Install Plugin

This is the install and configure doc for the Livefire created vRealize Orchestrator Plugin for AWS.

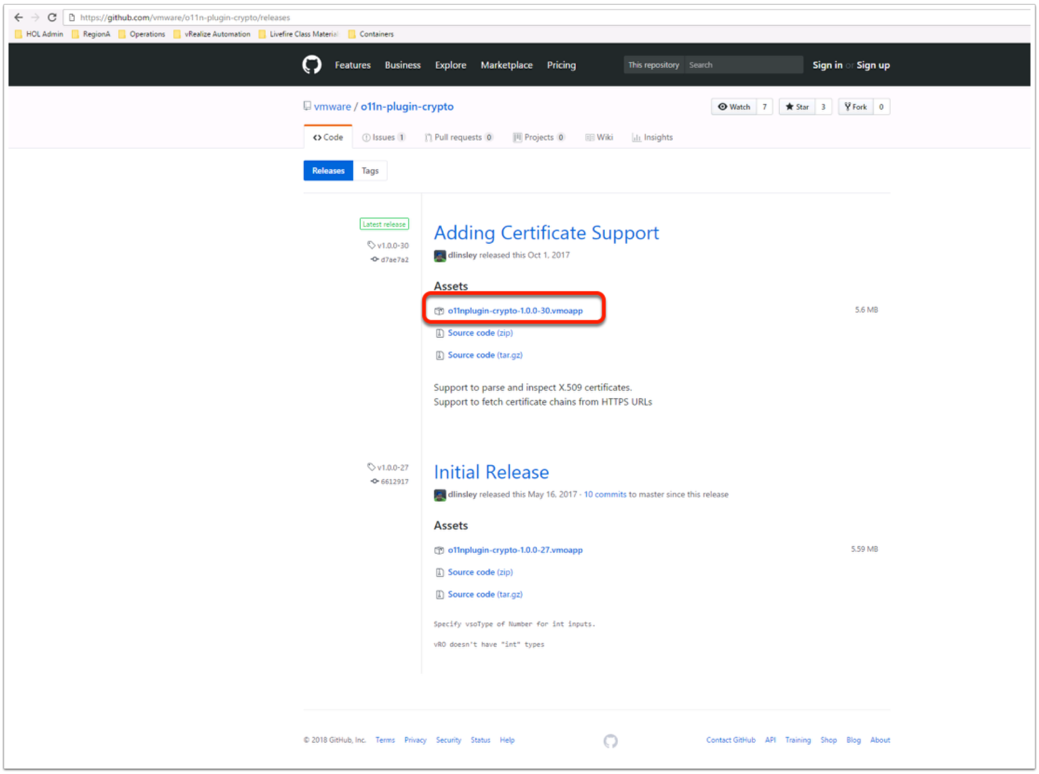
This plugin uses the AWS api using via REST. When you send HTTP requests to AWS, you sign the requests so that AWS can identify who sent them. You sign requests with your AWS access key, which consists of an access key ID and secret access key. This plugin takes care of the signing for you, and works for all regions tested.

Please send all feedback to toddb@vmware.com and cdecanini@vmware.com

Thanks and enjoy this free to use, Livefire makes no warranty that

- the software will meet your requirements
- the software will be uninterrupted, timely, secure or error-free
- the results that may be obtained from the use of the software will be effective, accurate or reliable
- the quality of the software will meet your expectations
- any errors in the software obtained from us will be corrected.

Download Crypto Plugin for vRO



Adding Certificate Support

dlinsley released this Oct 1, 2017

Assets

o11nplugin-crypto-1.0.0-30.vmoapp	5.6 MB
Source code (zip)	
Source code (tar.gz)	

Support to parse and inspect X.509 certificates.
Support to fetch certificate chains from HTTPS URLs

Initial Release

dlinsley released this May 16, 2017 · 10 commits to master since this release

Assets

o11nplugin-crypto-1.0.0-27.vmoapp	5.59 MB
Source code (zip)	
Source code (tar.gz)	

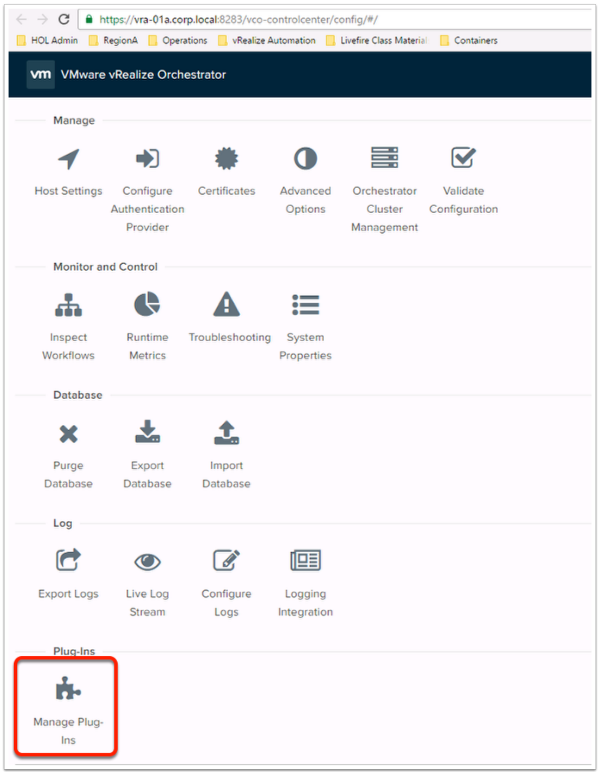
```
Specify vsoType of Number for int inputs.

vRO doesn't have "int" types
```

To do the crypto work to sign requests this plugin uses a crypto plugin for vRO available here - https://github.com/vmware/o11n-plugin-crypto/releases.

Download the .vmoapp file

Manage Plugins



In your browser go to https://<vro FQDN or IP Address>:8283/vco-controlcenter/

Log In

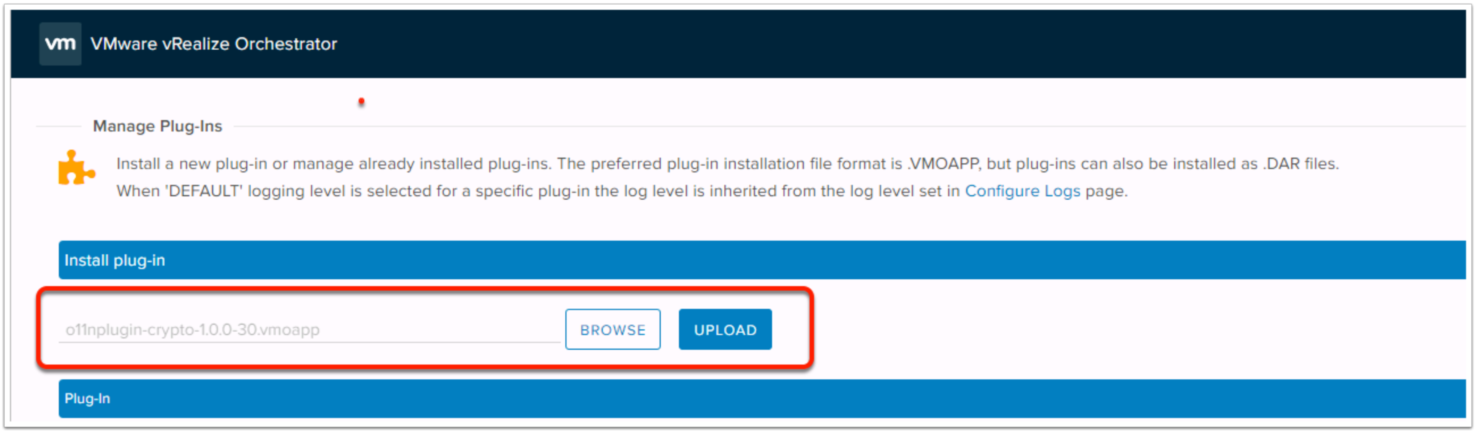
```
For Livefire PODs use:
UserID: root
Password: VMware1!
```

Click on Manage Plug-Ins

If you get a 503 Service Unavailable do some google searching on how to get the vRO Control center running.

HINT: putty (ssh) into the vRO Appliance.

Install Crypto Plugin

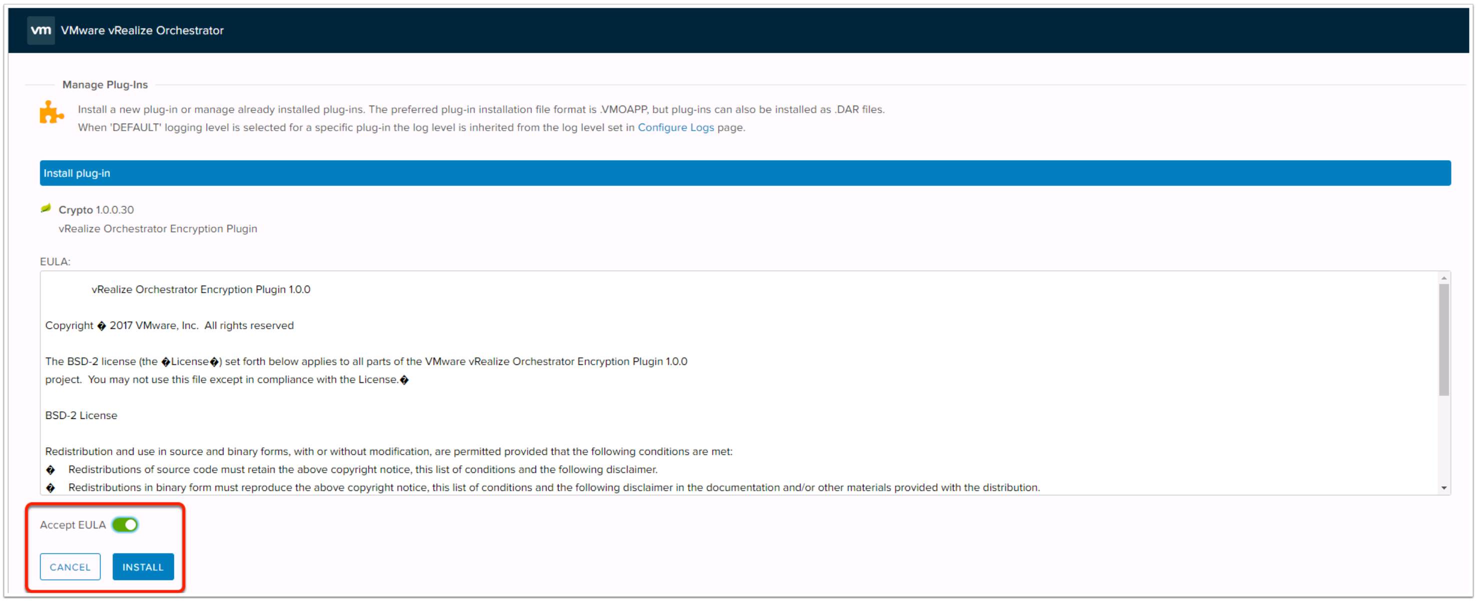


Install a new plug-in or manage already installed plug-ins. The preferred plug-in installation file format is .VMOAPP, but plug-ins can also be installed as .DAR files.
When 'DEFAULT' logging level is selected for a specific plug-in the log level is inherited from the log level set in Configure Logs page.

o11nplugin-crypto-1.0.0-30.vmoapp

Browse to the file you download in the first step o11nplugin-crypto-1.0.0-30.vmoapp click upload

Install Crypto Plugin



Install a new plug-in or manage already installed plug-ins. The preferred plug-in installation file format is .VMOAPP, but plug-ins can also be installed as .DAR files.
When 'DEFAULT' logging level is selected for a specific plug-in the log level is inherited from the log level set in Configure Logs page.

Crypto 1.0.0.30
vRealize Orchestrator Encryption Plugin

EULA:

vRealize Orchestrator Encryption Plugin 1.0.0

Copyright � 2017 VMware, Inc. All rights reserved

The BSD-2 license (the �License�) set forth below applies to all parts of the VMware vRealize Orchestrator Encryption Plugin 1.0.0 project. You may not use this file except in compliance with the License.�

BSD-2 License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

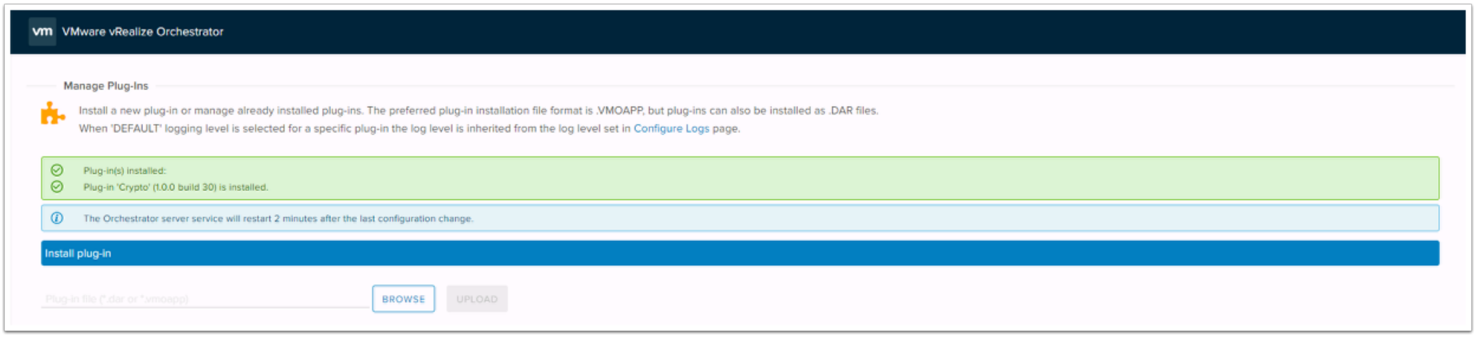
� Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

� Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Accept EULA ●

Accept the EULA

Click Install



VMware vRealize Orchestrator

Manage Plug-Ins

Install a new plug-in or manage already installed plug-ins. The preferred plug-in installation file format is .VMOAPP, but plug-ins can also be installed as .DAR files.
When 'DEFAULT' logging level is selected for a specific plug-in the log level is inherited from the log level set in Configure Logs page.

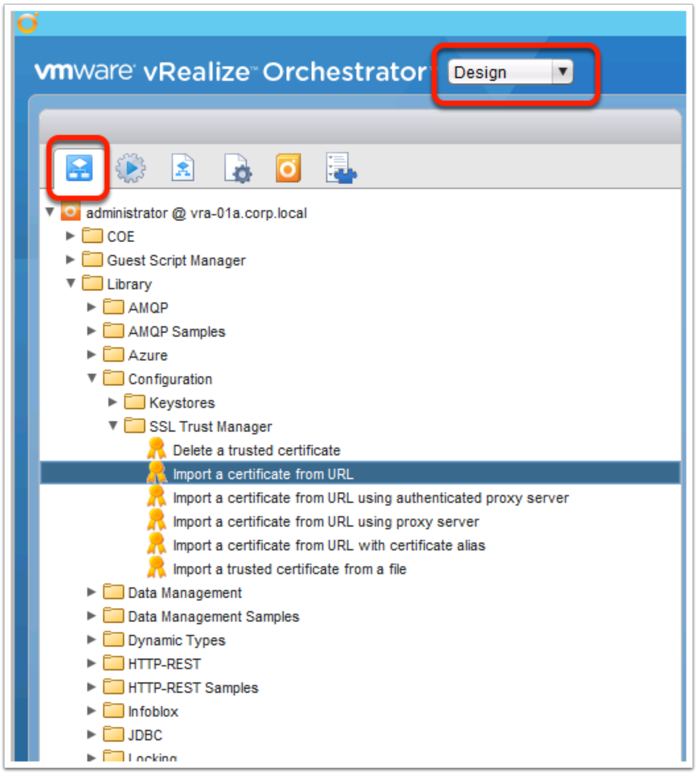
Plug-in(s) installed:
Plug-in 'Crypto' (1.0.0 build 30) is installed.

The Orchestrator server service will restart 2 minutes after the last configuration change.

Install plug-in

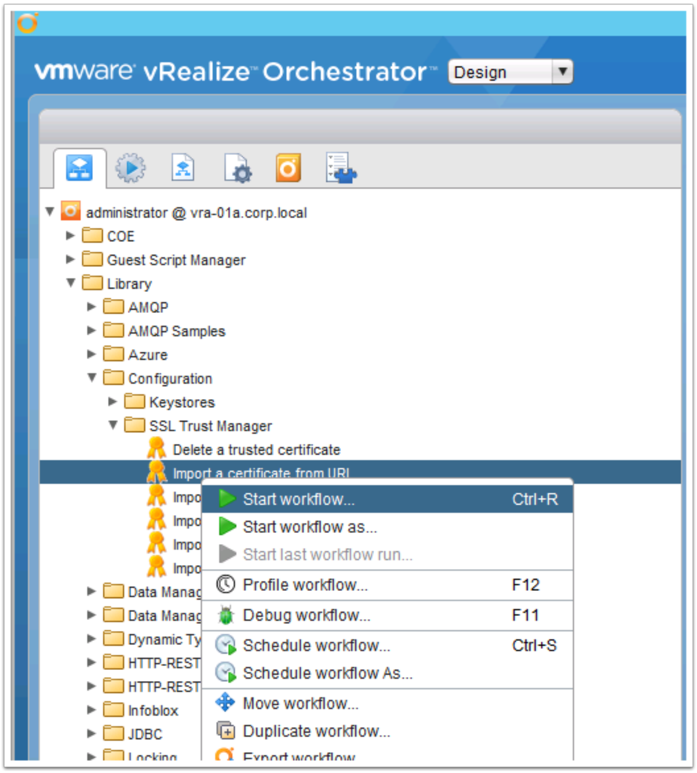
You should see **Plug-in installed**

Import AWS SSL Certificate


vmware vRealize Orchestrator
Design
administrator @ vra-01a.corp.local
COE
Guest Script Manager
Library
AMQP
AMQP Samples
Azure
Configuration
Keystores
SSL Trust Manager
Delete a trusted certificate
Import a certificate from URL
Import a certificate from URL using authenticated proxy server
Import a certificate from URL using proxy server
Import a certificate from URL with certificate alias
Import a trusted certificate from a file
Data Management
Data Management Samples
Dynamic Types
HTTP-REST
HTTP-REST Samples
Infoblox
JDBC
Locking

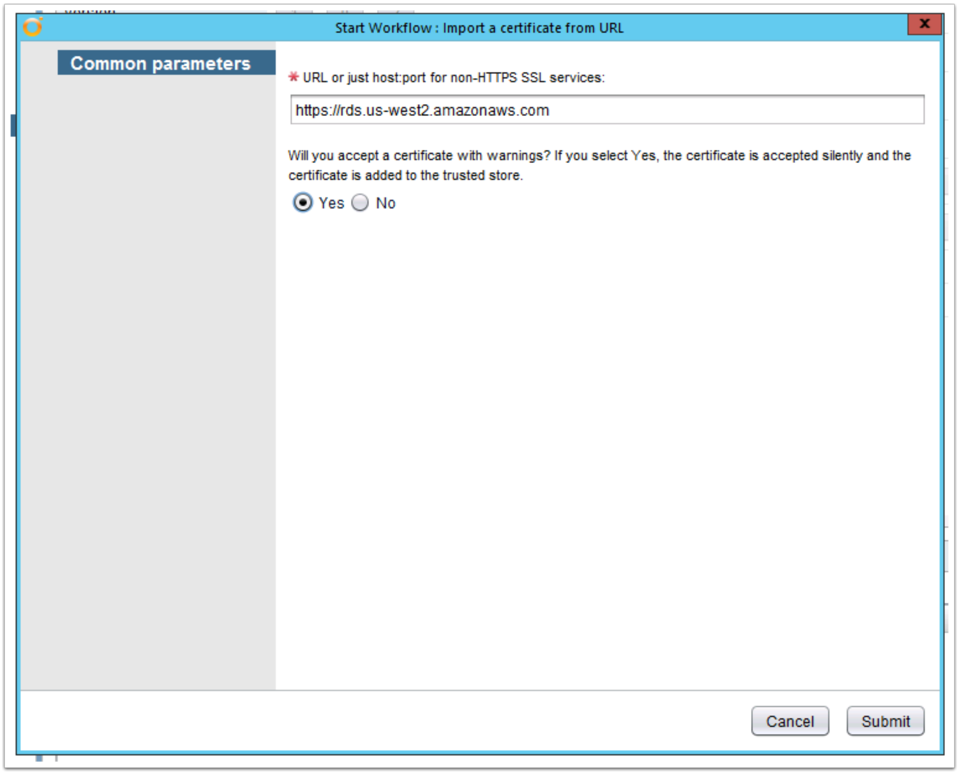
- Open the Orchestrator Client, and connect to vRO as administrator
- Change the Dropdown to **Design**
- Click **Workflows**
- Expand **Library** --> **Configuration** --> **SSL Trust Manager**

Import a certificate from URL



Right Click **Import certificate from URL**

Select **Start workflow**



Start Workflow : Import a certificate from URL

* URL or just host:port for non-HTTPS SSL services:

https://rds.us-west2.amazonaws.com

Will you accept a certificate with warnings? If you select Yes, the certificate is accepted silently and the certificate is added to the trusted store.

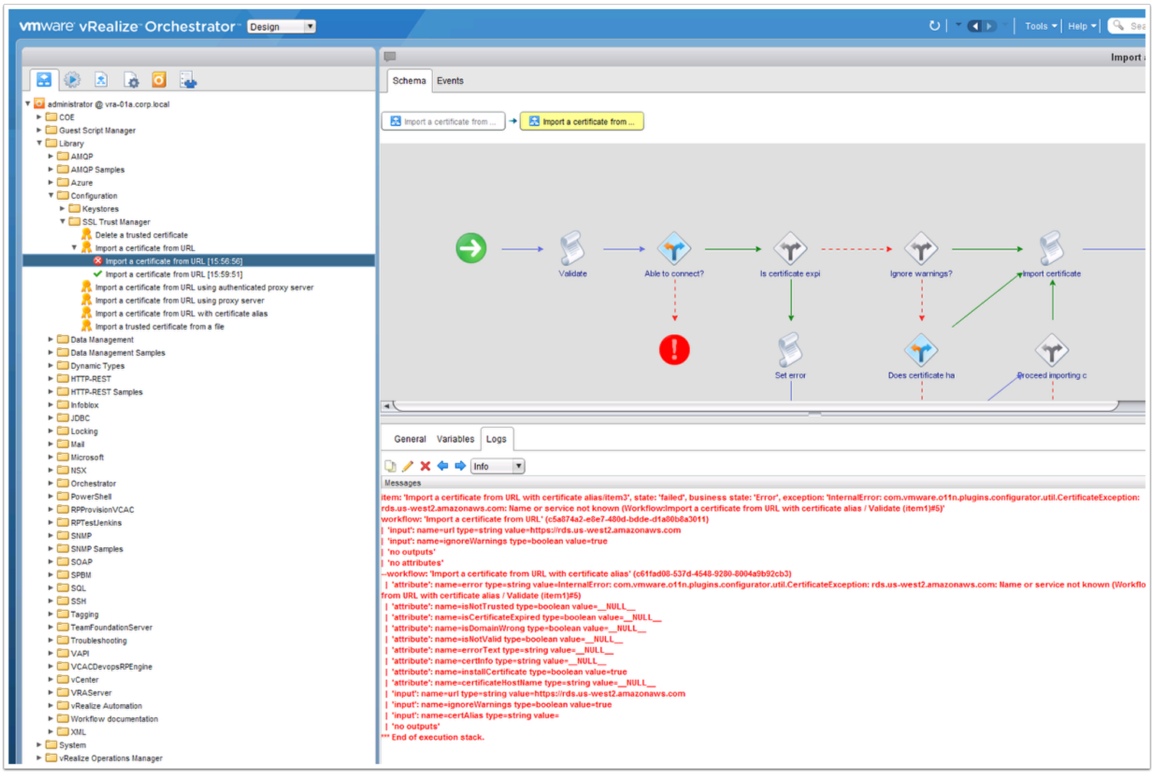
● Yes ○ No

For the URL Enter: https://rds.us-west-2.amazonaws.com

Check the **Yes** radio check button

Click **Submit**

Don't Fat Finger

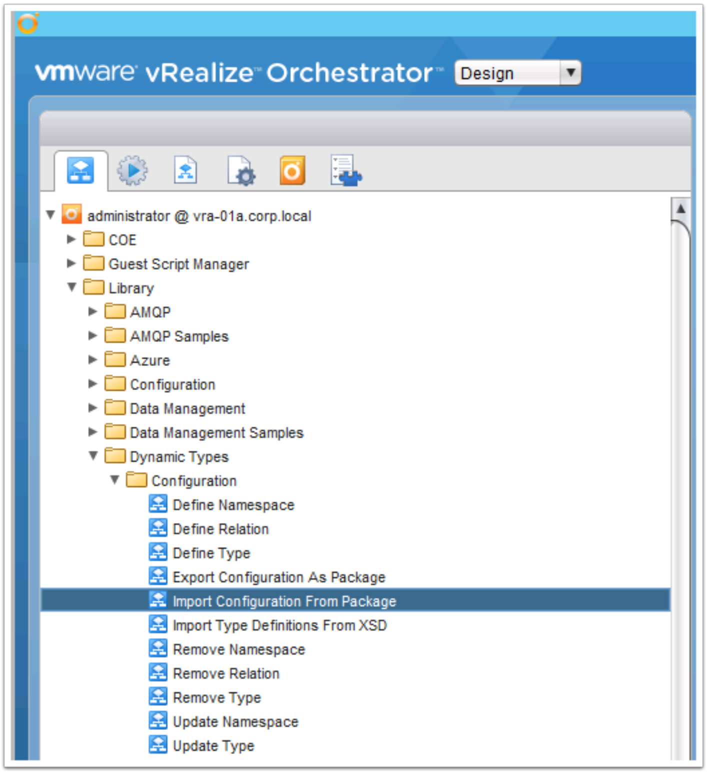


```
item: 'Import a certificate from URL with certificate alias/item3', state: 'failed', business state: 'Error', exception: 'InternalError: com.vmware.o11n.plugins.configurator.util.CertificateException:
rds.us-west2.amazonaws.com: Name or service not known (Workflow:Import a certificate from URL with certificate alias / Validate (item1)#5)'
workflow: 'Import a certificate from URL' (c5a874a2-e8e7-480d-bdde-d1a80b8a3011)
|  'input': name=url type=string value=https://rds.us-west2.amazonaws.com
|  'input': name=ignoreWarnings type=boolean value=true
|  'no outputs'
|  'no attributes'
--workflow: 'Import a certificate from URL with certificate alias' (c61fad08-537d-4548-9280-8004a9b92cb3)
  |  'attribute': name=error type=string value=InternalError: com.vmware.o11n.plugins.configurator.util.CertificateException: rds.us-west2.amazonaws.com: Name or service not known (Workflo
from URL with certificate alias / Validate (item1)#5)
  |  'attribute': name=isNotTrusted type=boolean value=__NULL__
  |  'attribute': name=isCertificateExpired type=boolean value=__NULL__
  |  'attribute': name=isDomainWrong type=boolean value=__NULL__
  |  'attribute': name=isNotValid type=boolean value=__NULL__
  |  'attribute': name=errorText type=string value=__NULL__
  |  'attribute': name=certInfo type=string value=__NULL__
  |  'attribute': name=installCertificate type=boolean value=true
  |  'attribute': name=certificateHostName type=string value=__NULL__
  |  'input': name=url type=string value=https://rds.us-west2.amazonaws.com
  |  'input': name=ignoreWarnings type=boolean value=true
  |  'input': name=certAlias type=string value=
  |  'no outputs'
*** End of execution stack.
```

Debug if needed.

In this case I typed https://rds.us-west2.amazonaws.com not https://rds.us-west-2.amazonaws.com

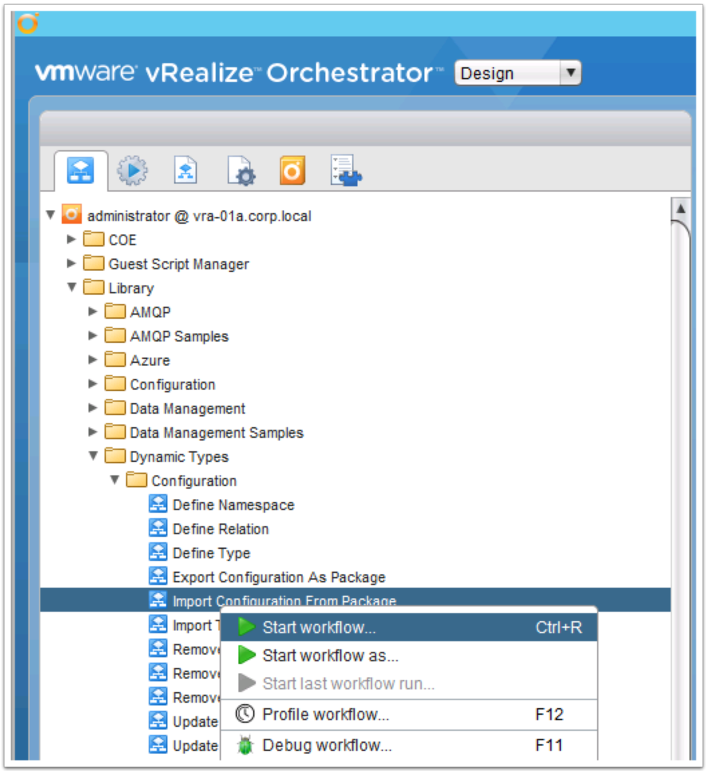
Run the Import Workflow



administrator @ vra-01a.corp.local
- COE
- Guest Script Manager
- Library
  - AMQP
  - AMQP Samples
  - Azure
  - Configuration
  - Data Management
  - Data Management Samples
  - Dynamic Types
    - Configuration
      - Define Namespace
      - Define Relation
      - Define Type
      - Export Configuration As Package
      - Import Configuration From Package
      - Import Type Definitions From XSD
      - Remove Namespace
      - Remove Relation
      - Remove Type
      - Update Namespace
      - Update Type

Expand Library --> Dynamic Types --> Configuration

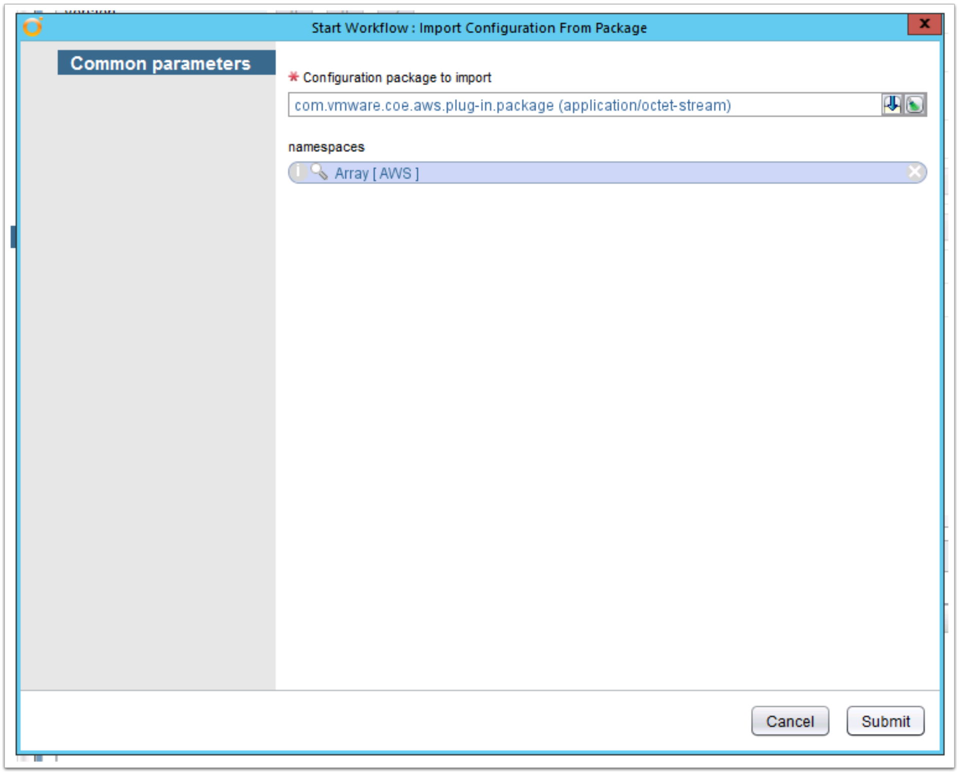
Run the Import Workflow



Right Click on **Import Configuration from Package**

Click **Start workflow**

Import the Plugin



Start Workflow : Import Configuration From Package

Common parameters

* Configuration package to import

com.vmware.coe.aws.plug-in.package (application/octet-stream)

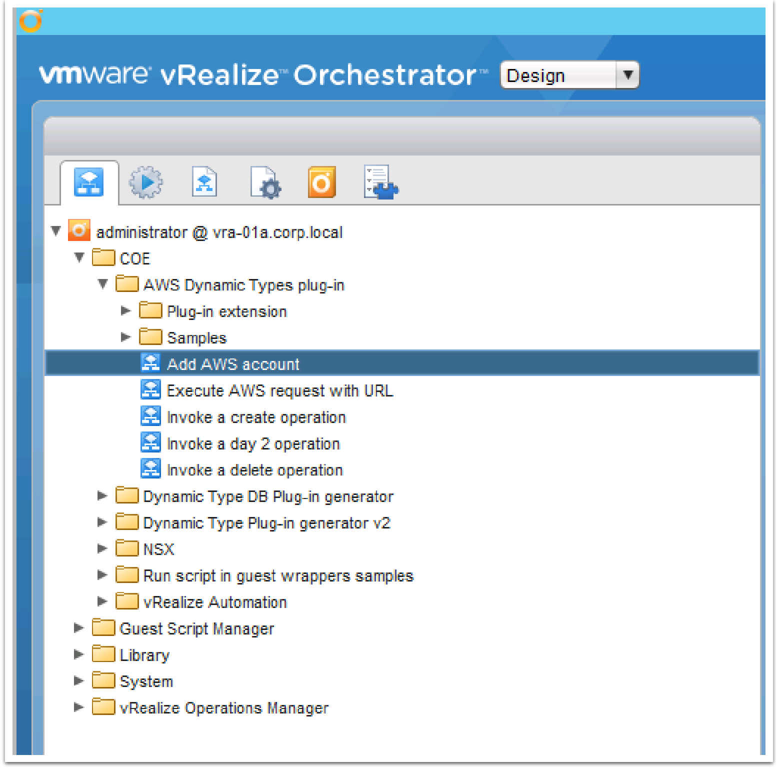
namespaces

Array [AWS]

Browse to the AWS Plug-In package file

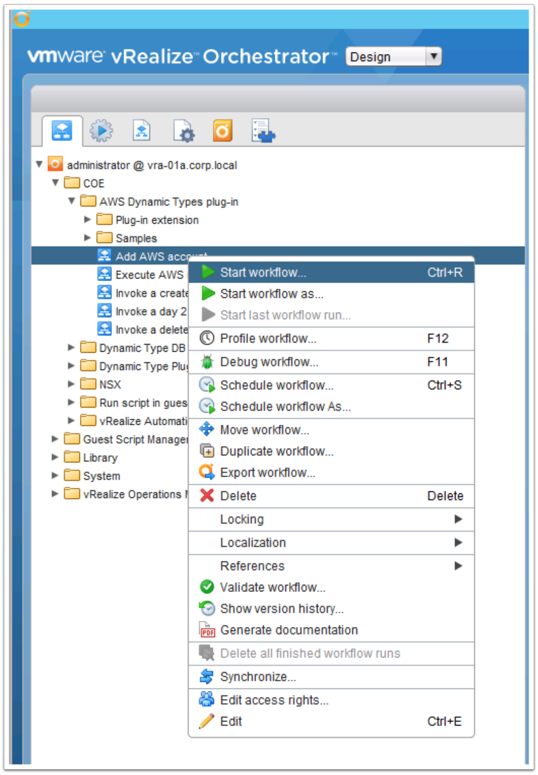
Click **Submit**

Configure the Plugin


vmware vRealize Orchestrator
Design
administrator @ vra-01a.corp.local
COE
AWS Dynamic Types plug-in
Plug-in extension
Samples
Add AWS account
Execute AWS request with URL
Invoke a create operation
Invoke a day 2 operation
Invoke a delete operation
Dynamic Type DB Plug-in generator
Dynamic Type Plug-in generator v2
NSX
Run script in guest wrappers samples
vRealize Automation
Guest Script Manager
Library
System
vRealize Operations Manager

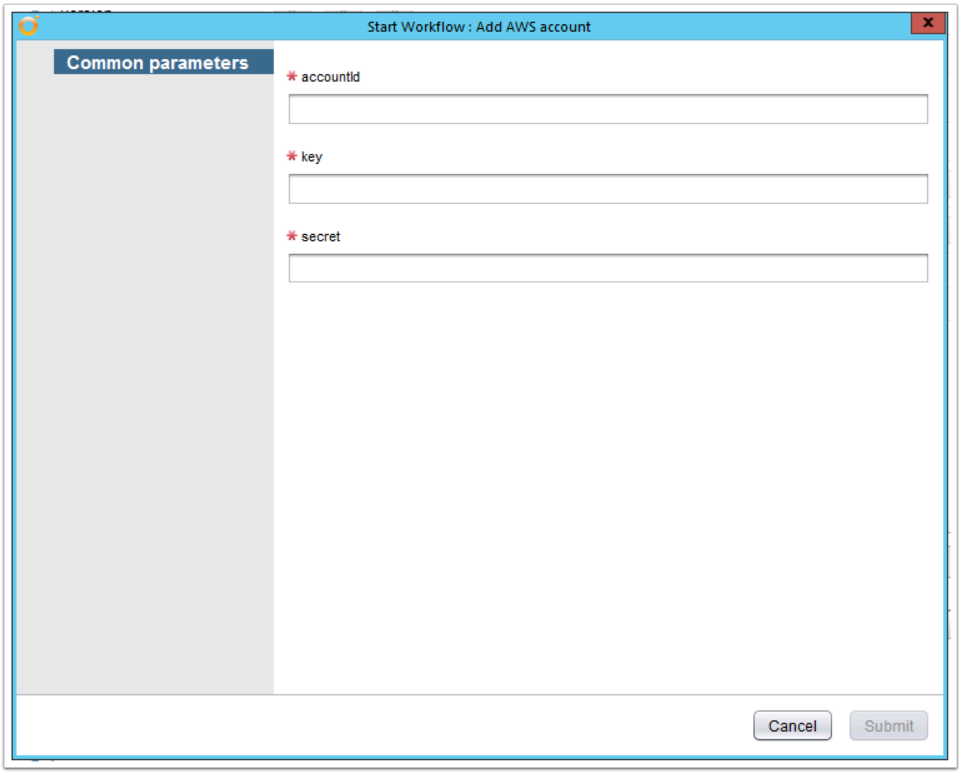
Finally we need to configure the plugin.

Expand COE --> AWS Dynamic Types plug-in



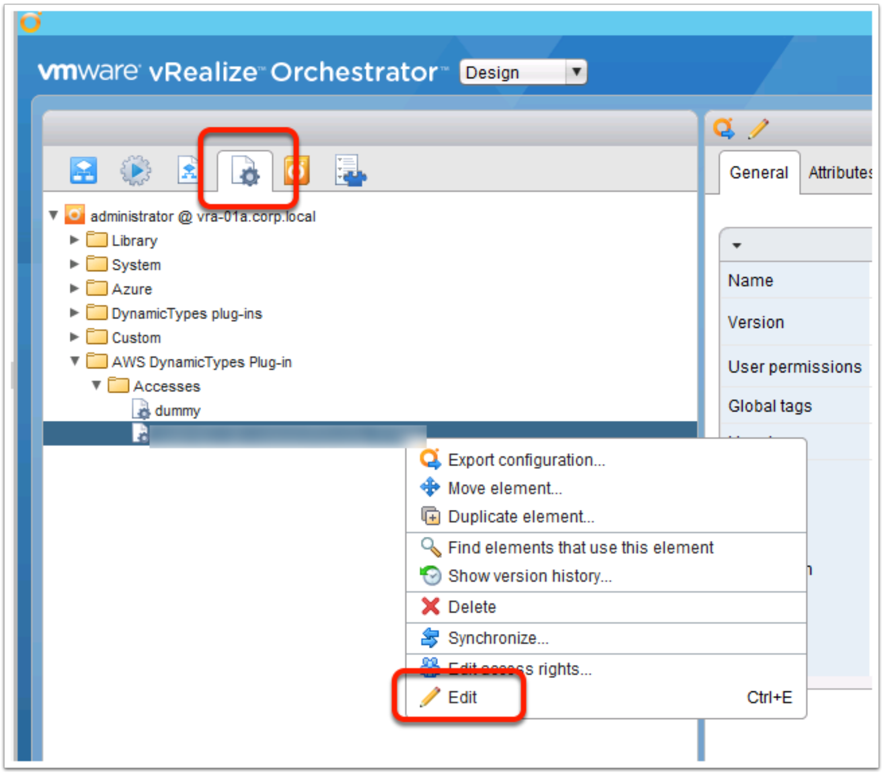
Run workflow Add AWS account.

Configure Plugin with AWS account and service ID info



Start Workflow : Add AWS account

Enter your **AWS account ID**, the **Access key ID** for the service account in your AWS account the plug-in will use. For the **Secret key** enter anything for now, this will be entered next.

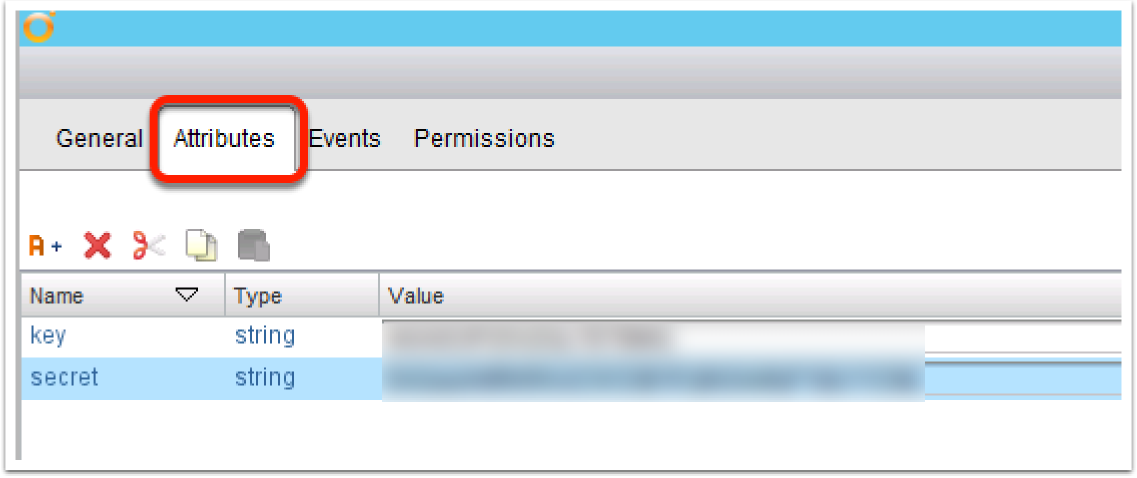


Click on the **Configurations** Tab

Expand **AWS Dynamic Types Plugin-In --> Accesses**

Right click the newly created configuration

Click **Edit**



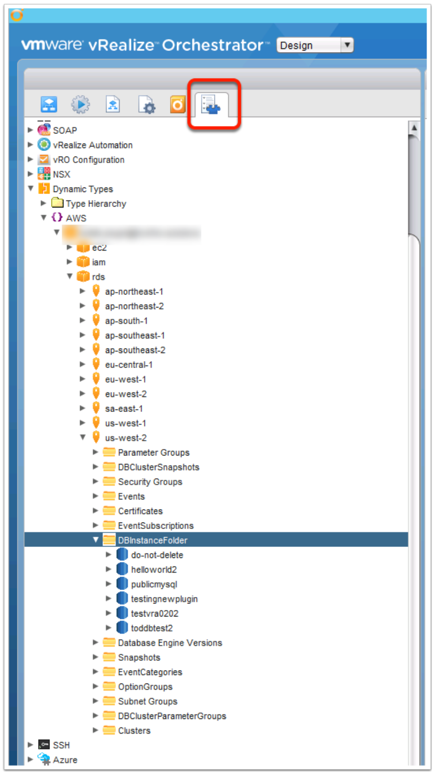
Name	Type	Value
key	string	
secret	string	

Click on the **Attributes** Tab

Enter the **Secret Access Key** that matches the **Access Key ID** being used by the service account in AWS

Click **Save and close**

The Plugin is now configured - Lets test it

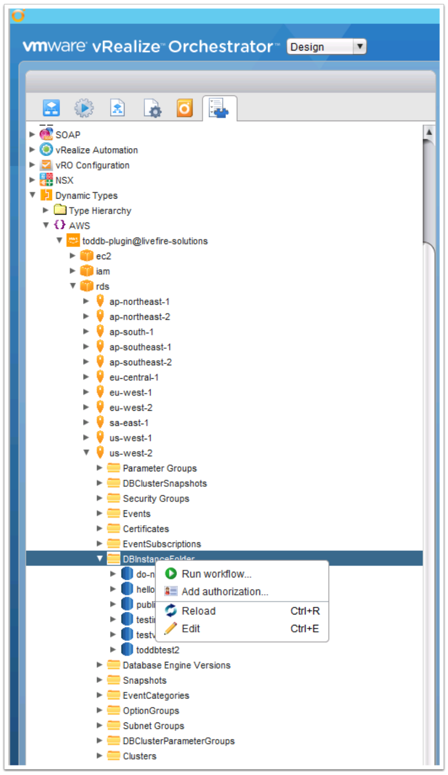


Click the **Inventory** Tab

Expand **Dynamic Types** --> **AWS** --> **RDS** --> **<the region you want to deploy an RDS instance to>** --> **DBINstanceFolder**

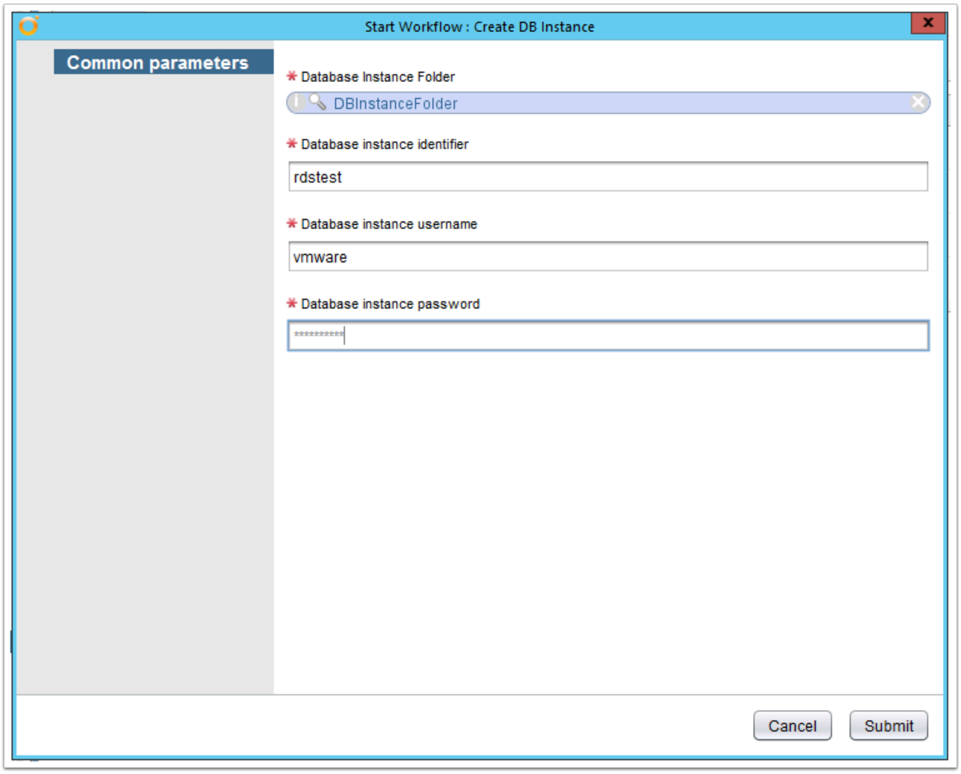
The Plugin will run a query and list all RDS instances you have in your AWS account in that chosen region.

Create a new RDS Instance



Right Click the **DBInstanceFolder** and click **Run workflow**

Create a new RDS Instance



* Database Instance Folder: DBInstanceFolder

* Database instance identifier: rdstest

* Database instance username: vmware

* Database instance password: **********

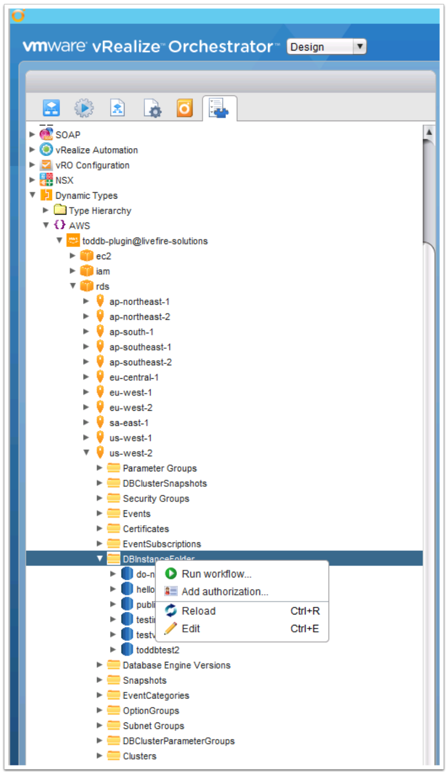
Fill In:

Database instance identifier: *<name of the database instance>*

Database instance username: *<userID to be used for the connecting to the Database Instance>*

Database instance password: *<password to be used for connecting to the Database Instance>*

Create a new RDS Instance



Right Click the **DBInstanceFolder** and click **Reload**

You should see the new DBInstance show up now