**DAP Integration with Jenkins**

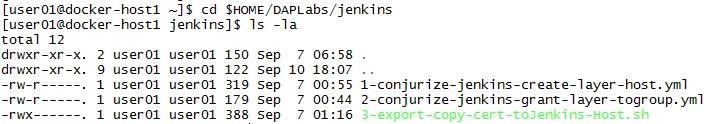
In this example, we will be integrating DAP with Jenkins. This will allow Jenkins to be able to retrieve secrets securely from DAP instead of having the secrets hard-coded in the pipeline and freestyle scripts.

**Conjurize Jenkins**

1. If not already, log into the Conjur CLI
2. First, we need to create a Host id and API key for Jenkins host which allows DAP/Conjur to uniquely identify and authenticate the Jenkins host. Navigate to the working folder and view the directory contents:

**cd $HOME/DAPLabs/jenkins**

**ls -la 2**

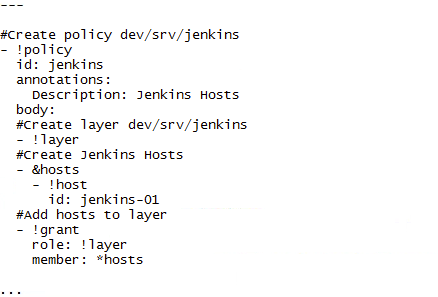


1. Next, create the Jenkins host using YAML file **1-conjurize-jenkins-create-layer- host.yml**:

* Create a sub policy **dev/srv/jenkins**
* Create a **layer** that uses the policy **dev/srv/jenkins** as its name
* Create a host **jenkins-01**
* Add host **jenkins-01** to the **layer**

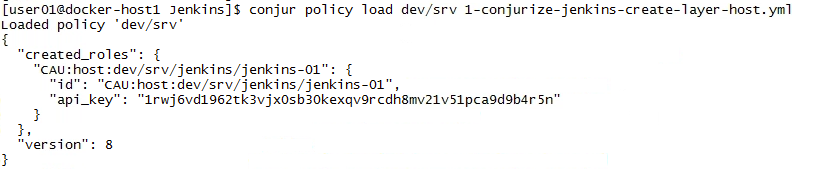
Preview file contents:

**cat 1-conjurize-jenkins-create-layer-host.yml**

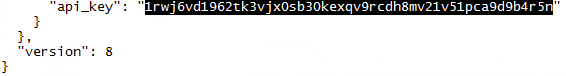


Remember to login to DAP CLI using credentials **admin** / **Cyberark1**. Load the policy into the **dev/srv** policy branch by executing the following commands:

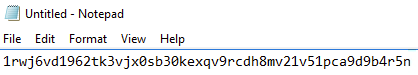
**conjur authn login -u admin  
Cyberark1  
conjur policy load dev/srv 1-conjurize-jenkins-create-layer-host.yml**



**NOTE:** Notice a unique API key is created for host **jenkins-01**. Copy this key by **selecting** it and simply **right-click** to copy it to the clipboard.



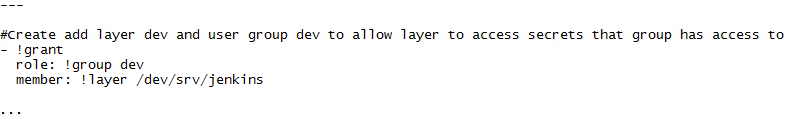
Launch Notepad++ or notepad, right-click and paste. We need to use this key later for the Jenkins setup. Note: Remove the double quotes **“**.



1. Next, we will execute the following action using YAML file **2-conjurize-jenkins- grant-layer-togroup.yml**:

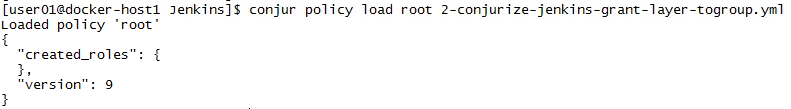
Add layer **dev/srv/jenkins** to group **dev** as a member  
This will allow the host **jenkins-01** to inherit the same privileges to the secrets which group **dev** has.

Preview the file contents:

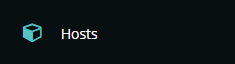
**cat 2-conjurize-jenkins-grant-layer-togroup.yml** 

Load the policy into the **root** policy branch by executing the following commands:

**conjur policy load root 2-conjurize-jenkins-grant-layer-togroup.yml**



1. Login to the Conjur UI using **admin** / **Cyberark1.** Click on **Hosts** and verify that **dev/srv/jenkins/jenkins-01** host is created.

page152image17256720

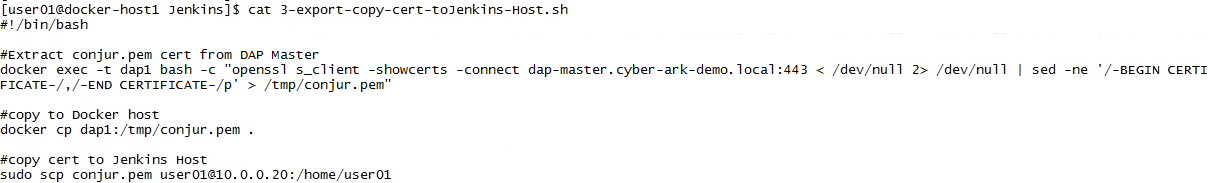
Click on the new host **dev/srv/jenkins/jenkins-01**, scroll down to the **Privileges Held** section and verify that the host could access the secrets created earlier:



1. Next, we need to extract the DAP Master cert and import it into Jenkins host keystore so Jenkins will be able communicate with DAP Master via RESTAPI. We will be using a prebuilt script **3-export-copy-cert-toJenkins-Host.sh** to extract the cert and copy it over to the Jenkins host.

Preview the file contents:

**cat 3-export-copy-cert-toJenkins-Host.sh**



Execute the following commands:

**sudo chmod 700 3-export-copy-cert-toJenkins-Host.sh ./3-export-copy-cert-toJenkins-Host.sh**Are you sure you want to continue...? **yes**

User01@10.0.0.20’s password: **Cyberark1**

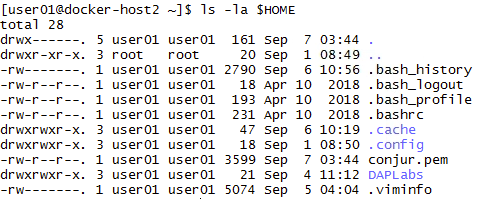


**Setup Jenkins**

1. Next, log into the **Docker-Host2 (10.0.0.20)** virtual machine using SSH and authenticate with the **user01 / Cyberark1** credentials

2. Once connected, verify the **conjur.pem** file is copied over.

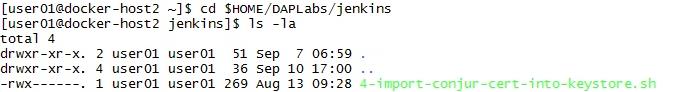
**ls -la $HOME**



3. Navigate to the **jenkins** working folder and list the contents:

**cd $HOME/DAPLabs/jenkins**

**ls -la**



4. We will be using the prebuilt script **4-import-conjur-cert-into-keystore.sh** to import the **conjur.pem** cert into the Java keystore used by Jenkins.

Preview the file contents:

**cat 4-import-conjur-cert-into-keystore.sh**

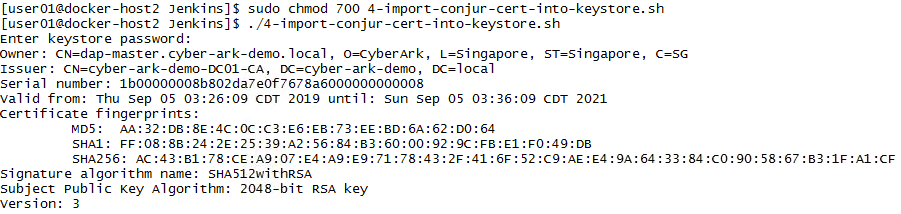


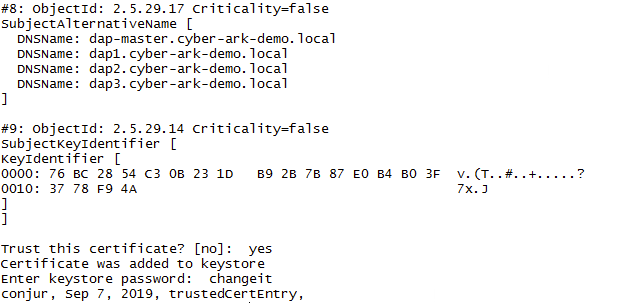
Execute the script via following commands and inputs:

**sudo chmod 700 4-import-conjur-cert-into-keystore.sh ./4-import-conjur-cert-into-keystore.sh**

Enter keystore password: **changeit**Trust this certificate? [no]: **yes**

Enter keystore password: **changeit**





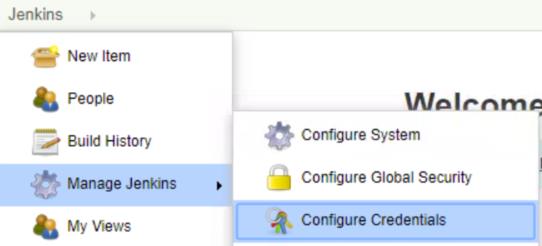
The boxed area proves that **conjur** cert is installed into the Java keystore.

5. We will now shift over to Jenkins and create a credential made up of the Jenkins host id **jenkins-01** and the API key. This credential will be used by Jenkins to authenticate and communicate with DAP Master. Secondly, we will be creating 2 DAP/Conjur secrets in Jenkins which will be used in pipeline and freestyle scripts use cases.

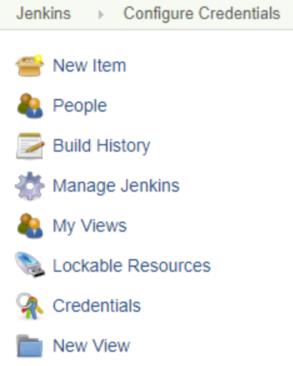
6. Launch Google Chrome browser, then open up a new tab and enter and visit the url: **jenkins.cyber-ark-demo.local:8080**. Login in using the **jenkinsadmin01** / **Cyberark1** credentials and click **Sign In**.



7. Click on the **Jenkins** dropdown list→mouse-over **Manage Jenkins**→click **Configure Credentials**.



8. Click **Credentials** on the left pane.



9. Under the **Credentials** pane, click **Jenkins**.



10.Under the **System** pane, click **Global Credentials (unrestricted)**.



11.Click **Add Credentials**.



12.Enter and select according to the configuration listed:

**Kind** (as default) **Username with password**

**Scope** (as default) **Global (Jenkins, nodes, items, all child items, etc)**

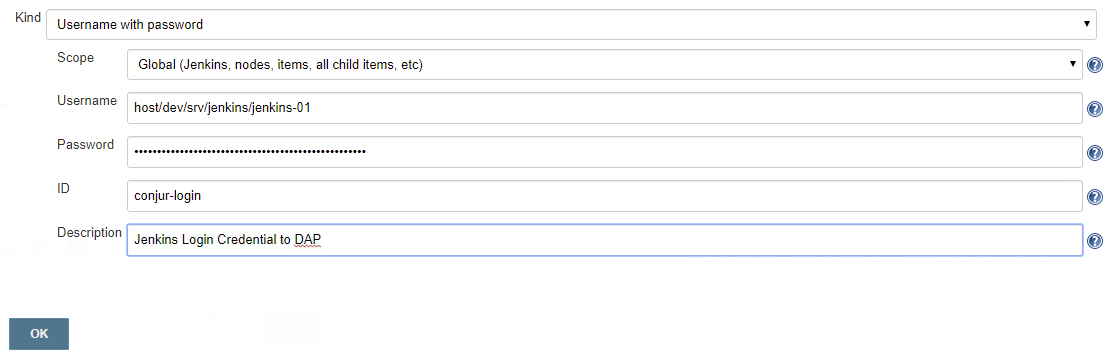
**Username: host/dev/srv/jenkins/jenkins-01**

**Password:** *Enter the Host API key that was copied to Notepad*

**ID: conjur-login**

**Description: Jenkins Login Credential to DAP**

Click **OK** to add the credential.

page158image32116096

**NOTE:**

**If the jenkins-01 host API key is successfully copied simply SKIP to step 13.**

Otherwise, if you did not copy the **jenkins-01** host API key from previous steps and paste it into Notepad++ and/or notepad, then switch back to SSH session to **Docker-Host1**, login as **admin** / **Cyberark1** on the DAP/Conjur cli. Execute the following command to generate a new API key:

**echo $(conjur host rotate\_api\_key -h dev/srv/jenkins/jenkins-01)**

page158image17068848

13.Now configure the secrets to use in the Jenkins scripts. Click **Add Credentials**.

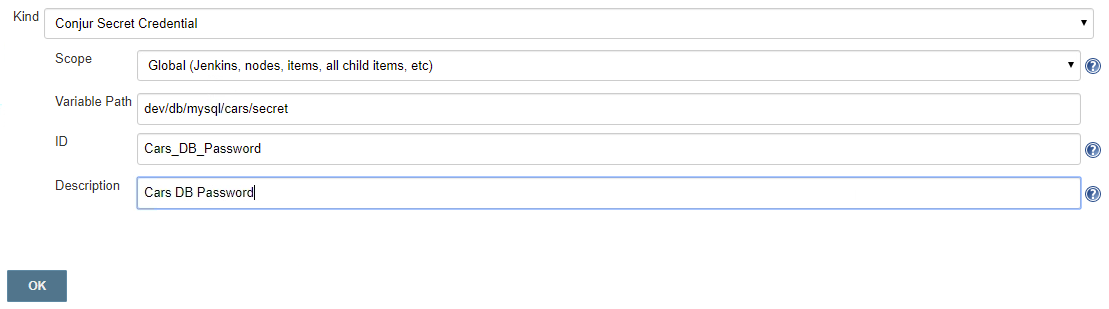


14.Enter and select according to configuration listed:

**Kind** Switch to **Conjur Secret Credential  
Scope** (as default) **Global (Jenkins, nodes, items, all child items, etc)**

**Variable Path: dev/db/mysql/cars/secret  
ID: Cars\_DB\_Password  
Description: Cars DB Password**

Click **OK** to have the secret created.

page159image31781440

15.Click **Add Credentials**.

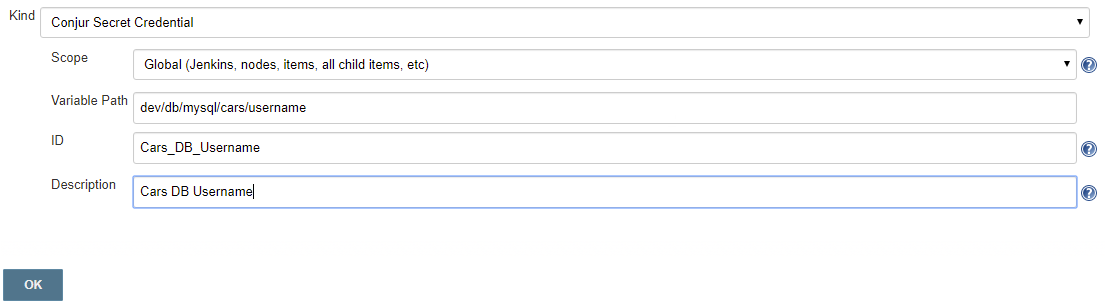


16.Enter and select according to configuration listed:

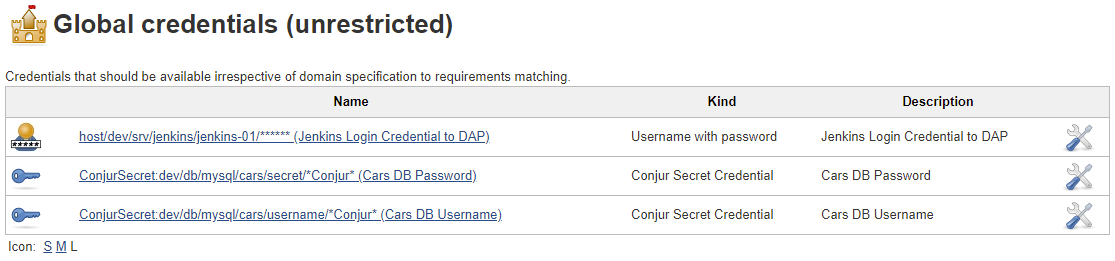
**Kind** Switch to **Conjur Secret Credential  
Scope** (as default) **Global (Jenkins, nodes, items, all child items, etc)**

**Variable Path: dev/db/mysql/cars/username  
ID: Cars\_DB\_Username  
Description: Cars DB Username**

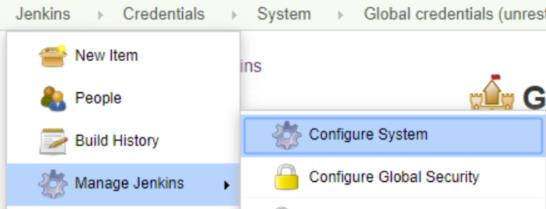
Click **OK** to have the secret created.



17.Verify that 1 credential and 2 secrets are created:



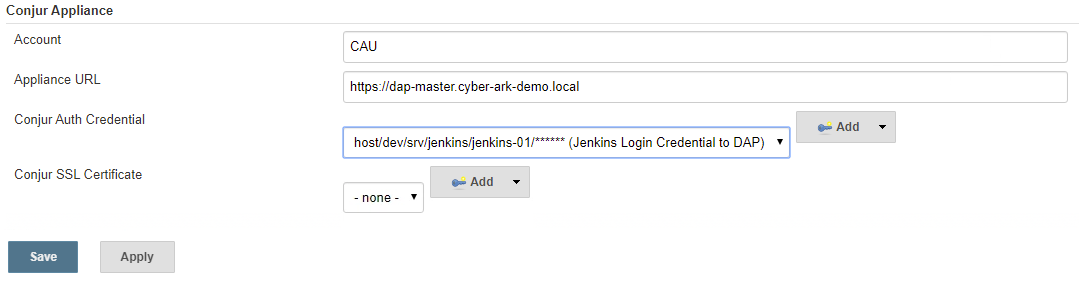
18.Next, we will configure Jenkins to connect to the DAP Master. Click on **Jenkins** dropdown list→mouse-over **Manage Jenkins**→click on **Configure System**.



Scroll down to the section **Conjur Appliance** and enter the following:

**Account CAU  
Appliance URL https://dap-master.cyber-ark-demo.local  
Conjur Auth Credential** select credential **host/dev/srv/jenkins/jenkins-01**

Click **Save** to save the configuration.

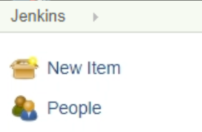


**Jenkins Configuration Complete!**

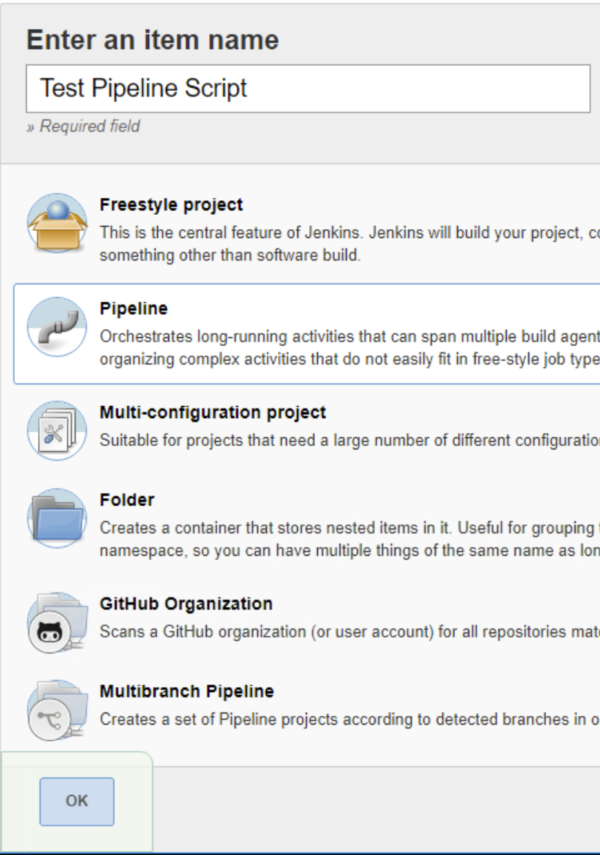
**Use Case – Conjur Secrets with Jenkins Pipeline Script**

1. If not already logged into Jenkins, then launch Chrome → go to URL: **jenkins.cyber-ark-demo.local** → login using the **jenkinsadmin01** / **Cyberark1** credentials.

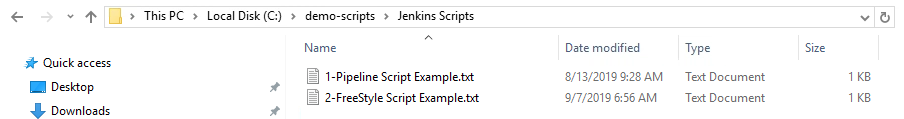
2. Click on **New Item** on the left pane.



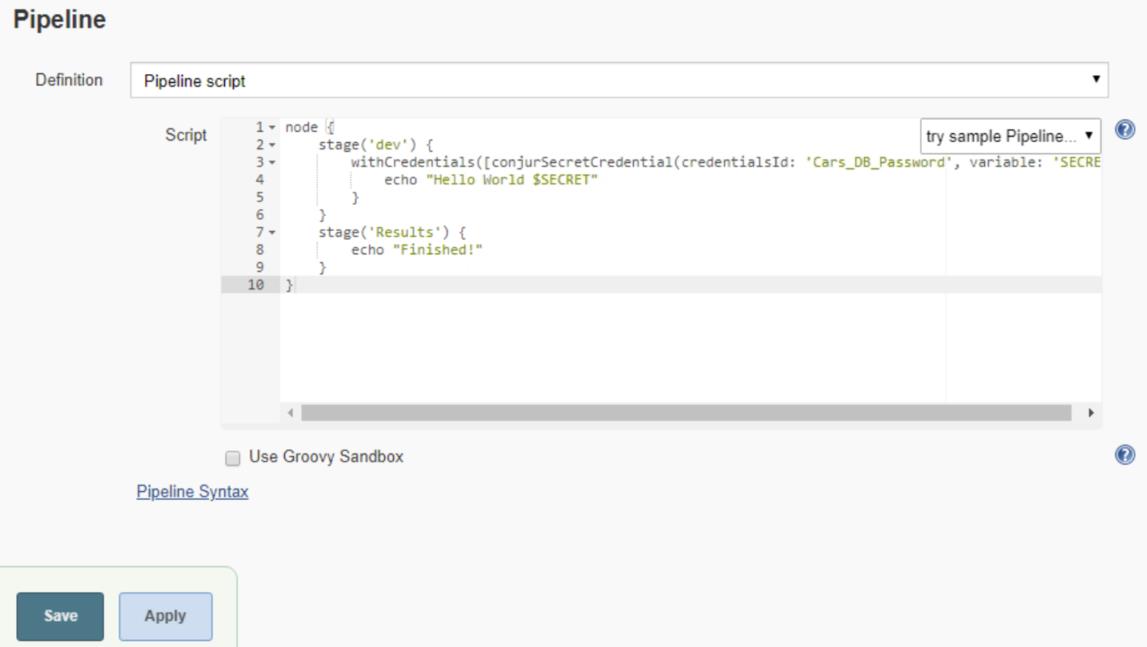
3. Create a **Pipeline** script. Enter **Test Pipeline Script** as item name. Select **Pipeline** and click **OK**.



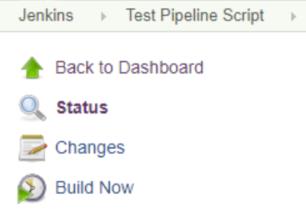
4. Scroll down to section **Pipeline**. On the component servers, use windows explorer to browse to and edit file **C:\demo-scripts\Jenkins Scripts\1-Pipeline Script Example.txt** using Notepad++.



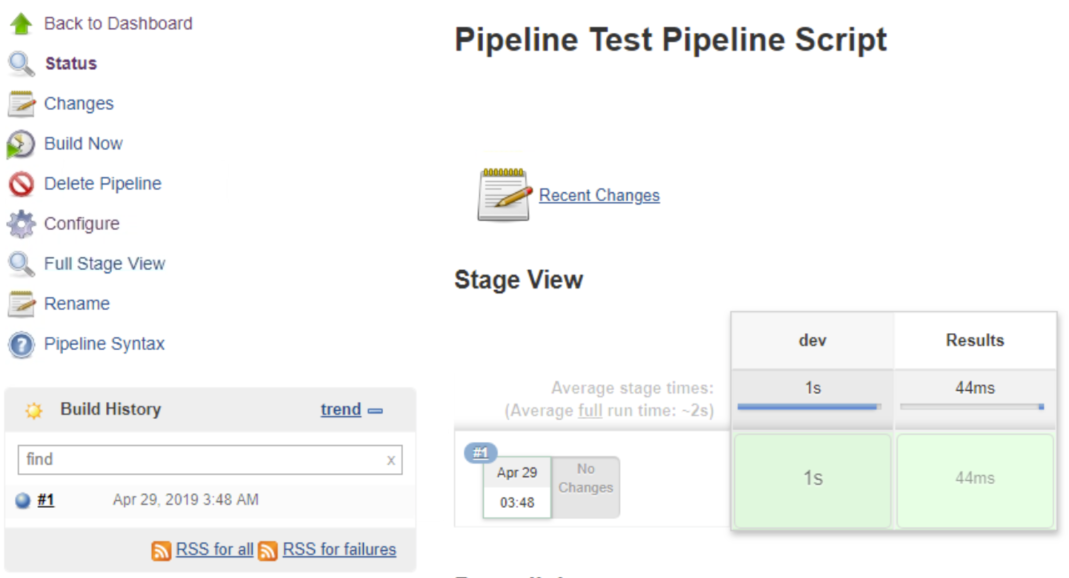
5. Copy all contents of the textfile and paste it into the **Script** window,uncheck **Use Groovy Sandbox** and click **Save**.



6. Click **Build Now** to test the script.



7. Verify build execution is successful as shown in screenshot.



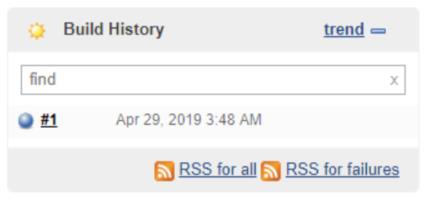
a. Click on the green area below **dev**→click **Logs**



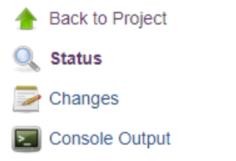
b. At the top, this message should appear. Close the message using **x**.

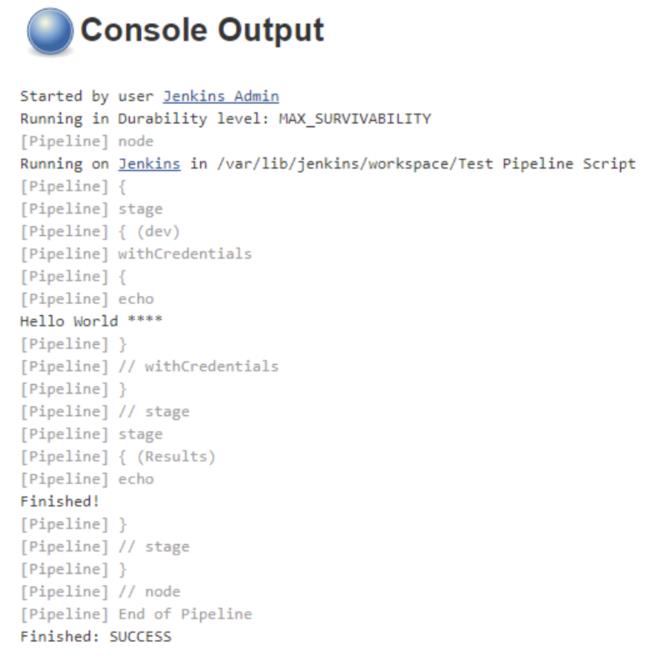


c. Click on the build number.



d. Click **Console Output** and verify if console output is similar to the screenshot below. Congratulations! Jenkins is now able to securely fetch secrets from Conjur.





1. Login to Conjur UI via https://dap-master.cyber-ark-demo.local as **admin** / **Cyberark1**.
2. Click **Secrets** > click **dev/db/mysql/cars/secret**. Scroll down to **Audit Events** and validate that host **dev/srv/jenkins/jenkins-01** fetched the secret successfully.