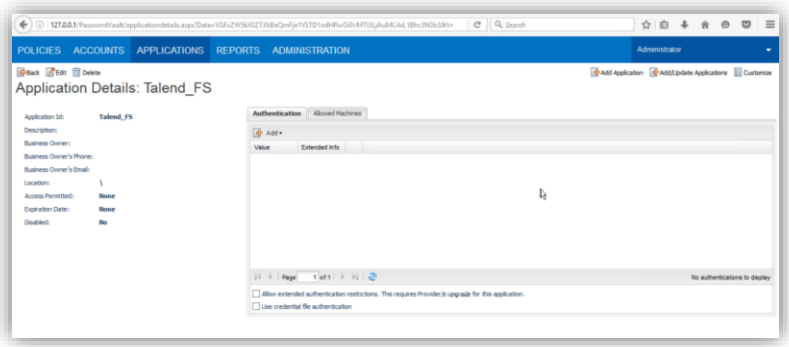
Creating Application

An application is what do you use in order to retrieve credential from the vault. Applications connect to CyberArk using their application id and application characteristics. Application characteristics are additional authentication factors the applications created in the CyberArk.

Each application should have a unique application id (application name). We can change it later but it may also require code change on the client side where this application is used to get password. In our case it's Talend Code.

Navigate to Applications and click on “Add Application”.

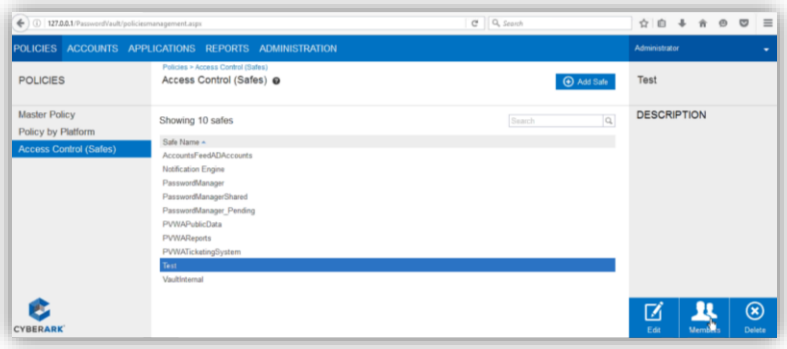




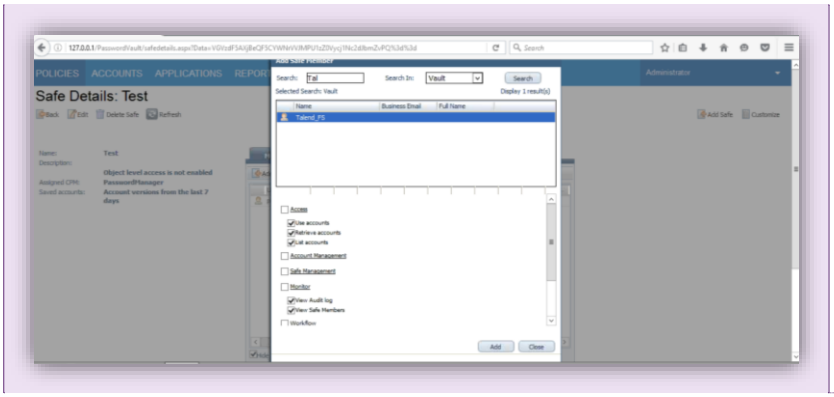
Now we have CyberArk – account and application id. Now give permissions to application id to retrieve credentials. Click on the **Allowed Machines** tab and enter the IP’s of the servers from where Talend will retrieve credentials.

Access to Applications from Safe

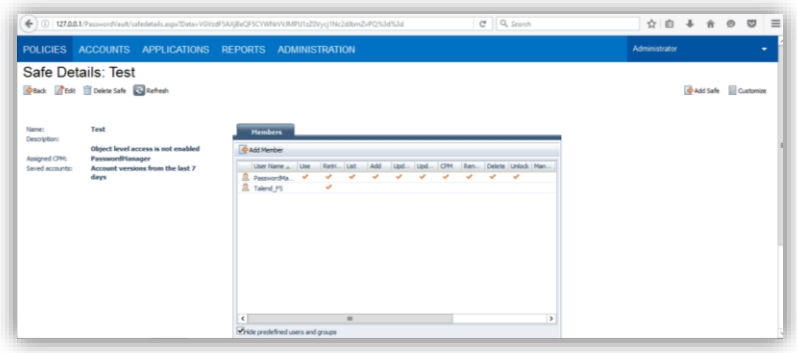
Navigate to policies -> access controls(safe) -> select your safe and click on members



Click on add member and search for your application, select it and check appropriate accesses and add it.



Commented [EA1]: The Application ID permissions are wrong. An Application only needs Retrieve Accounts permission. Please correct this screenshot.

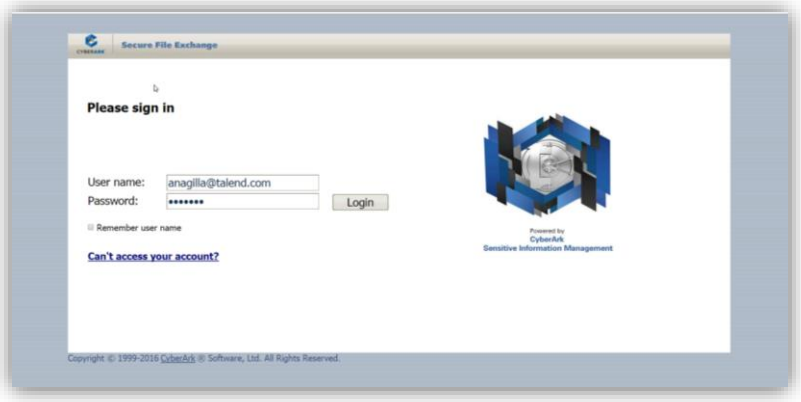


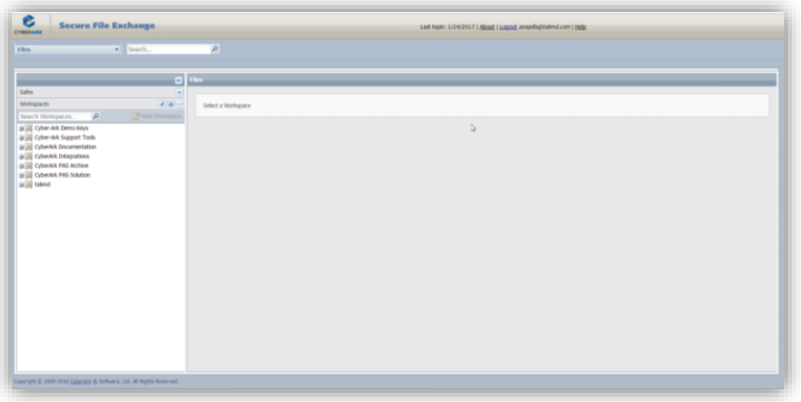
Now the next step is to install the credential provider in the development environment.

Installing credential provider (CP)

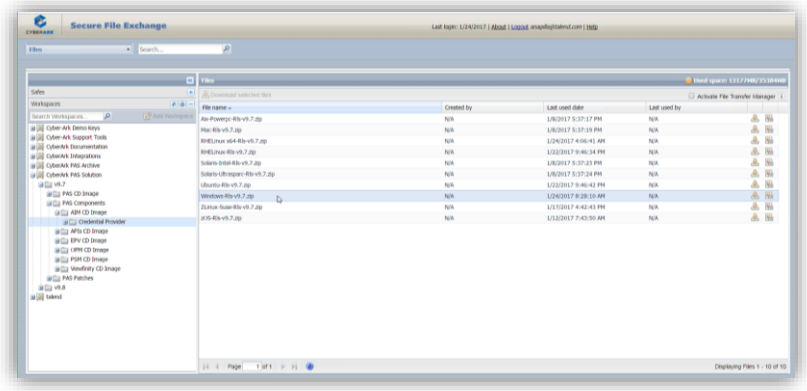
In order to retrieve credentials, we need to install a CyberArk module in the same box as your application (client) is running. This will deliver a java API that will call the credential provider and talks to your application through Java API and then talks to CyberArk vault through their own proprietary protocol and retrieve the credentials that you need and then delivers back to your application through the java API.

You will need to login to the CyberArk Support Vault in order to download the Credential Provider.

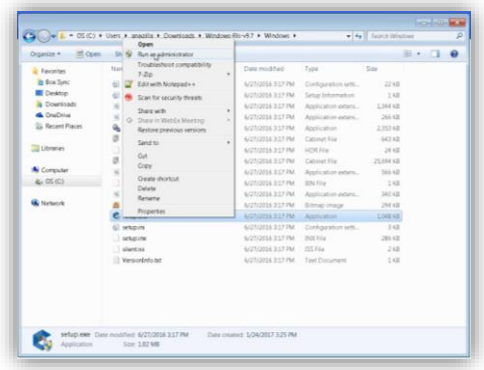




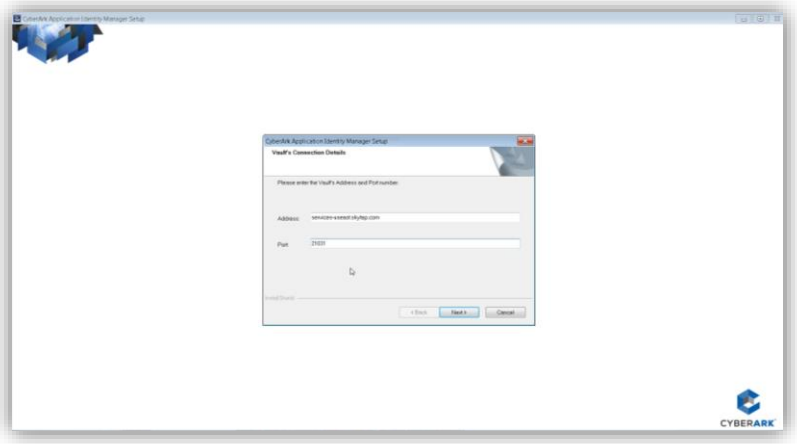
Navigate to CyberArk PAS solution -> v9.7 -> AIM CD Image -> Credential Provider -> select a platform specific CP and install it. In this case its “Windows-Rls-v9.7.zip”



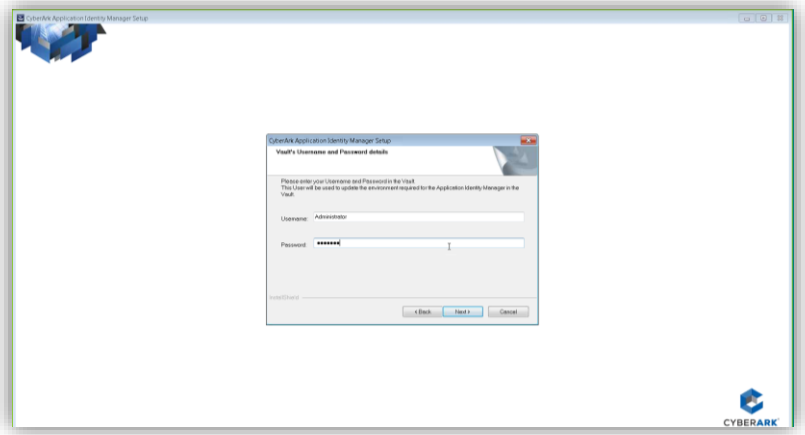
Once the download is finished, right-click on setup.exe and run as administrator

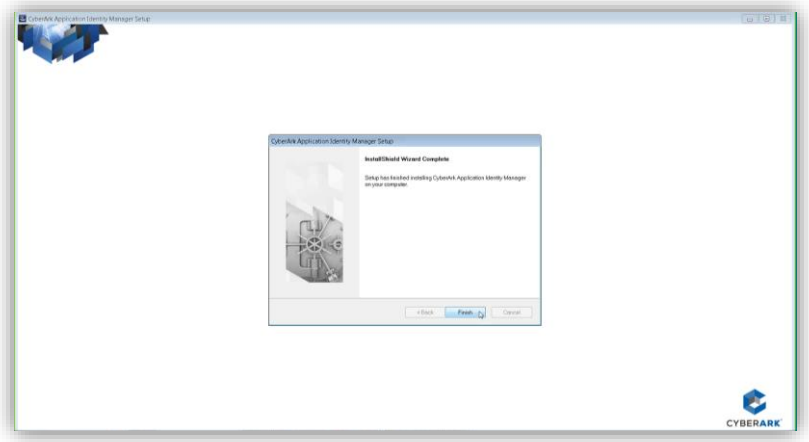


On the vault's connection details page, give address and port.

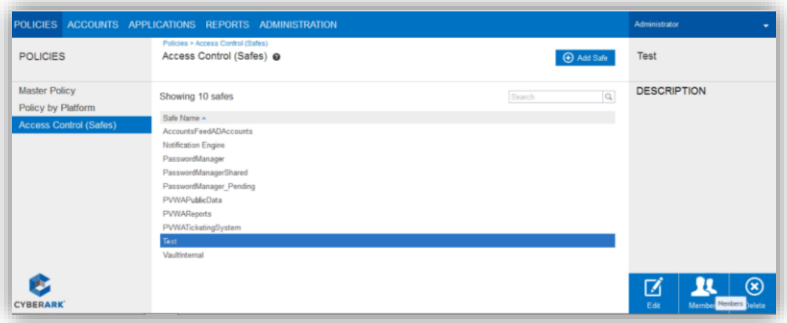


Enter a vault admin user and password

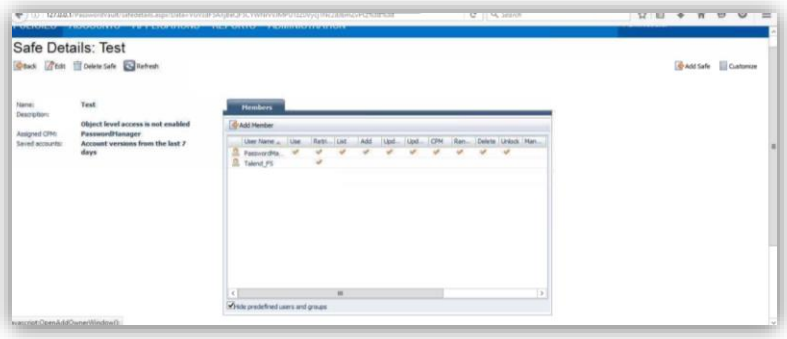




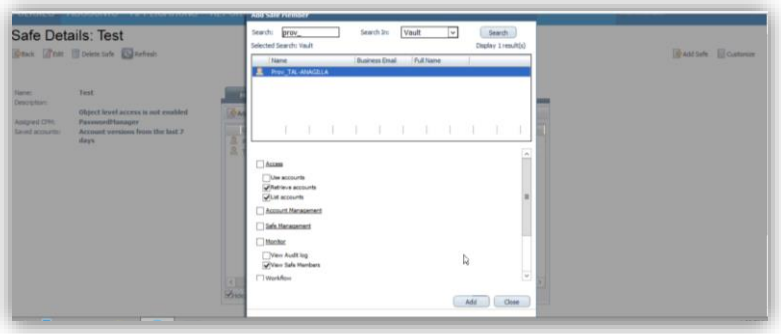
Now since the provider is installed it creates a user that we need to authorize to be a member of this safe. Login to CyberArk again and navigate to Policies -> Access Controls(Safes) -> Select “Test” and click on “Members”

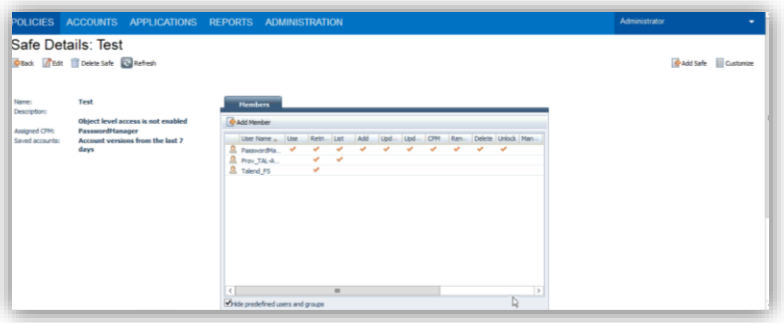


You will see below screen



Click on add member. All provider users are called as “prov_” + name of the host where they are installed. Select the “Retrieve Accounts”, “List Accounts”, and “View Safe Members” privileges. Then click “Add”.

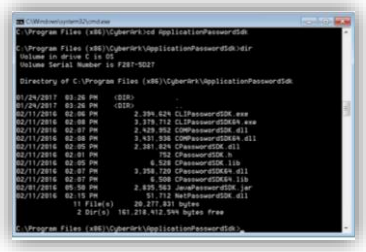




Now that provider has privileges to access vault, open command prompt from your job server where you installed CP and navigate to path where the CyberArk Credential Provider is installed.

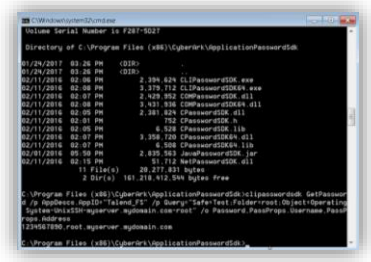
In this demo, path -> C:\Program Files (x86)\CyberArk\ApplicationPasswordSdk

Jar file name is -> name JavaPasswordSDK.jar



Now type the password retrieval command.

clipasswordsdk GetPassword /p AppDescs.AppID="Talend_FS" /p Query="Safe=Test;Folder=root;Object=Operating System-UnixSSH-myserver.mydomain.com-root" /o Password,PassProps.Username,PassProps.Address



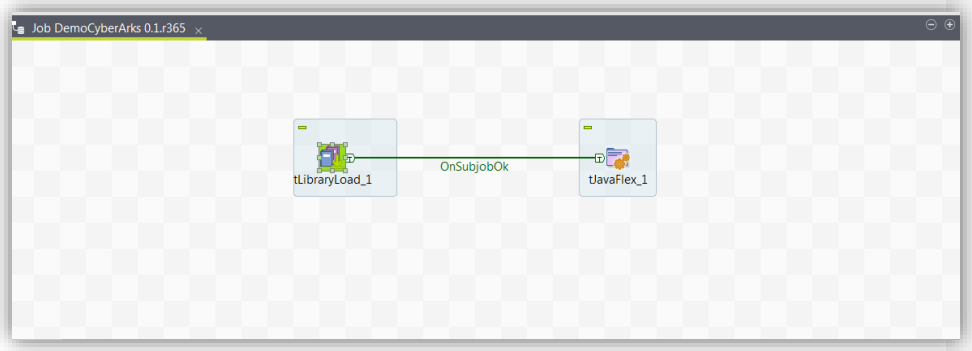
It went to the vault, it authenticated itself and retrieved the password for the root account on myserver.mydomain.com. (It's a dummy account created for demo).

This will end setup from CyberArk's end.

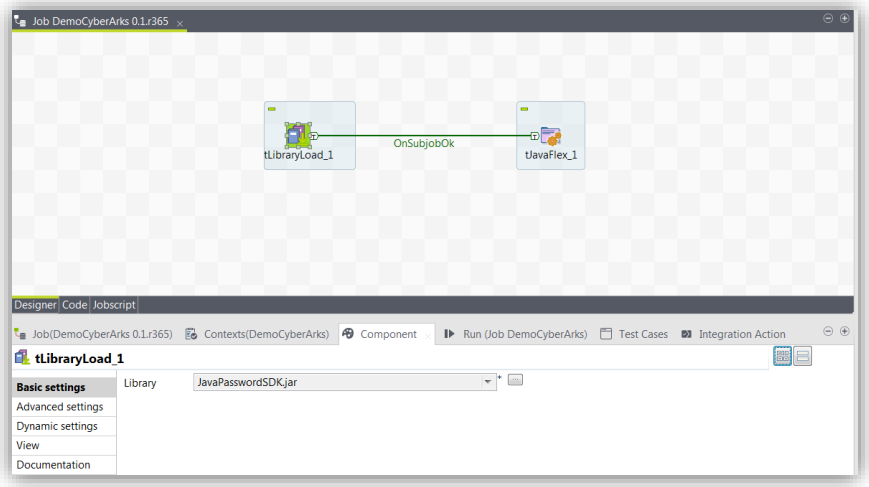
CyberArk + Talend

Retrieve password from Talend using Java API

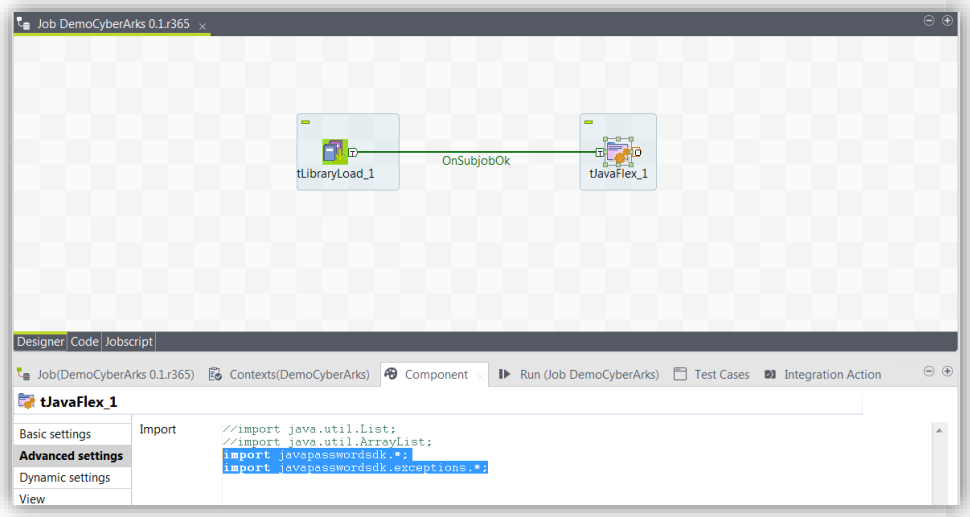
Create a Talend job with tLibrary_Load and a tjavaFlex.



Configure tLibraryLoad to “JavaPasswordSDK.jar” path. This is make sure that “JavaPasswordSDK.jar” is added to classpath by Talend during compilation.



And on tJavaFlex, navigate to advanced settings and make sure you import necessary classes for implementation.



In the basic setting of the tjavaFlex, below code is written to call CyberArk using Java API.

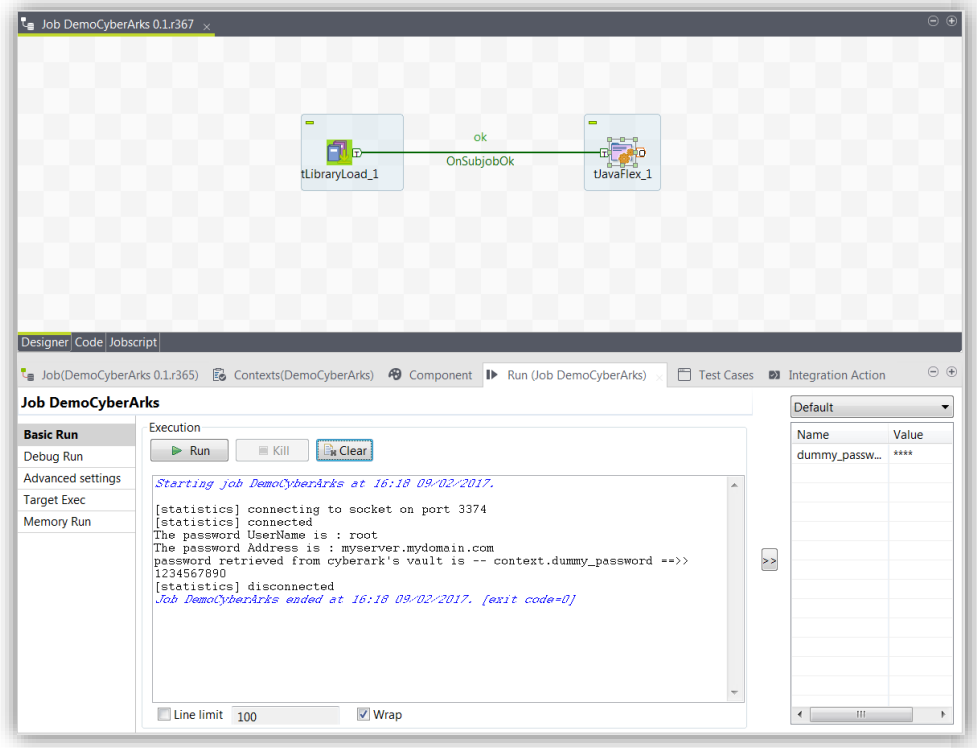
```
try
{
PSDKPasswordRequest passRequest = new PSDKPasswordRequest ();
PSDKPassword password = null;

passRequest.setAppID ("Talend_FS");
passRequest.setSafe ("Test");
passRequest.setFolder ("root");
passRequest.setObject ("Operating System-UnixSSH-myserver.mydomain.com-root");
passRequest.setReason ("This is a demo job for password retrival");


// Sending the request to get the password
password = javapasswordsdk.PasswordSDK.getPassword (passRequest);
// Analyzing the response
System.out.println ("The password UserName is : " +password.getUserName ());
System.out.println ("The password Address is : " +password.getAddress ());
context.dummy_password= password.getContent ();
System.out.println("password retrieved from cyberark's vault is --
context.dummy_password ==>> "+context.dummy_password);

}
catch (PSDKException ex)
{
System.out.println (ex.toString ());
}
```

Save the job and execute it.



It went to the vault, it authenticated itself and retrieved the password for the root account on myserver.mydomain.com. (It's a dummy account created for demo).