



CYBERARK SECRETS MANAGER FOR URBANCODE DEPLOY

Integration Guide



1 URBANCODE-CONJUR-AIM

A plugin which allows UrbanCode Deploy to get credentials from CyberArk Privileged Access Security/Privileged Cloud via CyberArk Secrets Manager, for CI/CD workflows

2 COMPILING SOURCE

gradle

3 INSTALLATION

- Login to UrbanCode Deploy web portal and go to "Settings > Automation Plugins"
- Click "Load Plugin". Select the compiled plugin file and click "Submit"
- A new plugin named "CyberArk" is installed

4 USAGE

The plugin can be used in process designer of "Process" & "Components", by dragging from the left menu area under "Security > CyberArk"

4.1 Secrets Manager Credential Providers

4.1.1 Get Password from Vault

This function allows credentials to be retrieve from Vault server via Secrets Manager Credential Provider.

4.1.1.1 Input Fields

Name	Description	Example
Name	Name of the step	Get Password from Vault
Path	Absolute file path to clipasswordsdk	/opt/CARKaim/sdk/clipasswordsdk
Safe	Safe of the credential stored	DevOps
Folder	Folder of the credential stored	Root
Object	Name of the credential object	Website-Conjur-httpseval.conjur.org-cf-

		spring-app-01
AppID	AppID defined in PVWA	UCD
Output Property - Password	Property for storing retrieved credential	CyberArk/Vault/Password
Output Property - User Name	Property for storing retrieved username	CyberArk/Vault/User
Output Property - Address	Property for storing retrieved address	CyberArk/Vault/Address

4.1.1.2 Output Fields

Name	Description
<specified by "Output Property - Password", e.g. CyberArk/Vault/Password>	Value of the credential
<specified by "Output Property - User Name", e.g. CyberArk/Vault/User>	User Name of the credential
<specified by "Output Property - Address", e.g. CyberArk/Vault/Address>	Address of the credential

4.1.2 Get Password from CCP (Web Service)

This function allows credentials to be retrieve from Vault server via Secrets Manager Central Credential Provider. CyberArk/username, CyberArk/address, and CyberArk/password

4.1.2.1 Input Fields

Name	Description	Example
Name	Name of the step	Get Password from CCP (Web Service)
Server URL	The URL of your CyberArk CCP	https://host:port/AIMWebService/api/accounts
Application ID	The unique ID of the application issuing the password request	
Safe	The name of the safe where the password is stored	
Folder	The name of the folder where the password is stored	
Object Name	The name of the password object to retrieve	

Process Property Prefix	The value to be prepended to each process request property that is created by this step. You may address these properties in subsequent steps with the syntax: <code>\${p:/password}</code> for instance	
Keystore File	The path to the agent machine's keystore file. This is required when the CyberArk server authenticates applications using client certificates	
Keystore Password	The password of the agent machine's keystore	
Keystore Type	The type of keystore on the agent machine	
Trust Invalid Certificates	Check this box to trust all SSL certificates on the agent machine. This will trust any certificate returned from the CyberArk server during connection	
SSL/TLS Debug Level	Specify a debug level to set the 'javax.net.debug' system property. A level of 'all' will log everything. You can specify more specific logging level with values. For instance 'ssl:handshake' will only log information regarding handshakes between the client and server.	

4.2 Conjur

4.2.1 Authenticate Conjur

This step gets a [short-lived access token](#), which can be used to authenticate requests to (most of) the rest of the Conjur API. A client can obtain an access token by presenting a valid login name and API key.

4.2.1.1 Input Fields

Name	Description	Example
Name	Name of the step	Authenticate Conjur
Account	Organization account name	
Login	Host name for authenticating Conjur	cf-spring-app-01

API Key	API Key for authenticating Conjur	
Conjur URL	URL of Conjur cluster	https://eval.conjur.org
Proxy	Proxy address for calling Conjur REST API. Leave it blank if direct connection is allowed	ipv4.124.244.113.228.hybrid-web.global.blackspider.com:80
Output Property - Access Token	Property for storing the return access token	CyberArk/Conjur/AccessToken

4.2.1.2 Output Fields

Name	Description
<specified by "Output Property - Access Token", e.g. CyberArk/Conjur/AccessToken>	Short-lived access token

4.2.2 Get Variable from Conjur

4.2.2.1 Input Fields

Name	Description	Example
Name	Name of the step	Get Variable from Conjur
Account	Organization account name	
Access Token	Short-lived access token	
Variable ID	ID of the variable	db/prod/pws/db01/serviceA
Conjur URL	URL of Conjur cluster	https://eval.conjur.org
Proxy	Proxy address for calling Conjur REST API. Leave it blank if direct connection is allowed	ipv4.124.244.113.228.hybrid-web.global.blackspider.com:80
Output Property - Variable	Property for storing the value of the secret	CyberArk/Conjur/Variable

4.2.2.2 Output Fields

Name	Description
<specified by "Output Property - Variable", e.g. CyberArk/Conjur/Variable>	Value of the secret