A vRO Plugin for Azure

Stephen Mak
Product Value Engineering, Broadcom
Version 0.24, March 6, 2024

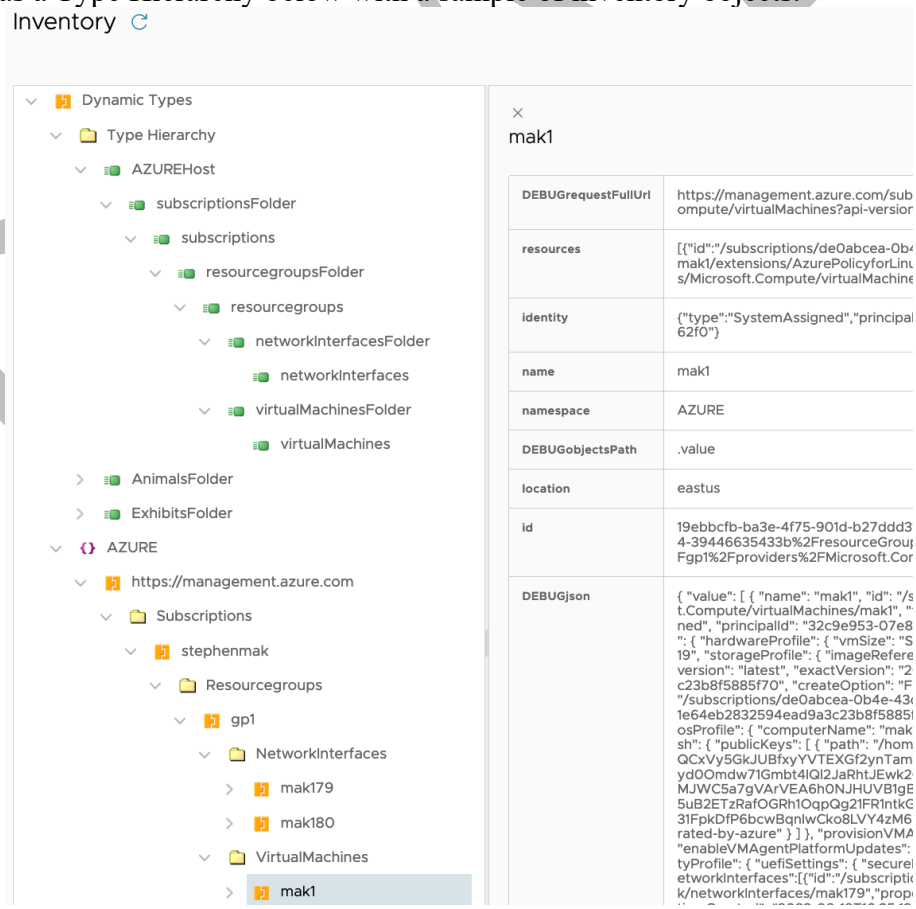
Introduction

This document lists instructions to create a vRO plugin for Azure. We define vRO Dynamic Types to interface with Azure public cloud for the purpose of consuming native cloud services.

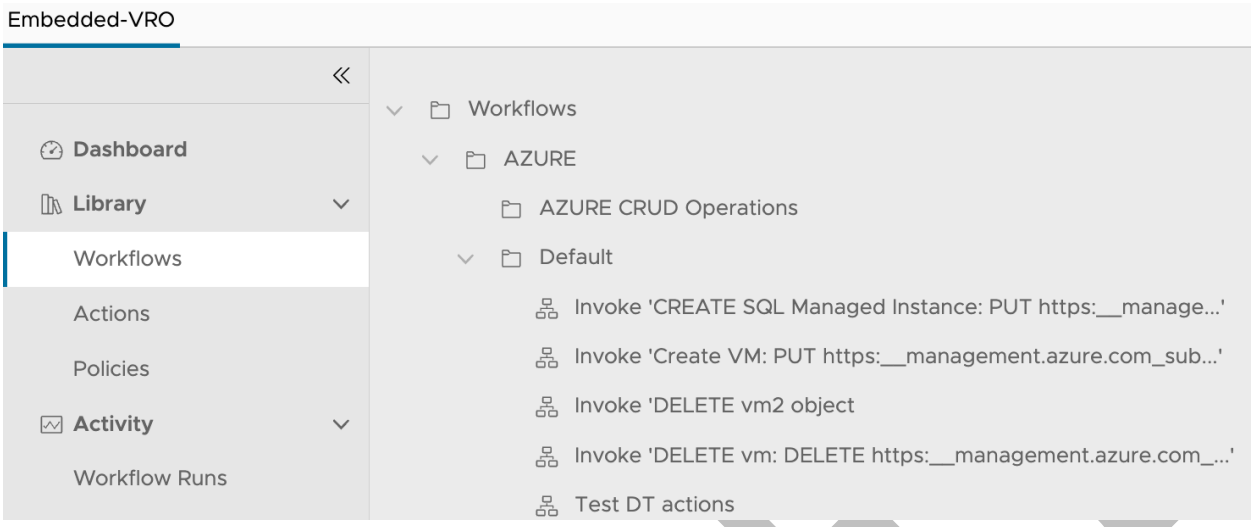
The plugin offers below capabilities.

1) A custom REST header action to authenticate with Azure
2) Collect subscriptions, resource groups, network interfaces and virtual machines via REST calls and store them in vRO as inventory objects
3) A workflow to create a virtual machine using input form actions to look up required resources
4) A workflow to delete a virtual machine with the VM object as input
5) A workflow to delete a virtual machine with a VM name, a subscription ID and a resource group name as inputs
6) Instruction to convert a REST call to a vRO workflow. Our tool creates a vRO workflow with inputs and Azure authentication.

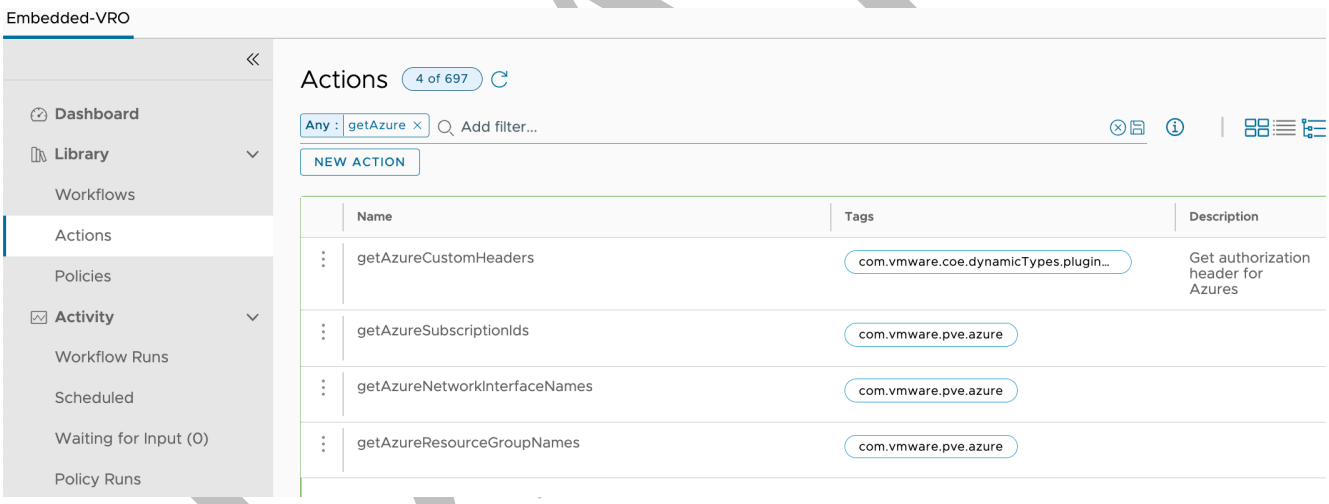
The plugin has a Type Hierarchy below with a sample of inventory objects.



The plugin stores pre-defined workflows in the AZURE/Default folder.

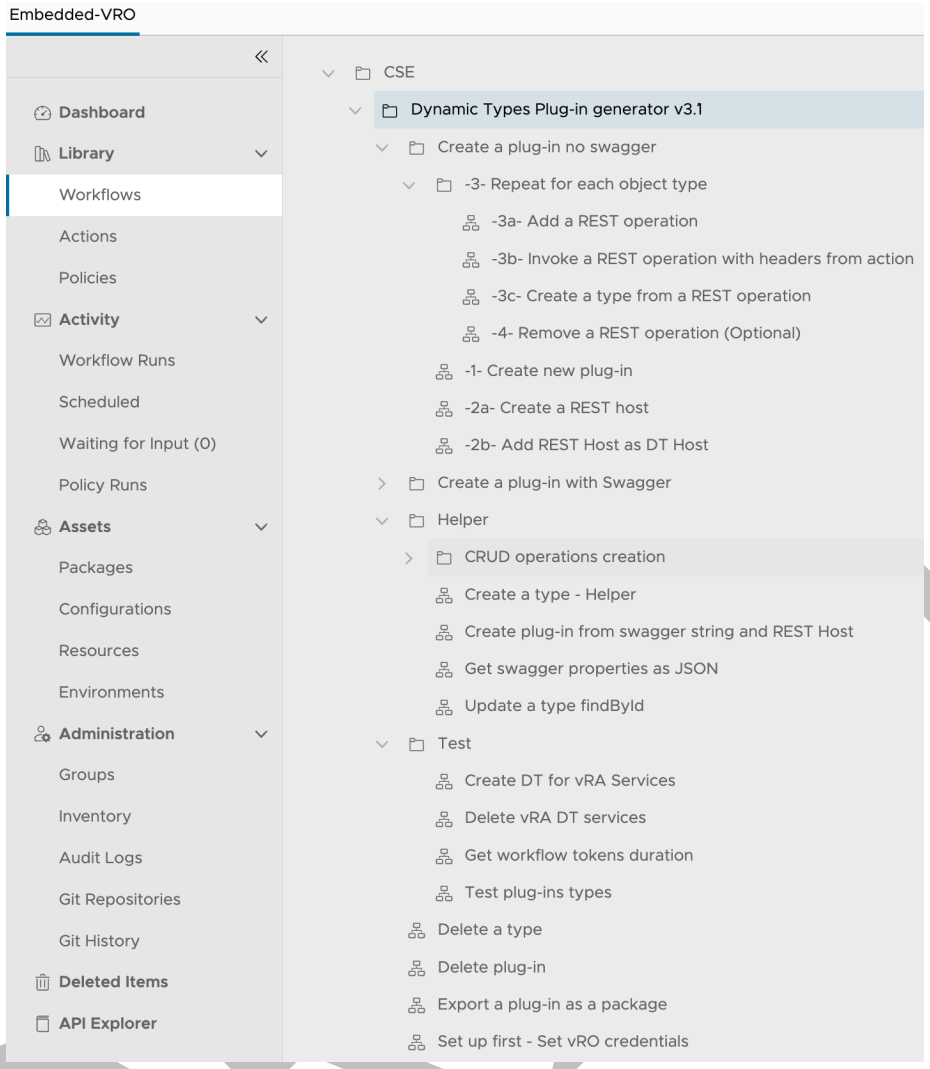


The plugin stores pre-defined 'get' actions in two modules, com.vmware.pve.azure and com.vmware.coe.dynamicTypes.pluginGeneratorV3.



Dynamic Type Plug-in generator

Reader will use Dynamic Type Plug-in generator v3.1 to create Dynamic Types for the plugin. The generator is consisted of a set of workflows and actions.

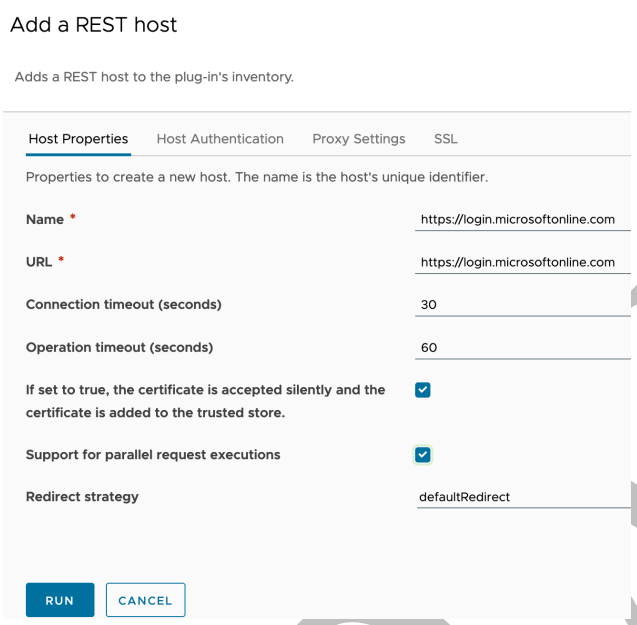

Embedded-VRO
Dashboard
Library
Workflows
Actions
Policies
Activity
Workflow Runs
Scheduled
Waiting for Input (0)
Policy Runs
Assets
Packages
Configurations
Resources
Environments
Administration
Groups
Inventory
Audit Logs
Git Repositories
Git History
Deleted Items
API Explorer
CSE
Dynamic Types Plug-in generator v3.1
Create a plug-in no swagger
-3- Repeat for each object type
-3a- Add a REST operation
-3b- Invoke a REST operation with headers from action
-3c- Create a type from a REST operation
-4- Remove a REST operation (Optional)
-1- Create new plug-in
-2a- Create a REST host
-2b- Add REST Host as DT Host
Create a plug-in with Swagger
Helper
CRUD operations creation
Create a type - Helper
Create plug-in from swagger string and REST Host
Get swagger properties as JSON
Update a type findById
Test
Create DT for vRA Services
Delete vRA DT services
Get workflow tokens duration
Test plug-ins types
Delete a type
Delete plug-in
Export a plug-in as a package
Set up first - Set vRO credentials

Install

Below are the steps.

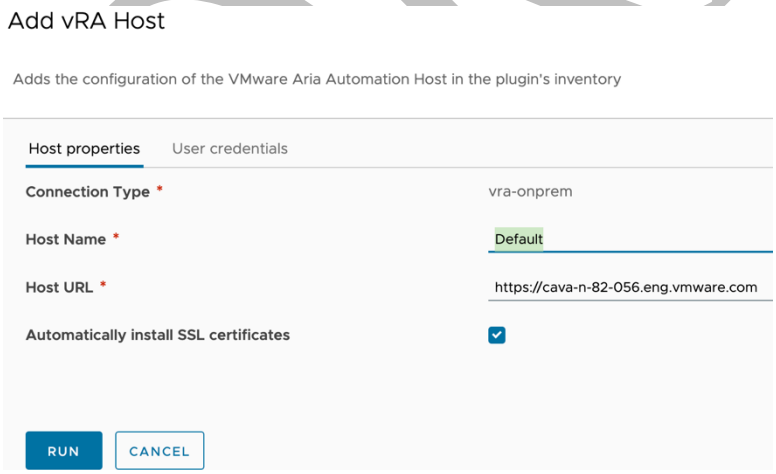
Create REST host – https://login.microsoftonline.com

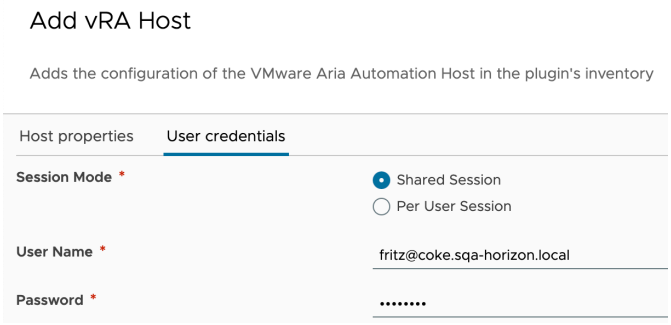
Run below workflow with no authentication to create a REST endpoint for Azure login. Use the input values listed.

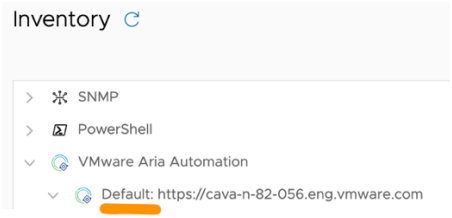


Create vRA host endpoint

Run below workflow with authentication to create a vRA host endpoint. Use 'Default' for Host name.

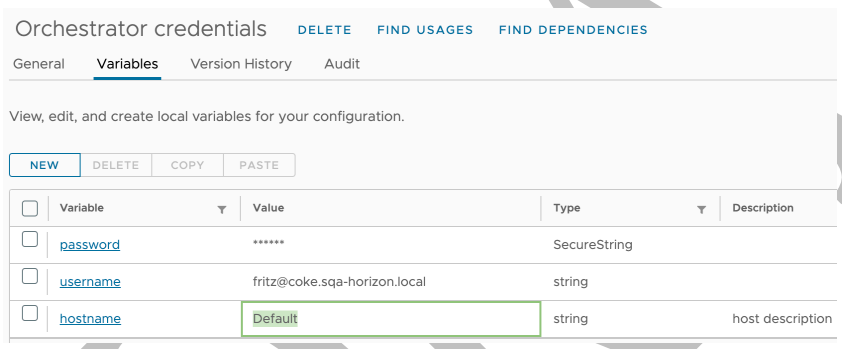


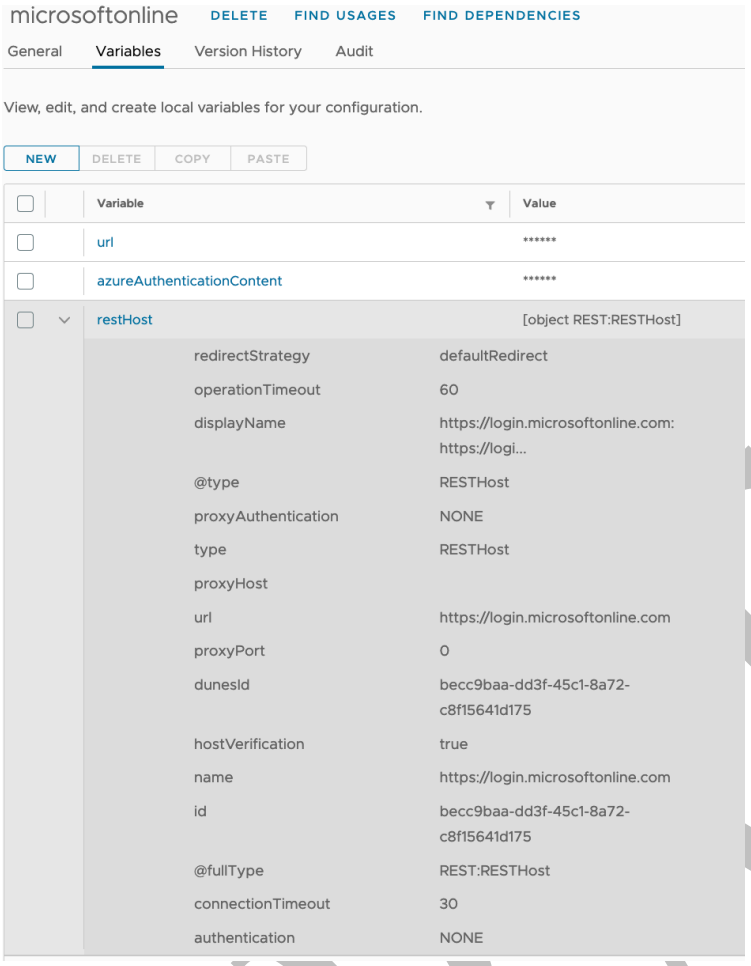




Set Configuration Elements

Update below values for your environment. Hostname is 'Default' matching vRA host object name in inventory.





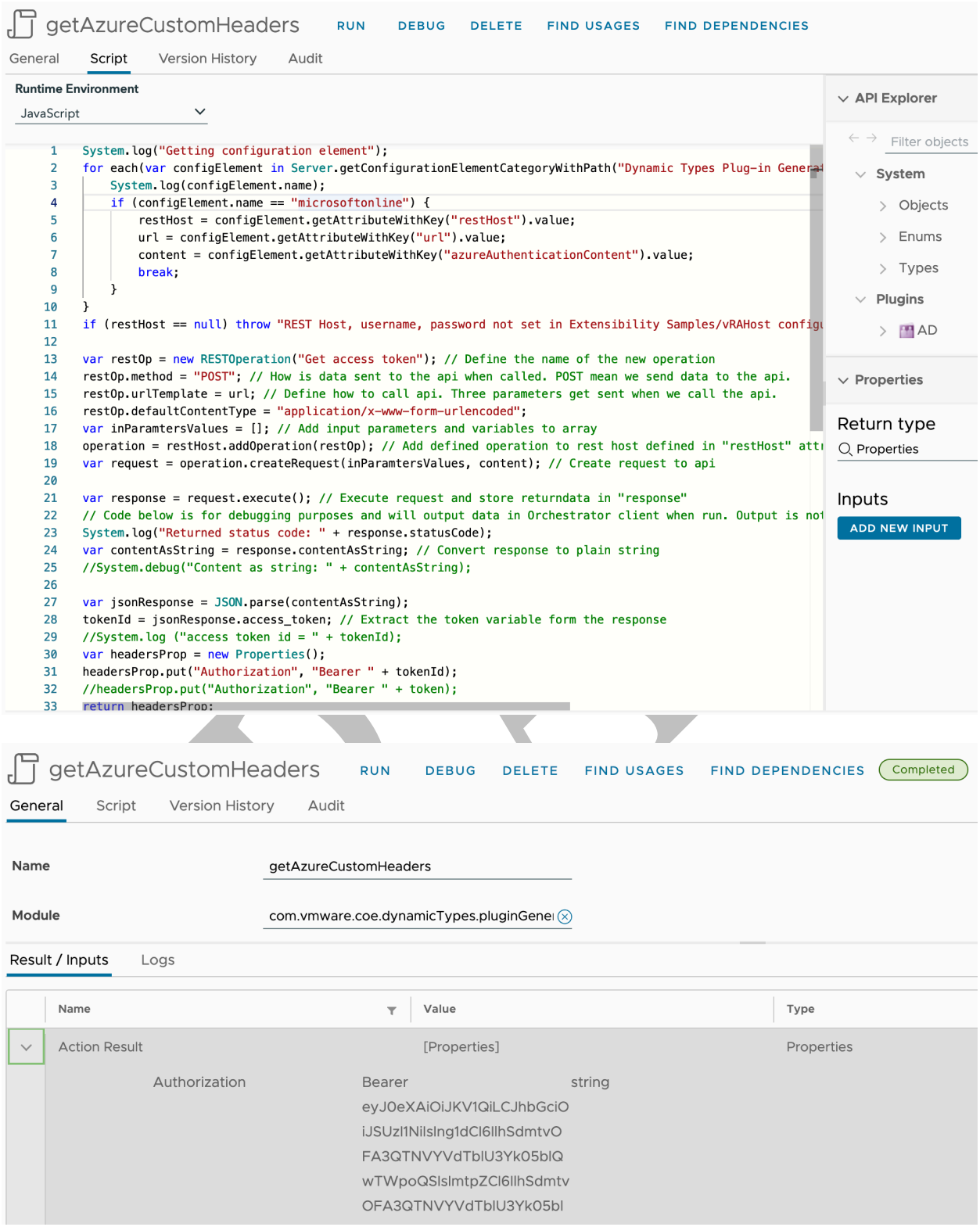
Where url is a string of below format:
`/[TENANT_ID]/oauth2/token`

azureAuthenticationContent is a string of below format:
`grant_type=client_credentials&client_id=[CLIENT_ID]&client_secret=[SECRET]&resource=https://management.azure.com`

See reference section Create Application Secret for detail.

Test getAzureCustomHeaders action

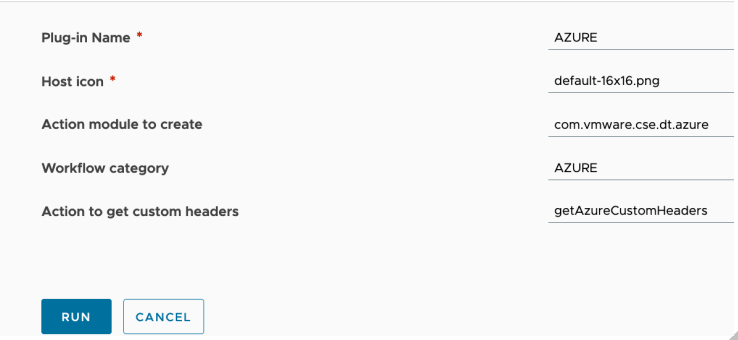
Run below action to obtain a bearer token verifying values of microsoftonline configuration element.



Create namespace – AZURE

Run workflow (1) in the CSE/Dynamic Types Plug-in generator v3 folder to create a namespace. Use the input values below.

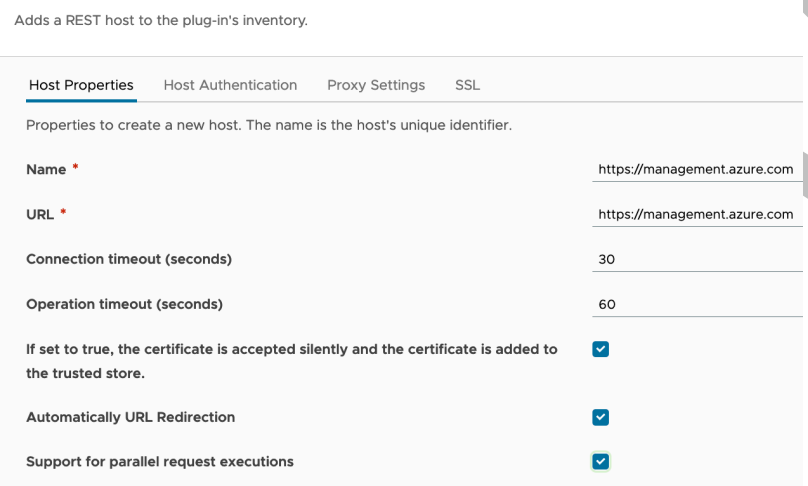
-1- Create new plug-in



Create REST host - https://management.azure.com

Run workflow (2a) with no authentication to add the AZURE management host.

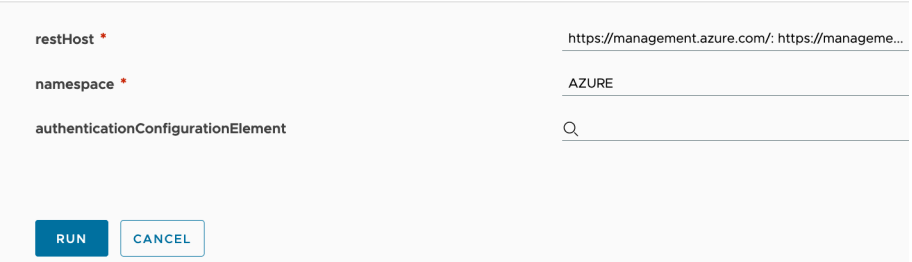
-2a- Create a REST host



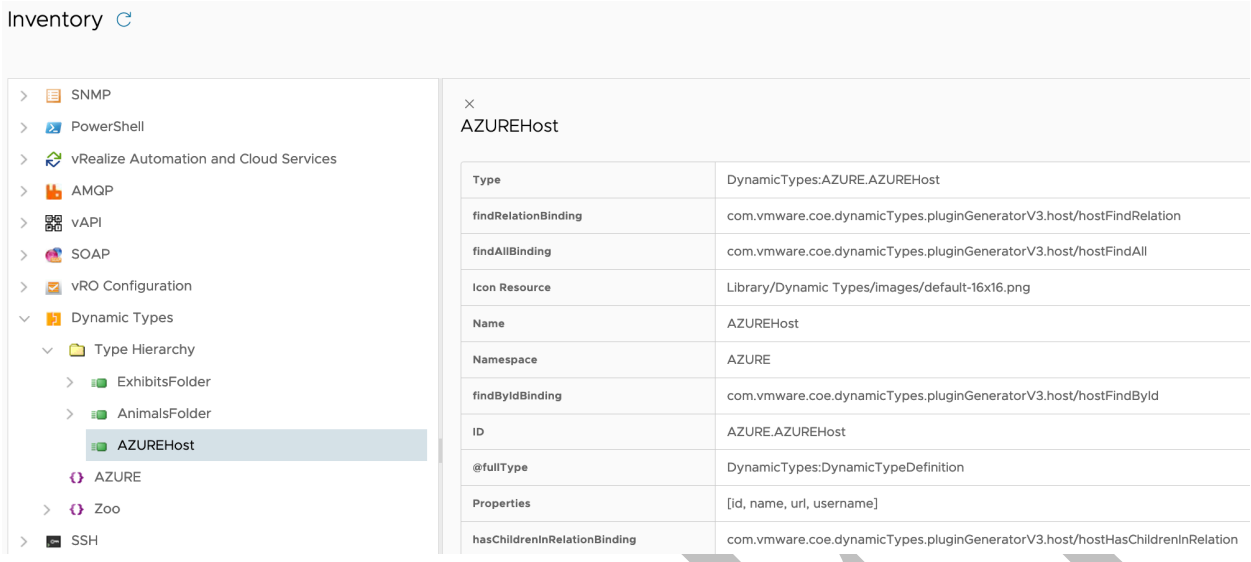
Create DT Host from REST Host – AZUREHost

Run workflow (2b) to create a Dynamic Type (DT) host. Use the input values below.

-2b- Add REST Host as DT Host

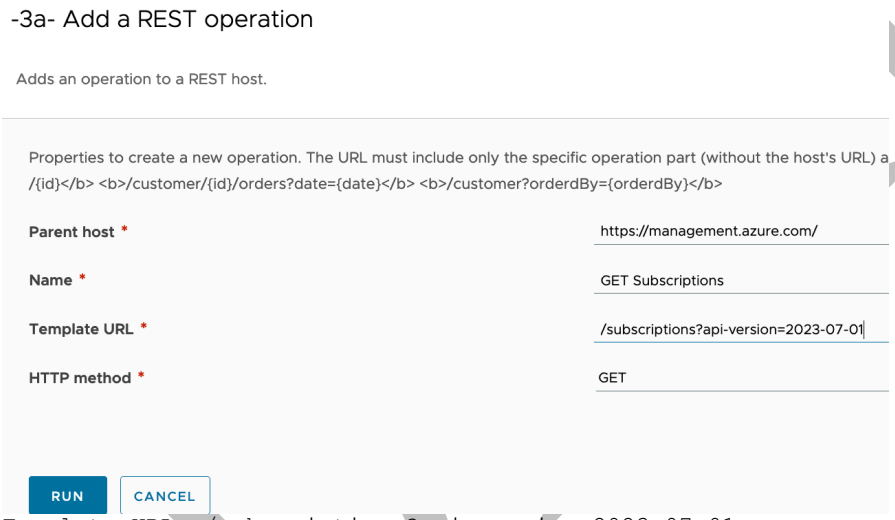


Examine AZUREHost in Inventory



GET Subscriptions

Run workflow (3a) to add the REST operation. Use the input values below.

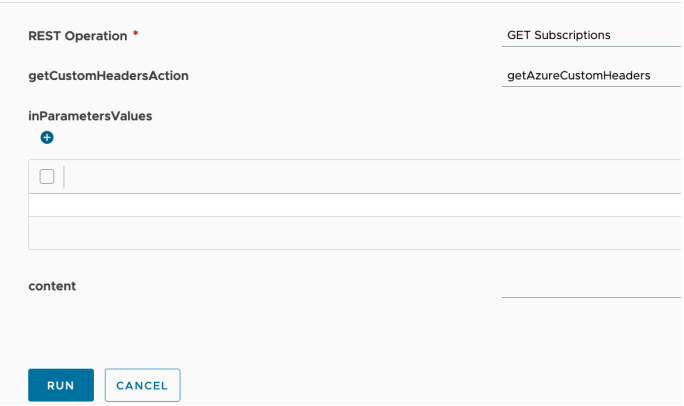



```
Template URL: /subscriptions?api-version=2023-07-01
```

Test REST Operation

Run workflow (3b) to test the operation.

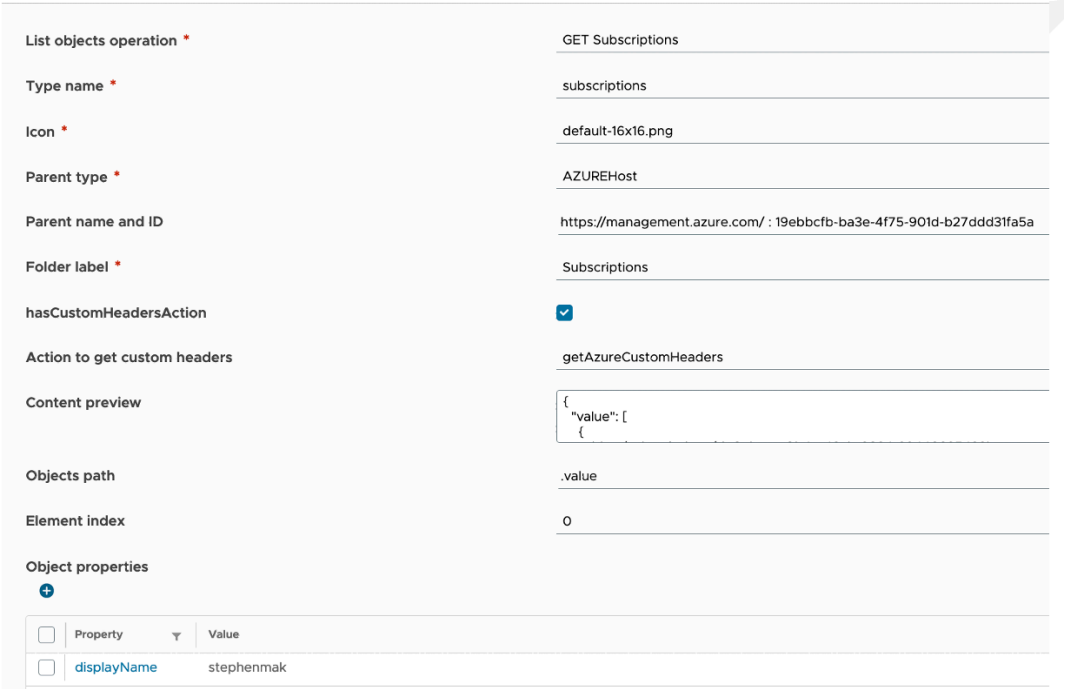
-3b- Invoke a REST operation with headers from action

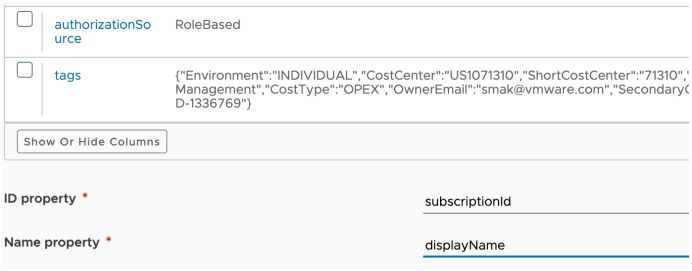


Create subscriptions type from REST operation

Run workflow (3c) to create a dynamic type for the REST operation. Use the input values below. Map ID property to subscriptionId of REST response. Map Name property to displayName.

-3c- Create a type from a REST operation

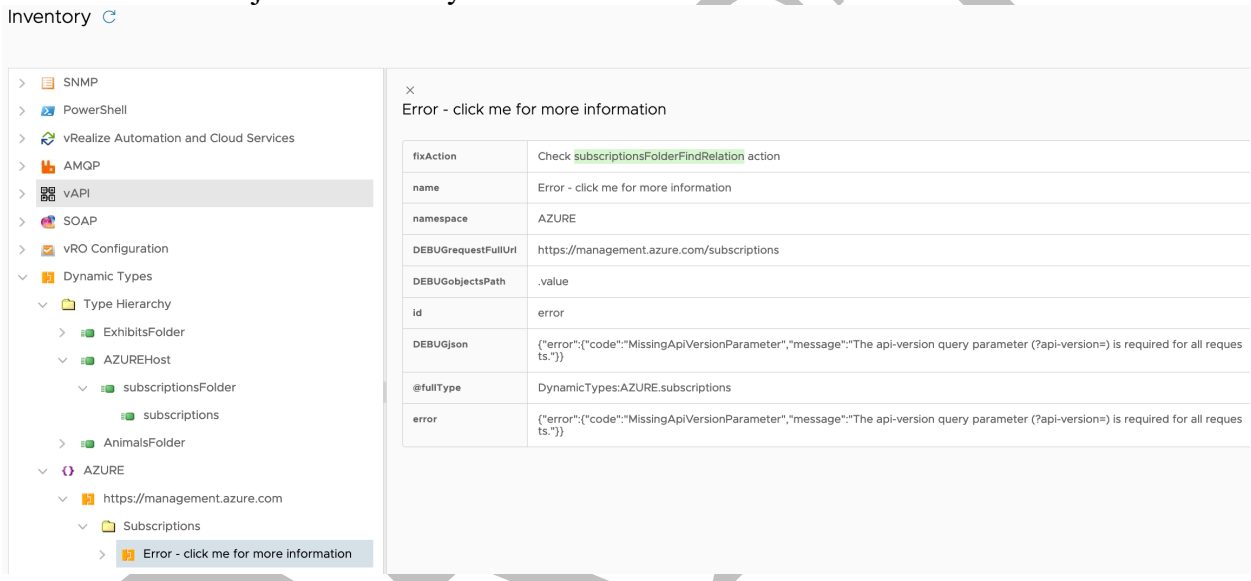




In case of an error, correct the error, delete the type and its actions then retry. Type delete instruction is provided in a section below.

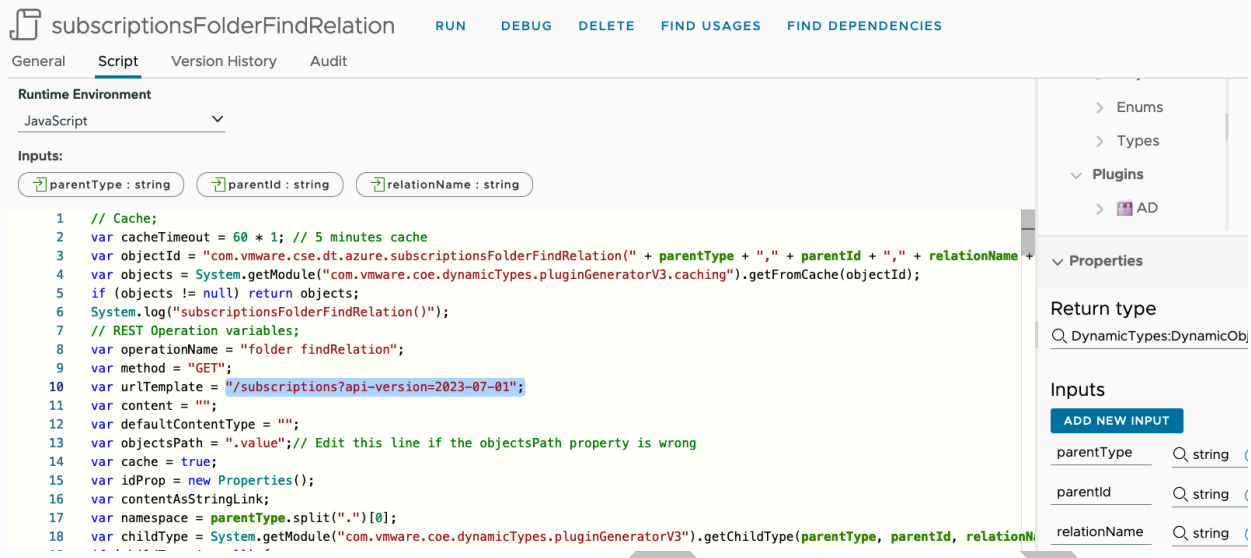
Examine Subscriptions in inventory

Examine the DT object in inventory for error.



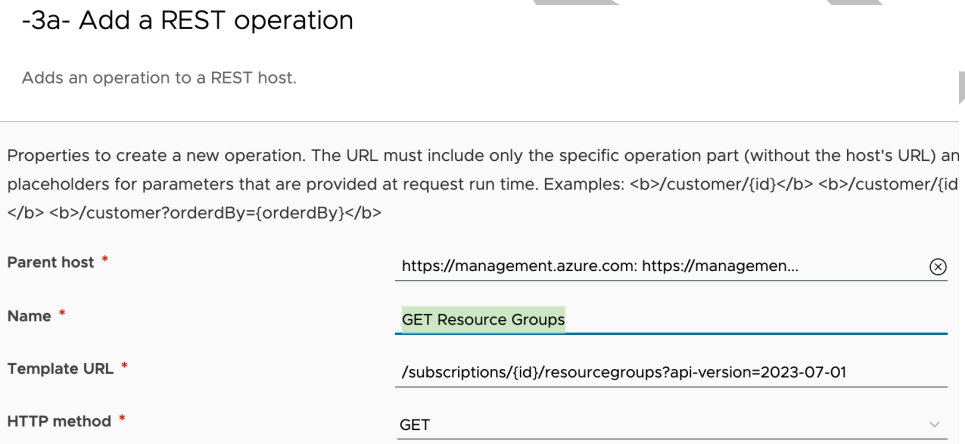
Update subscriptionsFolderFindRelation action

Error: MissingApiVersionParameter
Solution: update urlTemplate as below



GET Resource Groups

Run workflow (3a) to add the REST operation. Use the input values below. Note: an input parameter {id} is used in URL.

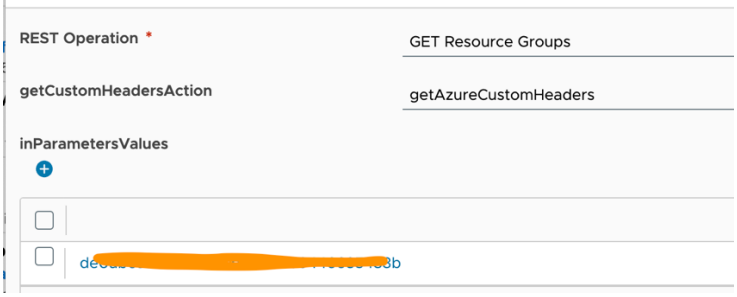



```
Template URL: /subscriptions/{id}/resourcegroups?api-version=2023-07-01
```

Test REST Operation

Run workflow (3b) to test the operation. Supply a subscription id as input.

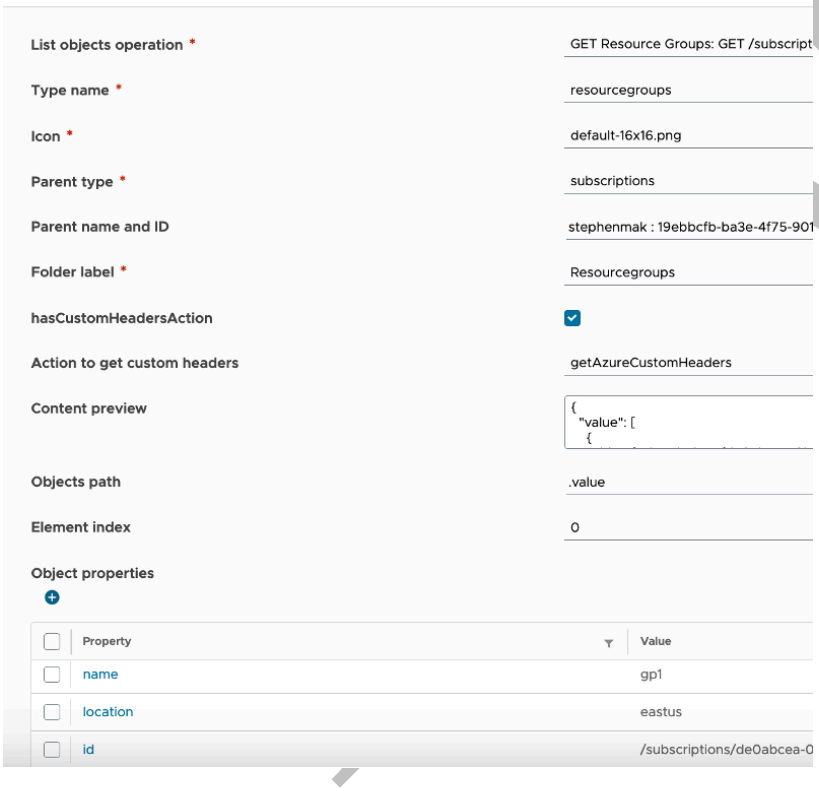
-3b- Invoke a REST operation with headers from action



Create resourcegroups type from REST operation

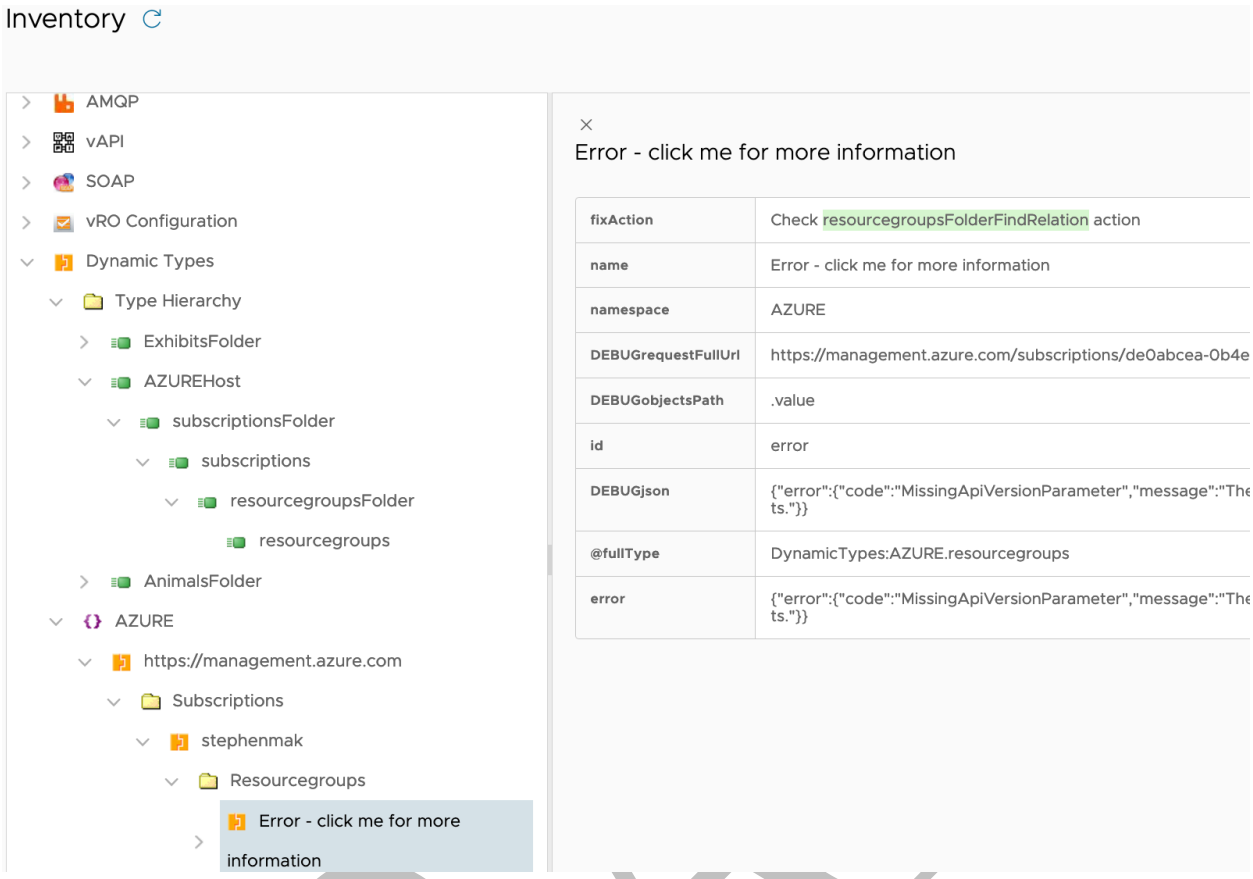
Run workflow (3c) to create a dynamic type for the REST operation. Use the input values below. Update parent type value. Map ID and Name to the corresponding property in REST response.

-3c- Create a type from a REST operation



Examine Resourcegroups in inventory

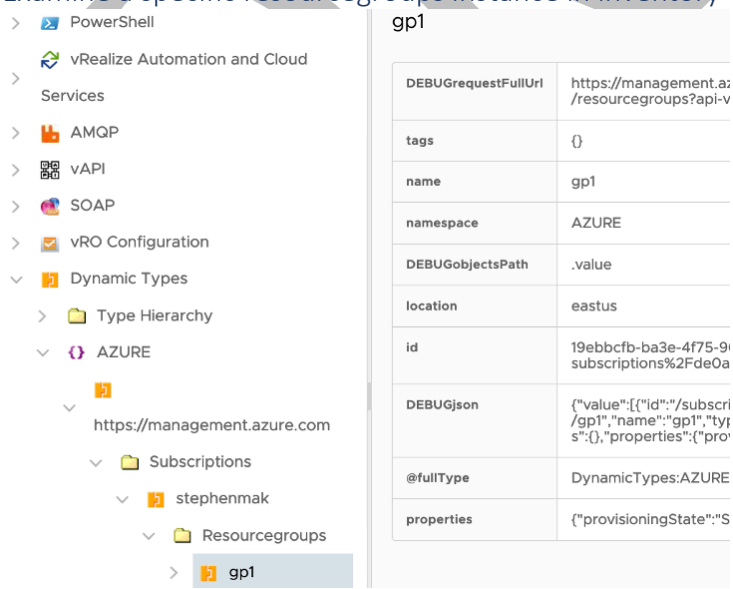
Examine the DT object in inventory for error.



Update resourcegroupsFolderFindRelation action

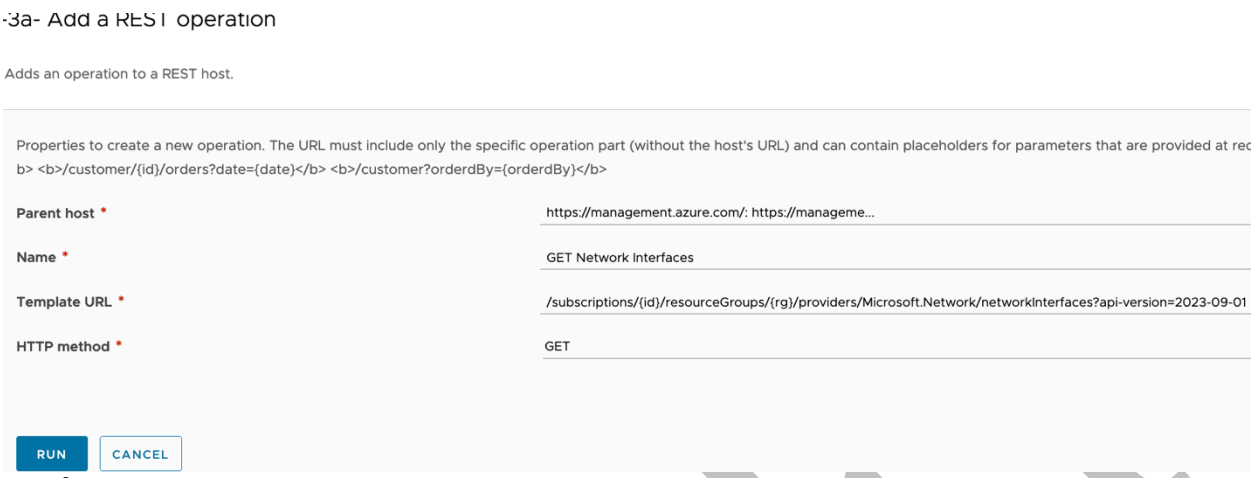
Error: MissingApiVersionParameter
Solution: update urlTemplate

Examine a specific resourcegroups instance in inventory



GET Network Interfaces

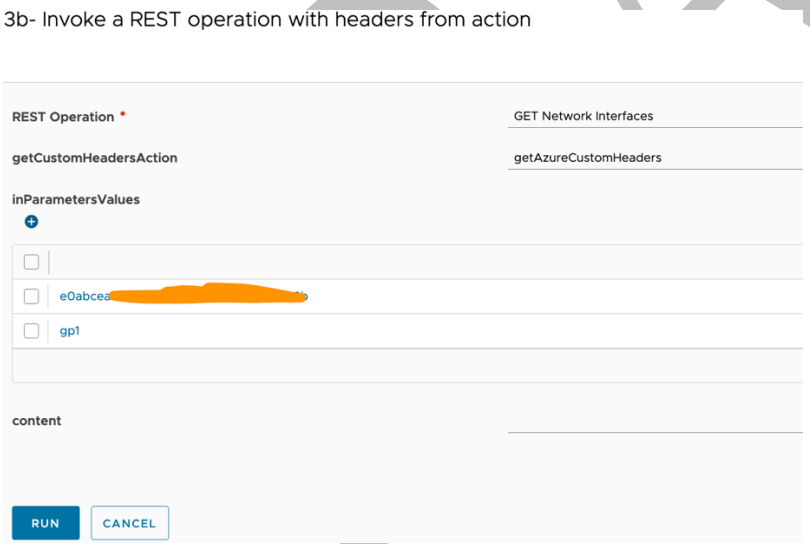
Run workflow (3a) to add the REST operation. Use the input values below. Note: two input parameters, {id} and {rg}, are used in URL.




```
Template URL:
/subscriptions/{id}/resourceGroups/{rg}/providers/Microsoft.Network/networkInterfaces?
api-version=2023-09-01
```

Test REST Operation

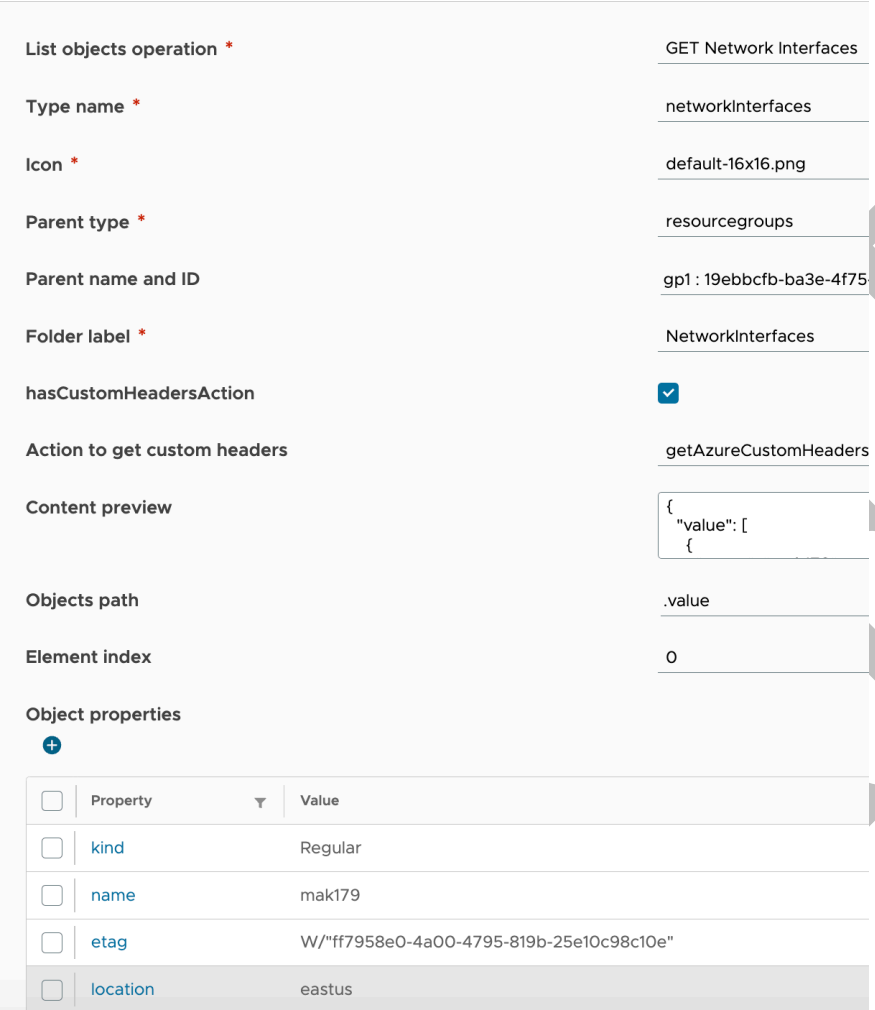
Run workflow (3b) to test the operation. Supply a value to each parameter in the order as listed in URL.



Create interfaces type from REST operation

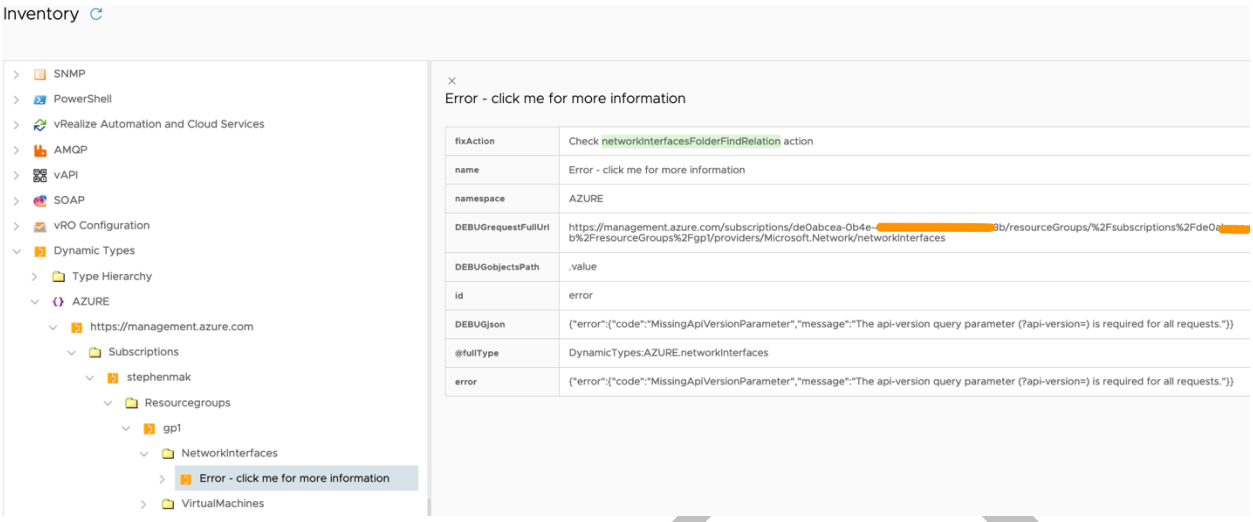
Run workflow (3c) to create a dynamic type for the REST operation. Use the input values below. Update parent type value to resourcegroups. Map ID and Name to the corresponding property in REST response.

-3c- Create a type from a REST operation



Examine NetworkInterfaces in inventory

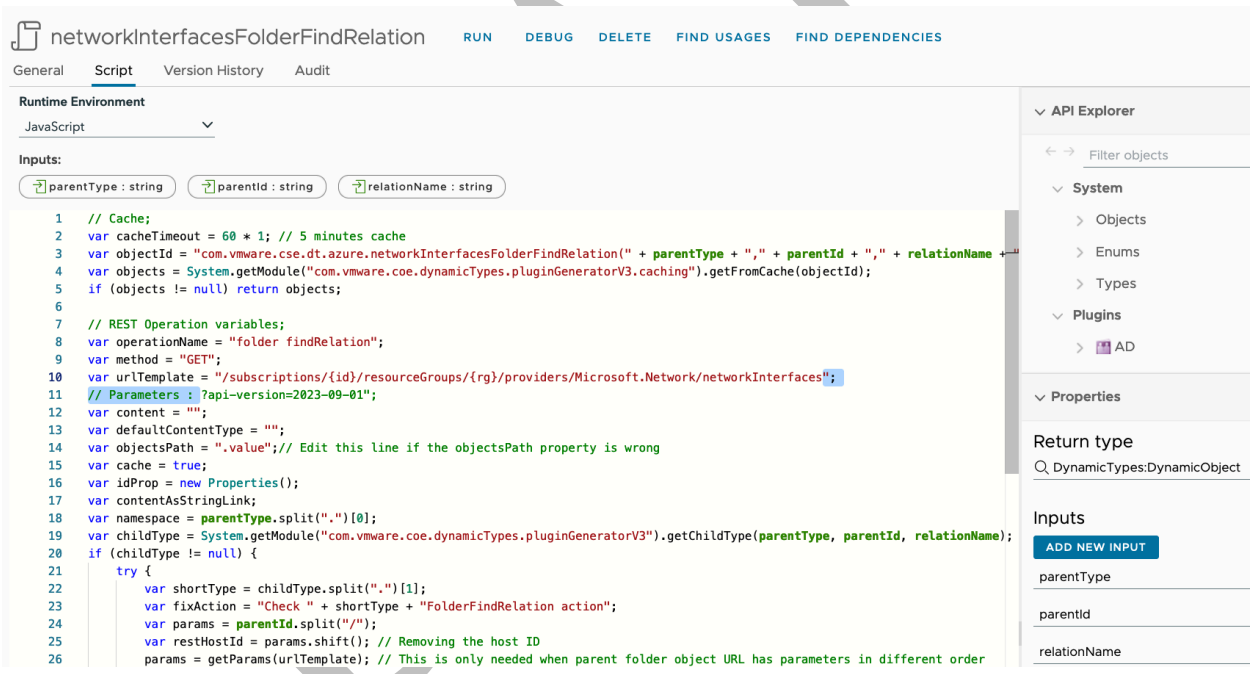
Examine the DT object in inventory for error.



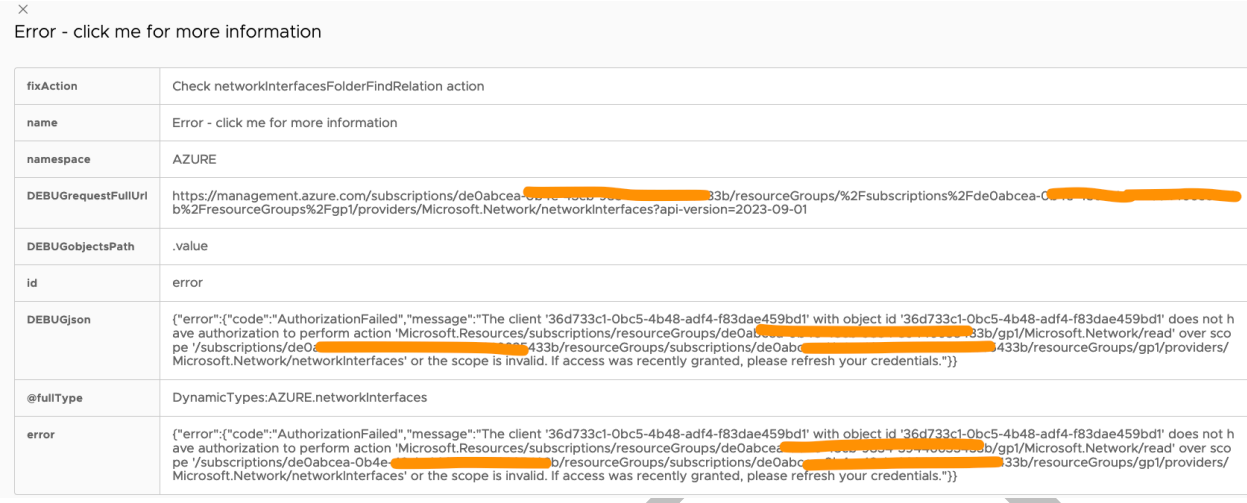
Update networkInterfacesFolderFindRelation action

Error 1: MissingApiVersionParameter

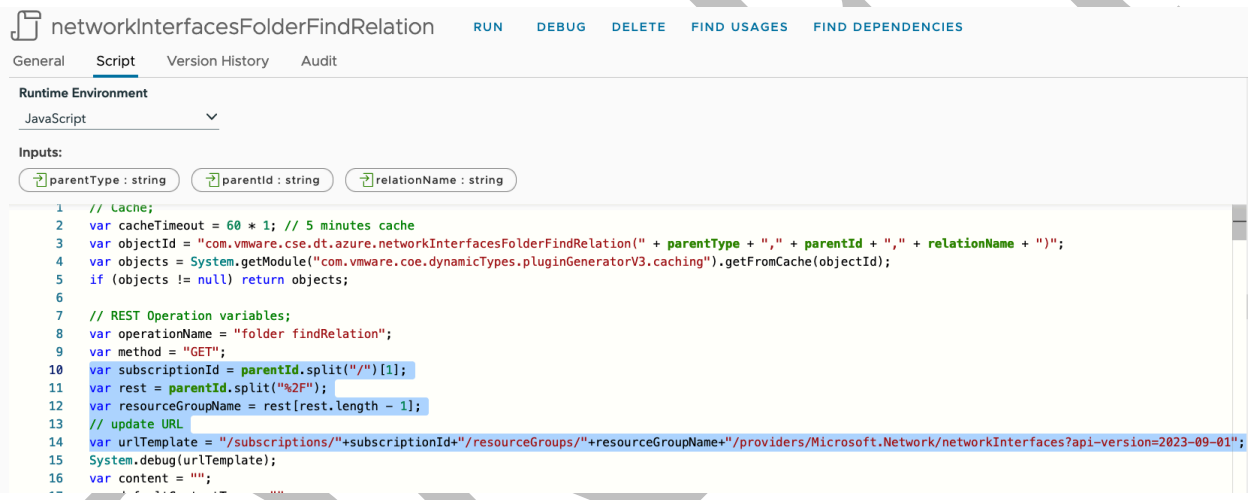
Solution: update urlTemplate



Error 2: invalid subscription Id and resource group name



Solution: Update subscriptionId and resourceGroupName value in action as below




```
// REST Operation variables;
var operationName = "folder findRelation";
var method = "GET";
var subscriptionId = parentId.split("/")[1];
var rest = parentId.split("%2F");
var resourceGroupName = rest[rest.length - 1];
// update URL
var urlTemplate =
"/subscriptions/"+subscriptionId+"/resourceGroups/"+resourceGroupName+"/providers/Micr
osoft.Network/networkInterfaces?api-version=2023-09-01";
System.debug(urlTemplate);
```

GET Virtual Machines

Run workflow (3a) to add the REST operation. Note: two input parameters, {subscriptionId} and {resoureGroupName}, are used in URL.

-3a- Add a REST operation

Adds an operation to a REST host.

Properties to create a new operation. The URL must include only the specific operation part (without the host's URL) and can contain placeholders for parameters that are provided at request run time. Examples: /customer/{id} /customer?orderdBy={orderdBy}

Parent host *	https://management.azure.com
Name *	GET Virtual Machines
Template URL *	/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines?api-version=2023-07-01
HTTP method *	GET

```
Template URL:
/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft
.Compute/virtualMachines?api-version=2023-07-01
```

Create virtualMachines type from REST Operation

Run workflow (3c) to create a dynamic type for the REST operation. Use the input values below. Update parent type value. Map ID and Name to the corresponding property in REST response.

-3c- Create a type from a REST operation

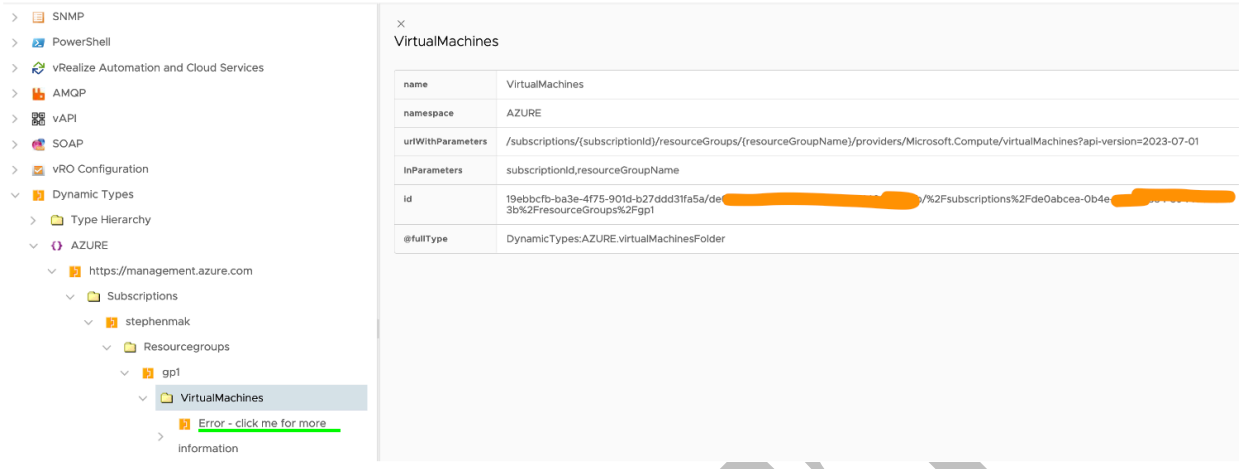
List objects operation *	GET Virtual Machines
Type name *	virtualMachines
Icon *	default-16x16.png
Parent type *	resourcegroups
Parent name and ID	gp1 : 19ebbcfb-ba3[redacted]01d-b27ddd31fa5a/de0a[redacted]/gp1
Folder label *	VirtualMachines
hasCustomHeadersAction	☑
Action to get custom headers	getAzureCustomHeaders
Content preview	{ "value": [{
Objects path	.value
Element index	0

Object properties

Property	Value
identity	{"type":"SystemAssigned","principalId":"32c9e953-07e8-443[redacted]0659a8db","tenantId":"b39138ca-3cee-4[redacted]2f0"}
name	mak1

Examine VirtualMachines in inventory

Examine the DT object in inventory for error.

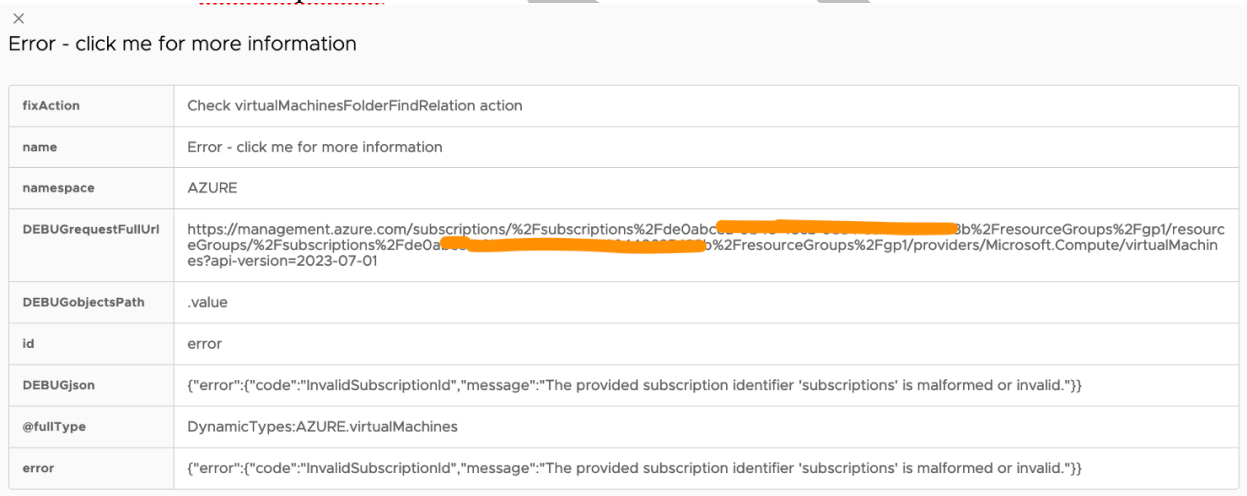


Update vritualMachinesFolderFindRelation action

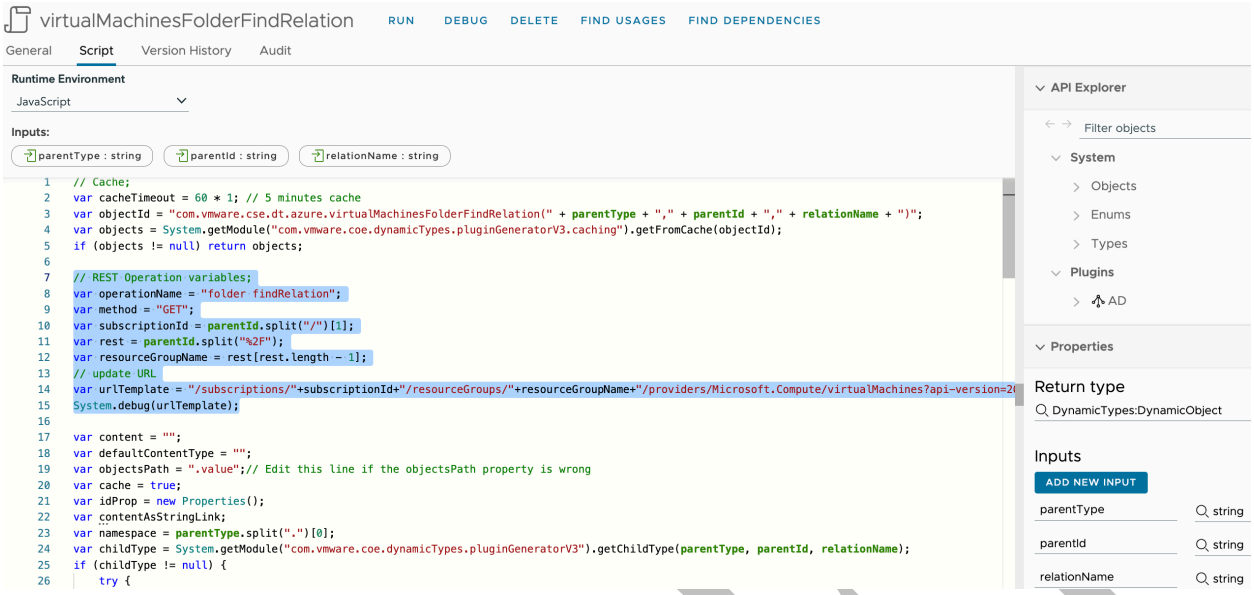
Error 1: MissingApiVersionParameter

Solution: update urlTemplate

Error 2: invalid subscriptionId



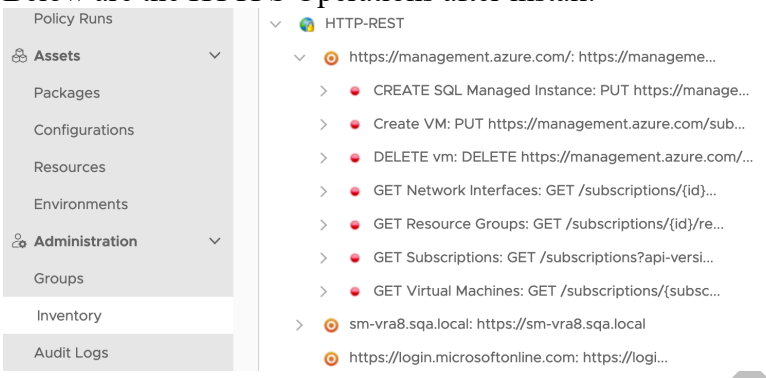
Solution:
Update urlTemplate in virtualMachinesFolderFindRelation action as below.




```
// REST Operation variables;
var operationName = "folder findRelation";
var method = "GET";
var subscriptionId = parentId.split("/")[1];
var rest = parentId.split("%2F");
var resourceGroupName = rest[rest.length - 1];
// update URL
var urlTemplate =
"/subscriptions/"+subscriptionId+"/resourceGroups/"+resourceGroupName+"/providers/Micr
osoft.Compute/virtualMachines?api-version=2023-07-01";
System.debug(urlTemplate);
```

HTTPS Operations

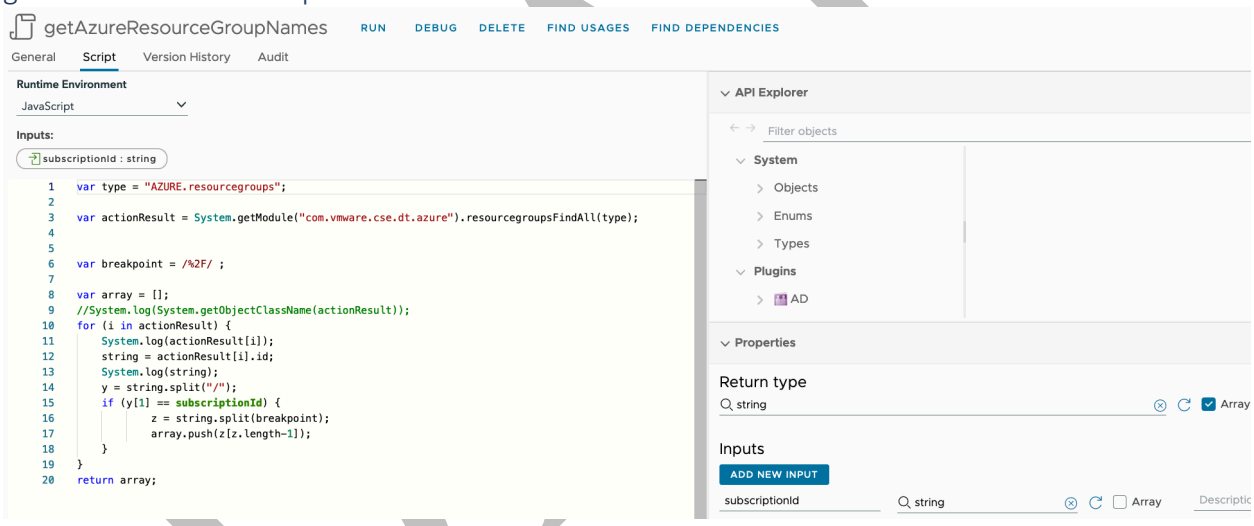
Below are the HTTPS Operations after install.



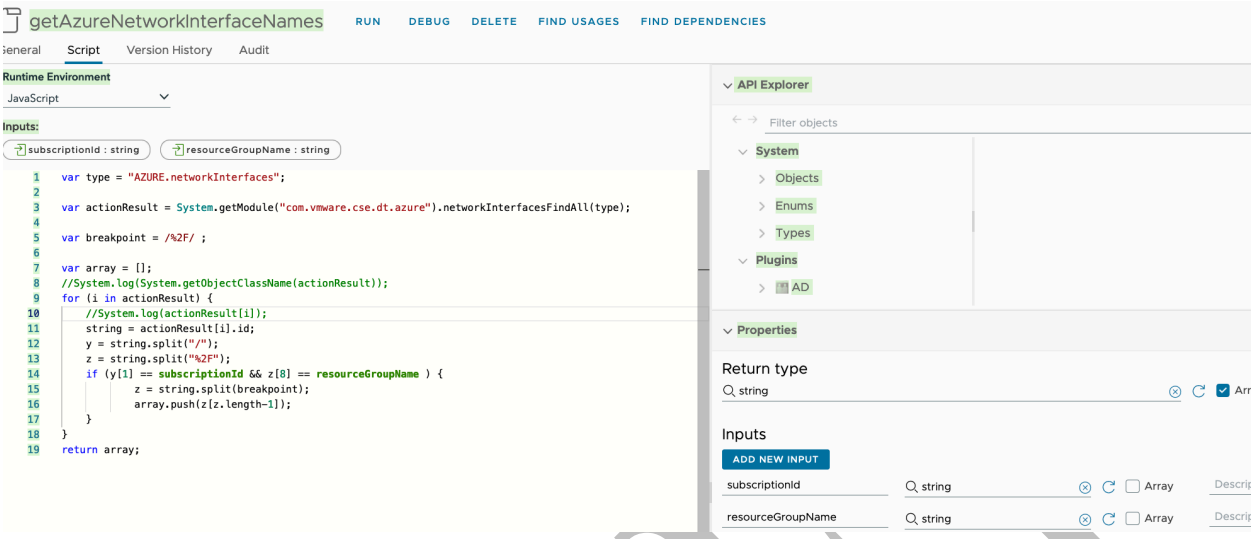
Input Form Actions

com.vmware.pve.azure module has pre-defined input form actions. Reader can run them with no changes.

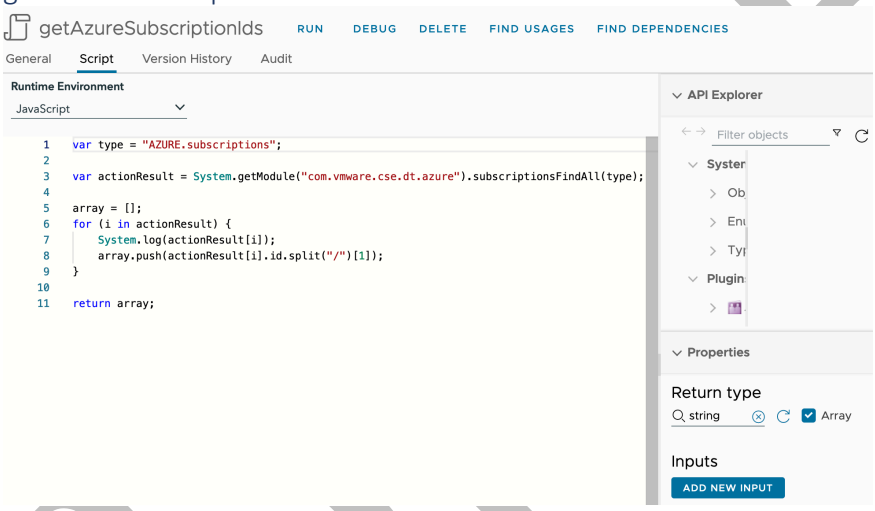
getAzureResourceGroupNames action



getAzureNetworkInterfaceNames action



getAzureSubscriptionIds action



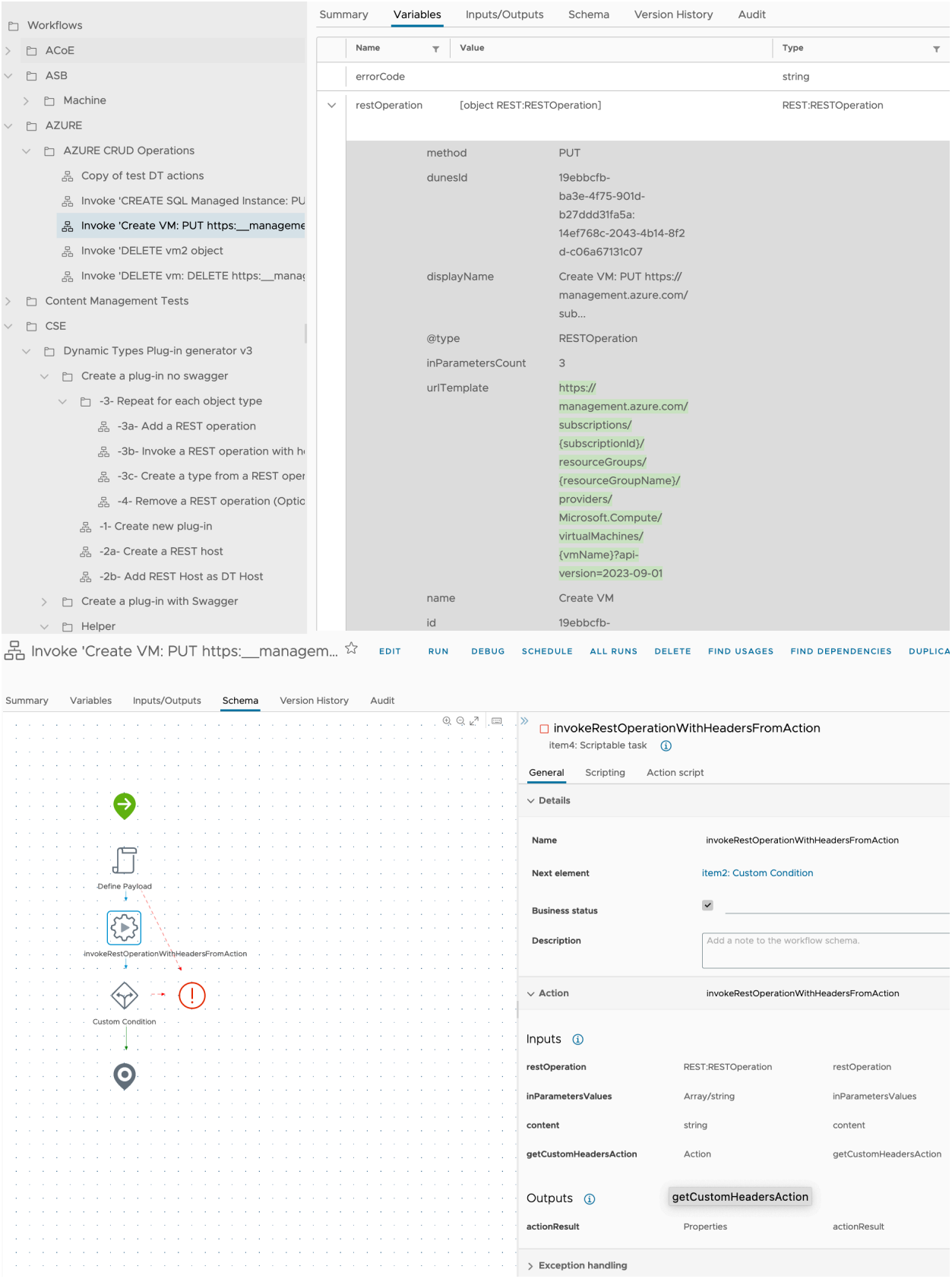
Provisioning Workflows

Folder AZURE/Default contains pre-defined workflows. Reader needs create a REST operation and assign it to restOperation parameter in workflow before use. In addition, reader may need re-attach input form actions and field inputs to workflow.

Invoke 'Create VM: PUT https:__management.azure.com_sub.

Use below url to create a REST operation:

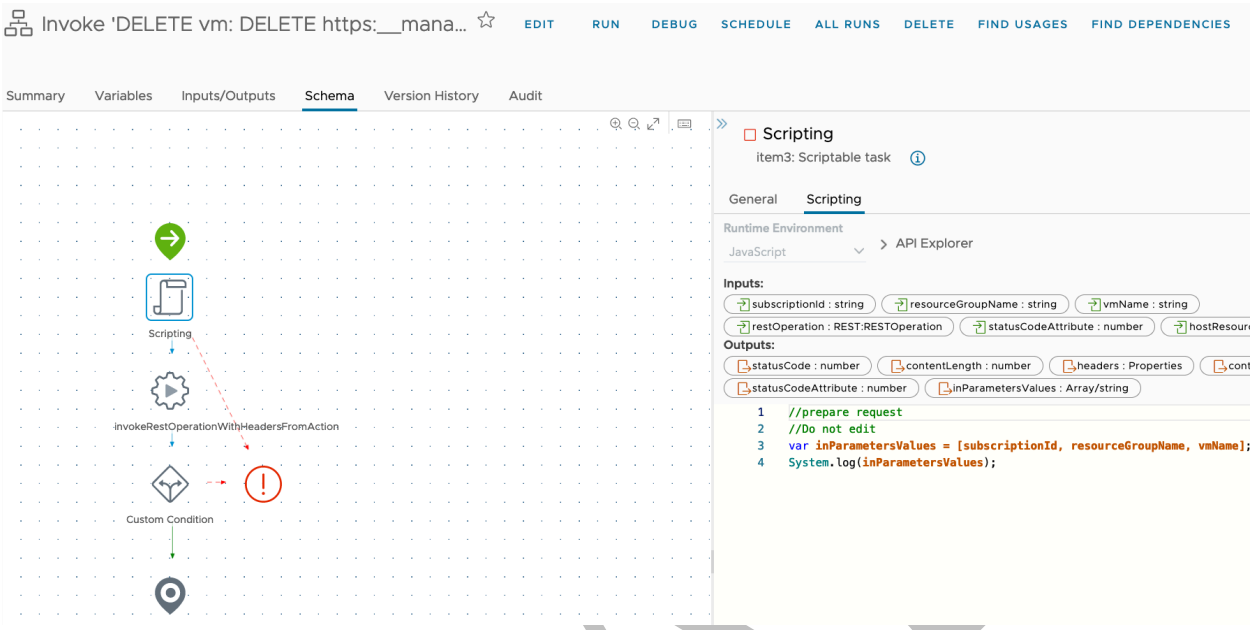
```
PUT
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2023-09-01
```


Workflows
ACoE
ASB
Machine
AZURE
AZURE CRUD Operations
Copy of test DT actions
Invoke 'CREATE SQL Managed Instance: PU
Invoke 'Create VM: PUT https:__manageme
Invoke 'DELETE vm2 object
Invoke 'DELETE vm: DELETE https:__manag
Content Management Tests
CSE
Dynamic Types Plug-in generator v3
Create a plug-in no swagger
-3- Repeat for each object type
-3a- Add a REST operation
-3b- Invoke a REST operation with h
-3c- Create a type from a REST oper
-4- Remove a REST operation (Optic
-1- Create new plug-in
-2a- Create a REST host
-2b- Add REST Host as DT Host
Create a plug-in with Swagger
Helper
Summary
Variables
Inputs/Outputs
Schema
Version History
Audit
Name
Value
Type
errorCode
string
restOperation
[object REST:RESTOperation]
REST:RESTOperation
method
PUT
dunesId
19ebbcfb-ba3e-4f75-901d-b27ddd31fa5a:14ef768c-2043-4b14-8f2d-c06a67131c07
displayName
Create VM: PUT https://management.azure.com/sub...
@type
RESTOperation
inParametersCount
3
urlTemplate
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2023-09-01
name
Create VM
id
19ebbcfb-
Invoke 'Create VM: PUT https:__managem...
EDIT
RUN
DEBUG
SCHEDULE
ALL RUNS
DELETE
FIND USAGES
FIND DEPENDENCIES
DUPLICA
Summary
Variables
Inputs/Outputs
Schema
Version History
Audit
Define Payload
invokeRestOperationWithHeadersFromAction
Custom Condition
invokeRestOperationWithHeadersFromAction
item4: Scriptable task
General
Scripting
Action script
Details
Name
invokeRestOperationWithHeadersFromAction
Next element
item2: Custom Condition
Business status
Description
Add a note to the workflow schema.
Action
invokeRestOperationWithHeadersFromAction
Inputs
restOperation
REST:RESTOperation
restOperation
inParametersValues
Array/string
inParametersValues
content
string
content
getCustomHeadersAction
Action
getCustomHeadersAction
Outputs
getCustomHeadersAction
actionResult
Properties
actionResult
Exception handling

Invoke 'DELETE vm: DELETE https:__managem

Use below url to create a REST operation:

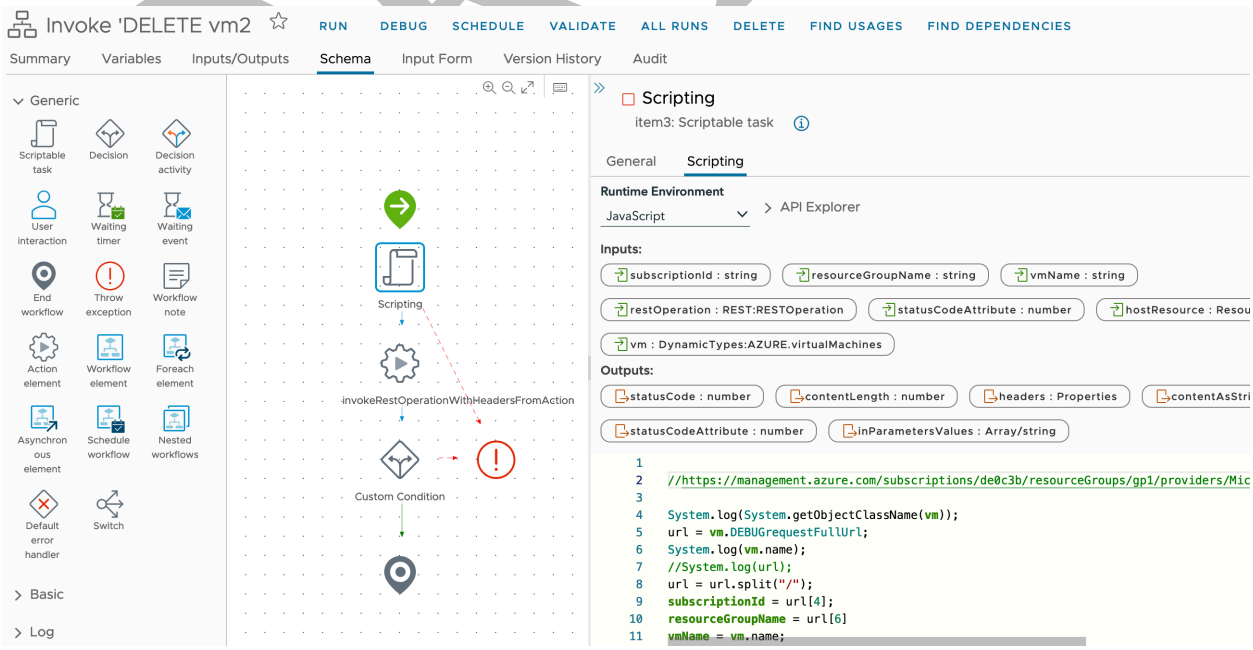
```
DELETE
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGr
oupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2023-09-01
```



Invoke 'DELETE vm2

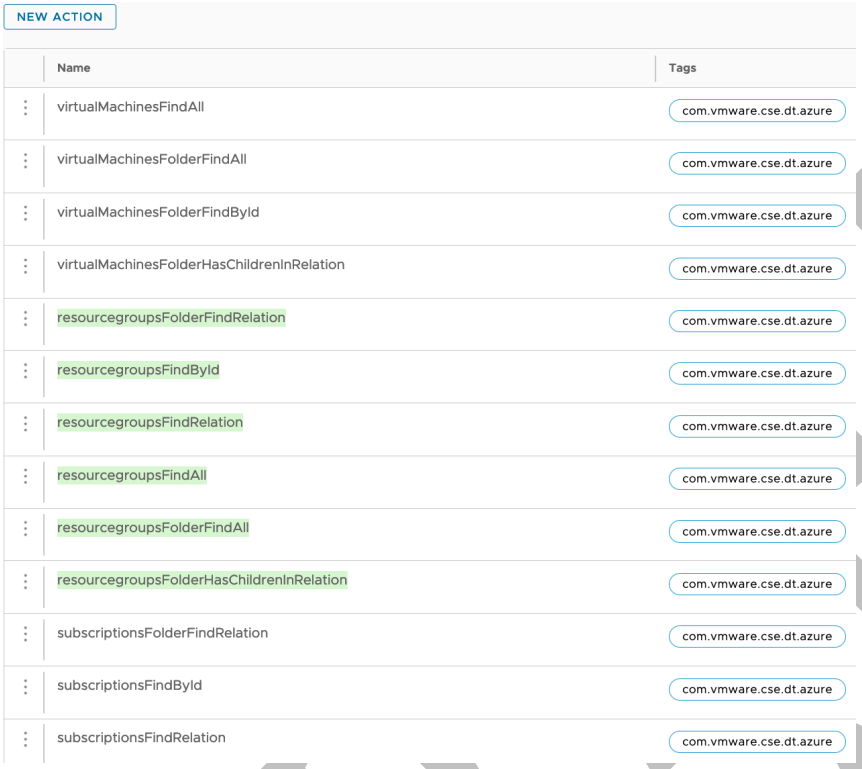
Use below url to create a REST operation:

```
DELETE
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGr
oupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2023-09-01
```



com.vmware.cse.dt.azure module

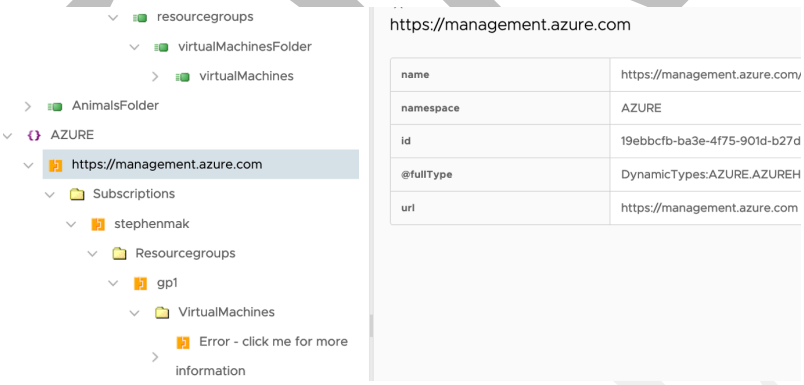
The tool auto-generates a list of plugin actions. Table below highlights all actions of resourcegroups Dynamic Type.



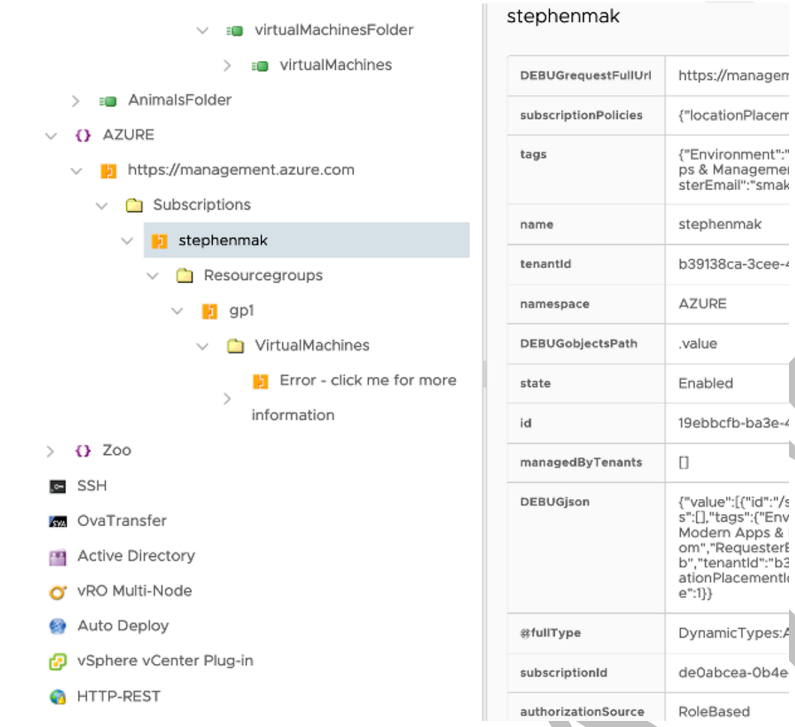
Inventory Objects and Properties

Below are some inventory objects after install.

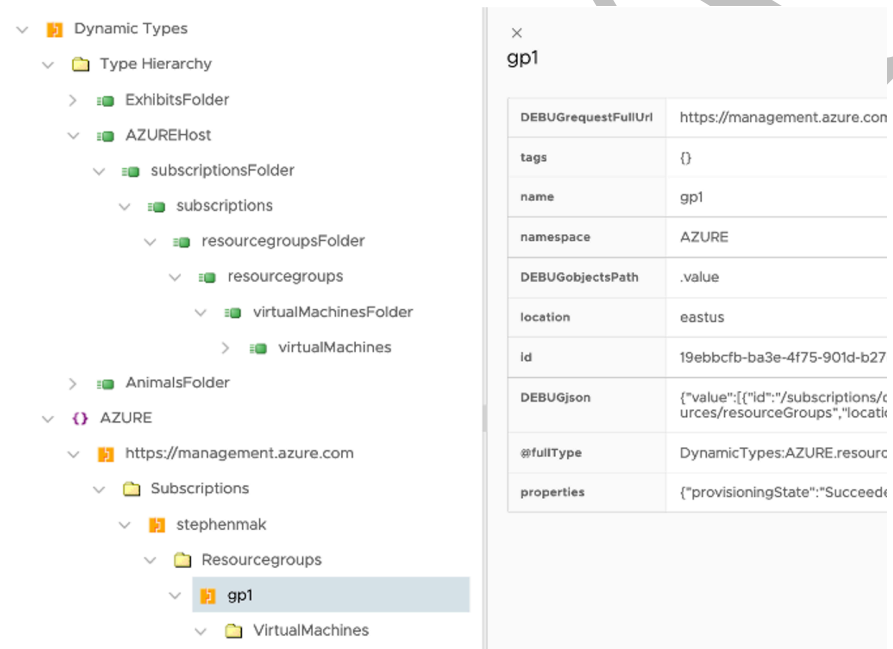
DT Host



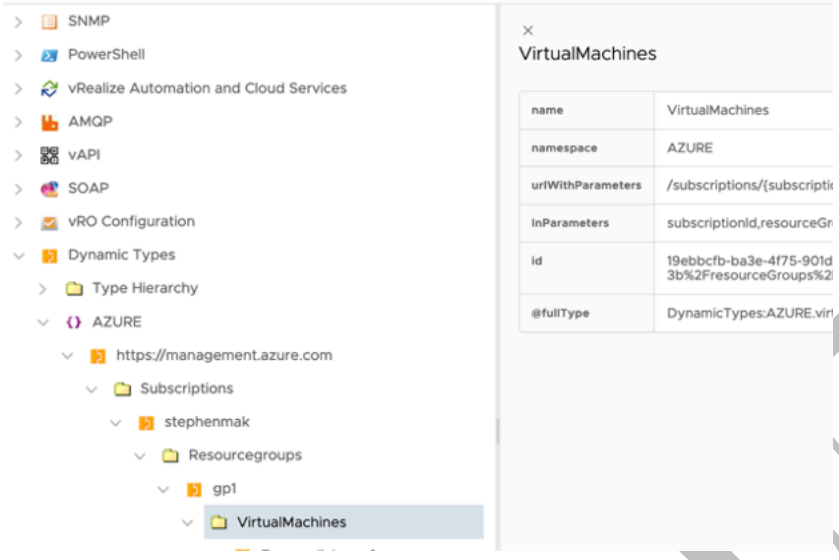
Subscriptions



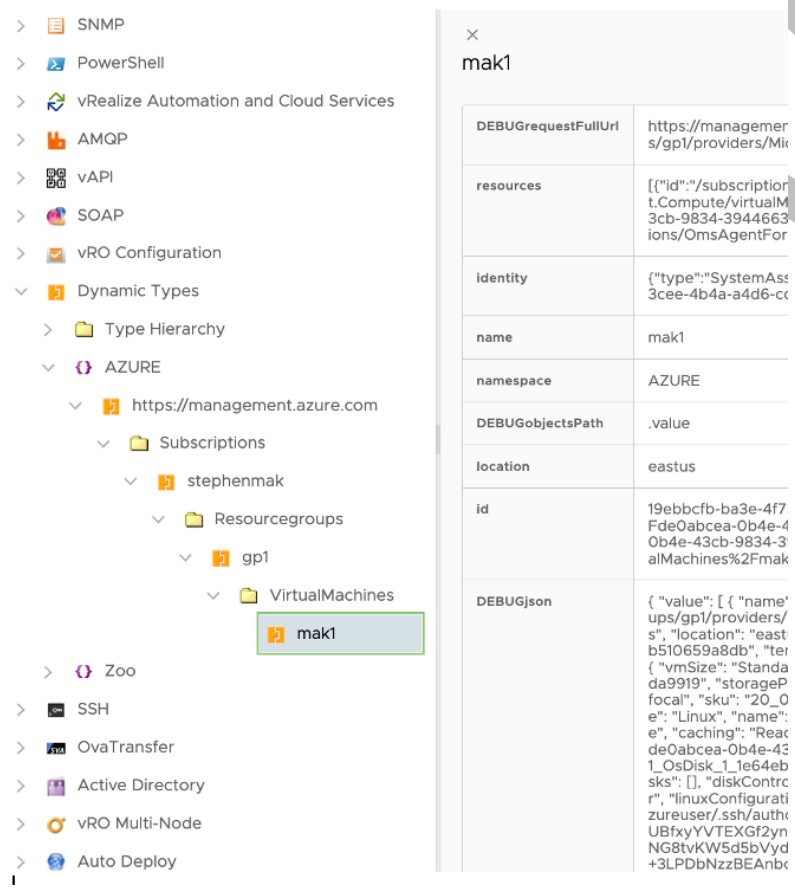
Resource groups



VirtualMachines



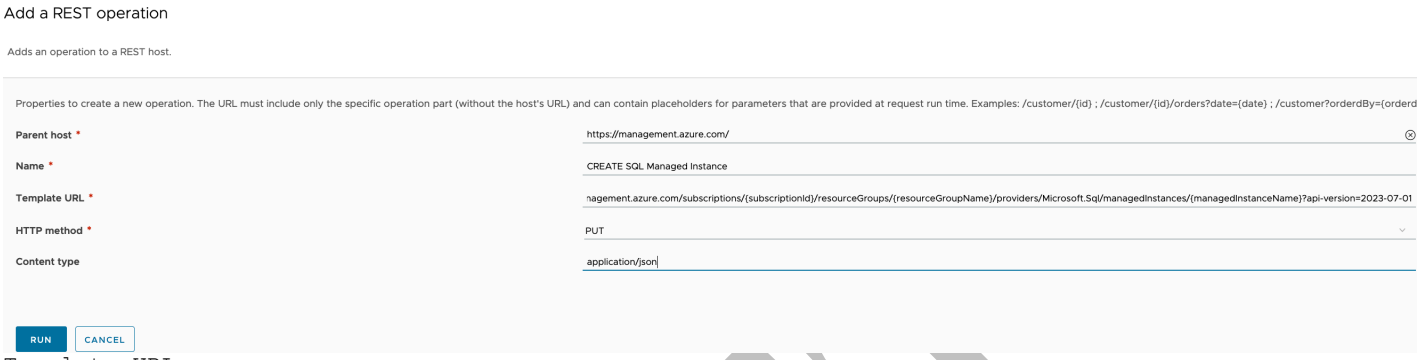
Virtual Machine Instance



Add additional REST Operations

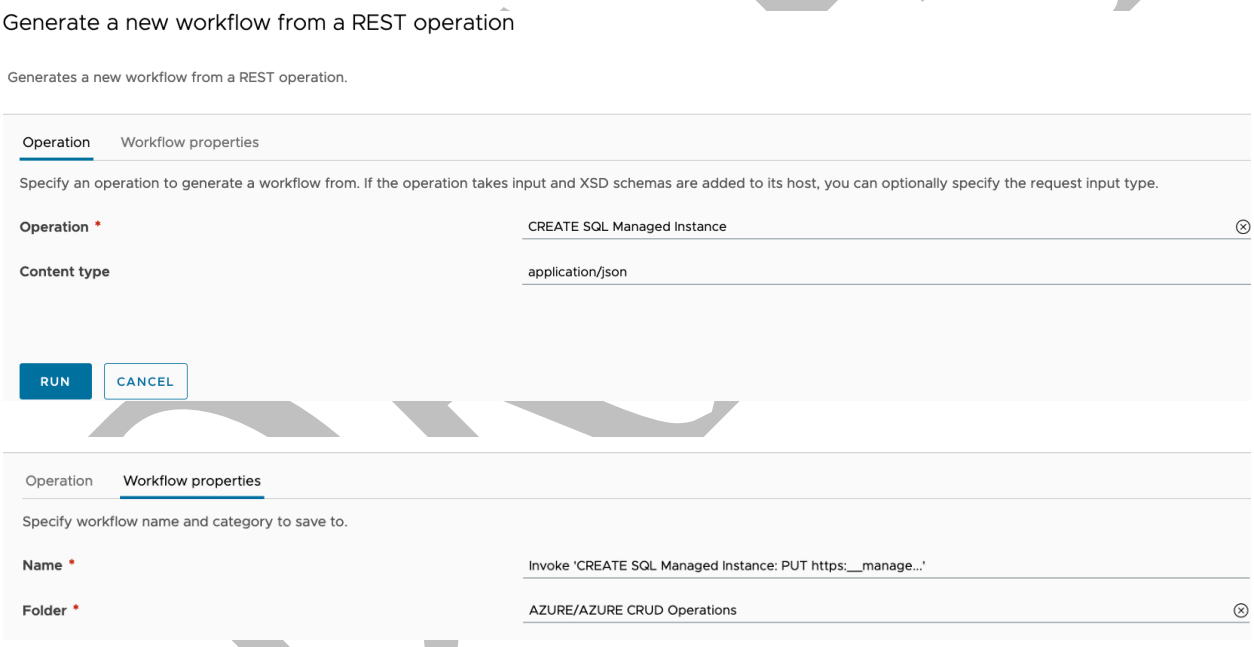
Reader can map a REST call to a vRO workflow. Our tool creates a vRO workflow with inputs and Azure authentication. Below are the instructions.

a) Run below workflow to add a REST Operation. Note: three input parameters are used in URL.

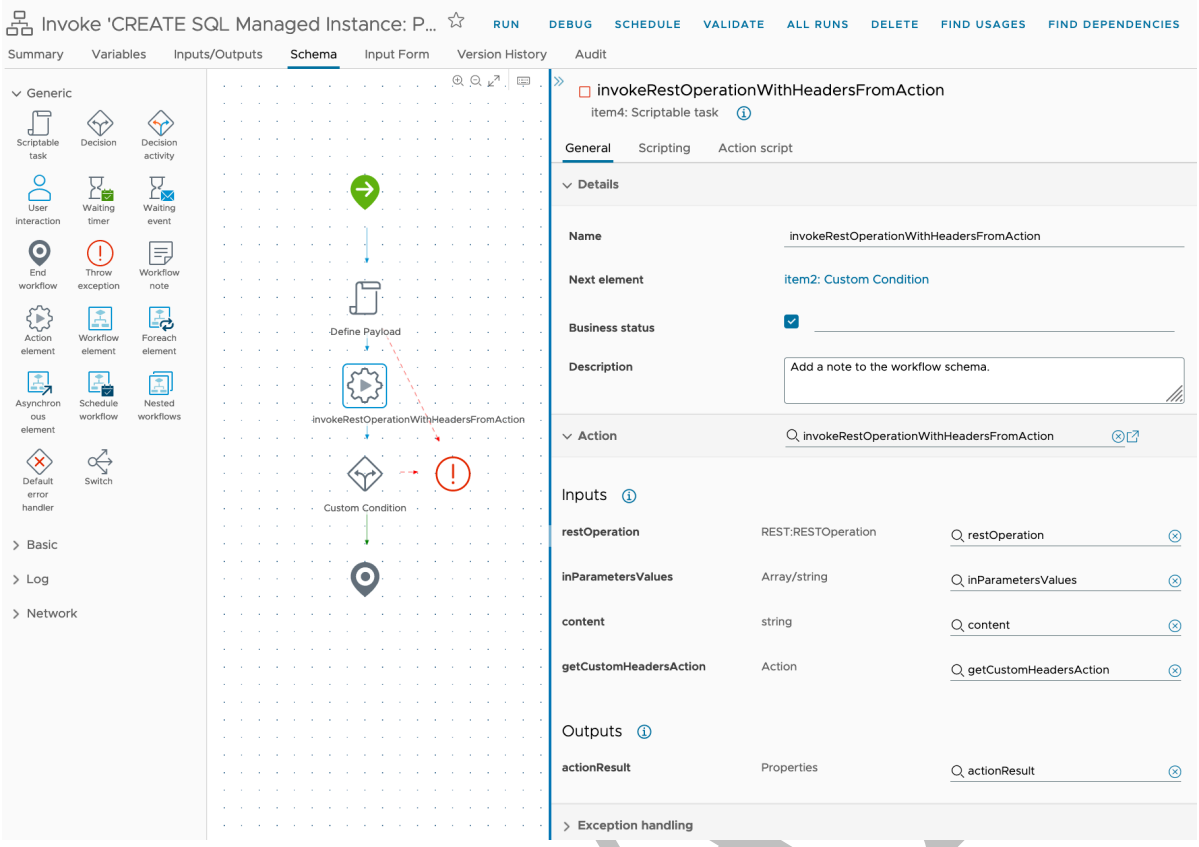



```
Template URL:
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGr
oupName}/providers/Microsoft.Sql/managedInstances/{managedInstanceName}?api-
version=2023-07-01
```

b) Run below workflow to generate a workflow from the REST Operation



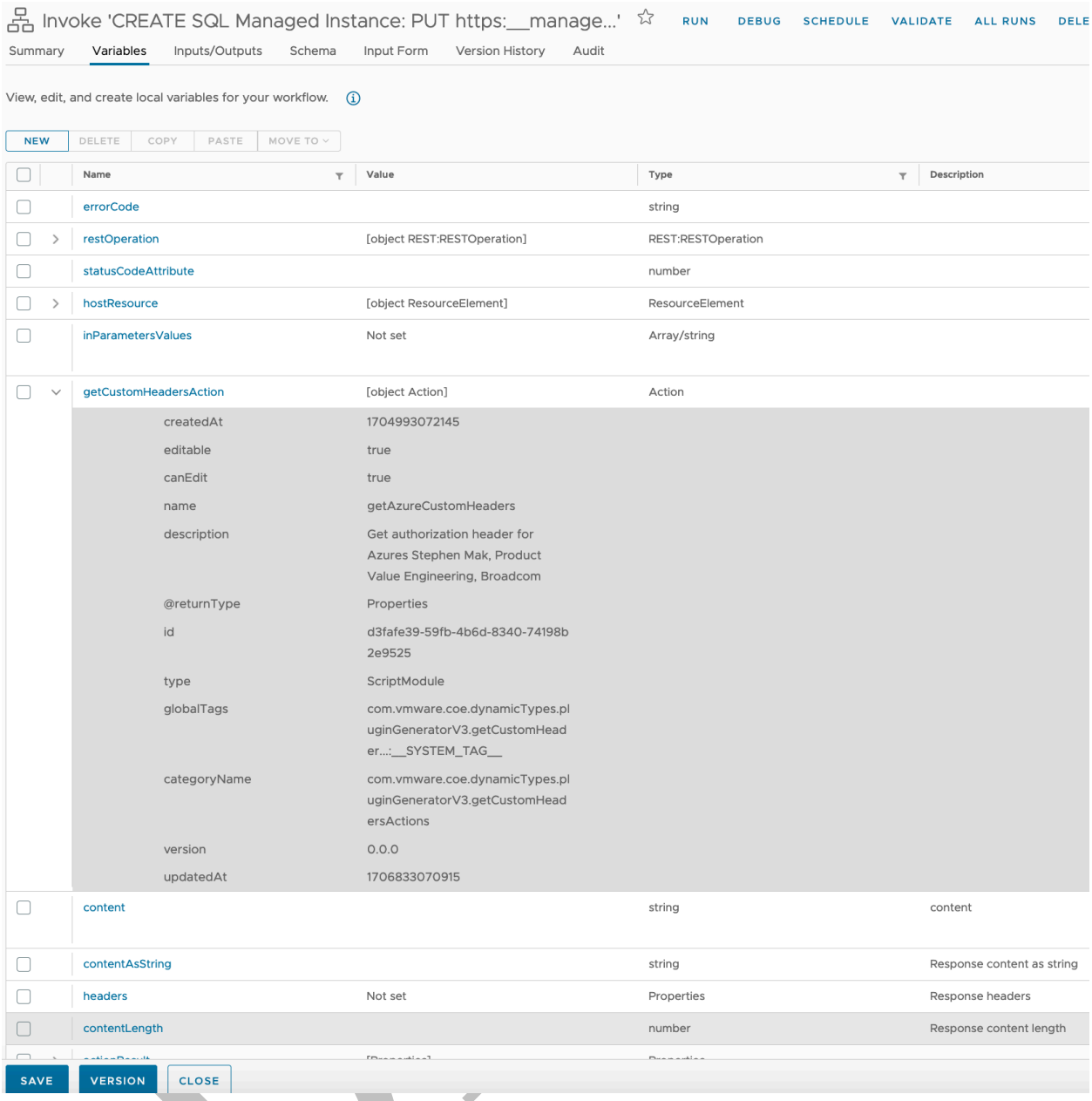
c) Add two new tasks, Define Payload and invokeRestOperationWithHeadersFromAction, to workflow



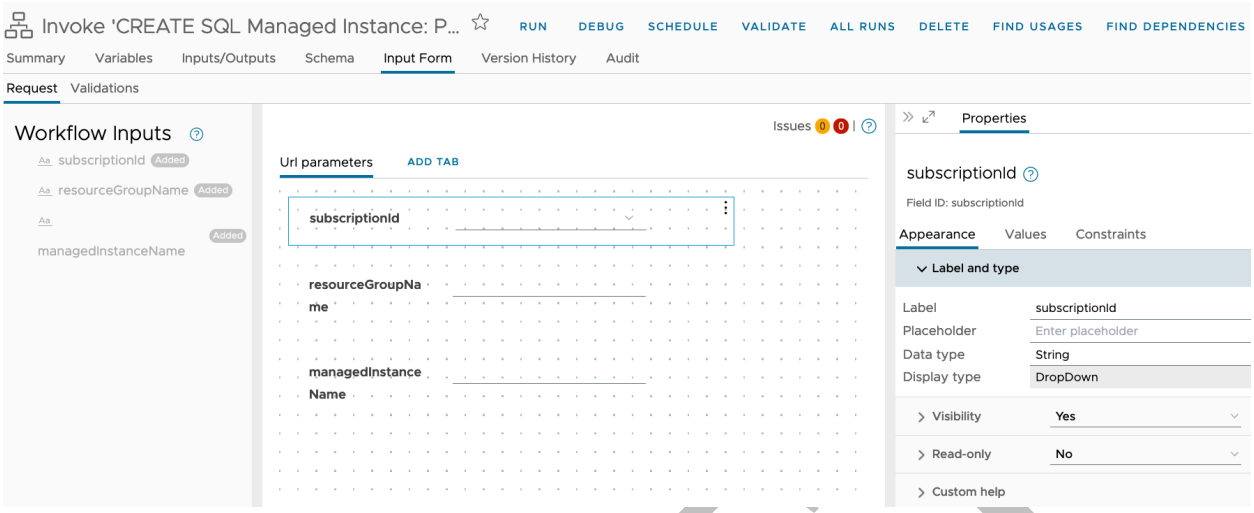
d) Edit “Define Payload” task. Paste in scripting the GET response of a resource instance in JSON format (e.g., virtual machine). Update keys and values accordingly. See an example in pre-defined workflow, 'Create VM: PUT https:__management.azure.com_sub.”

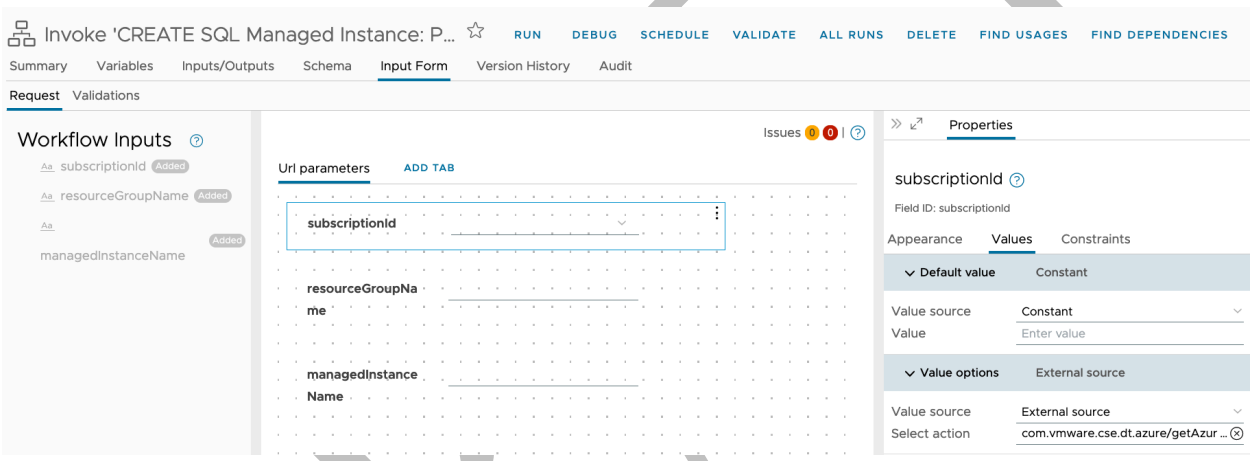
```
GET https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2023-09-01
```

e) Edit invokeRestOperationWithHeadersFromAction. Assign getAzureCustomHeadersAction to input getCustomHeadersAction.

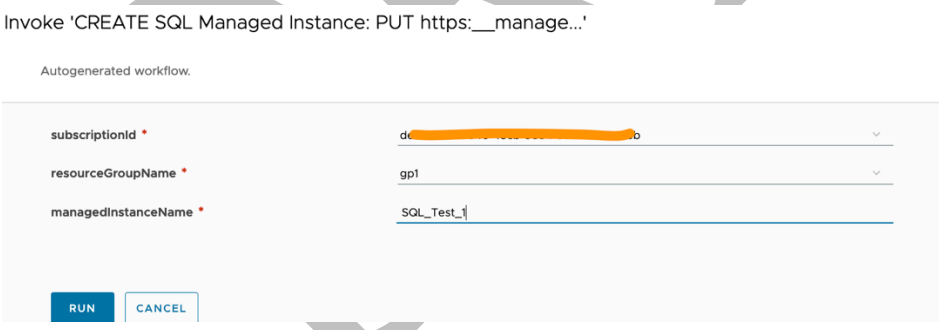


f) Apply input form actions (e.g., getAzureSubscriptionIds)





g) Test workflow



References

Authentication Failure

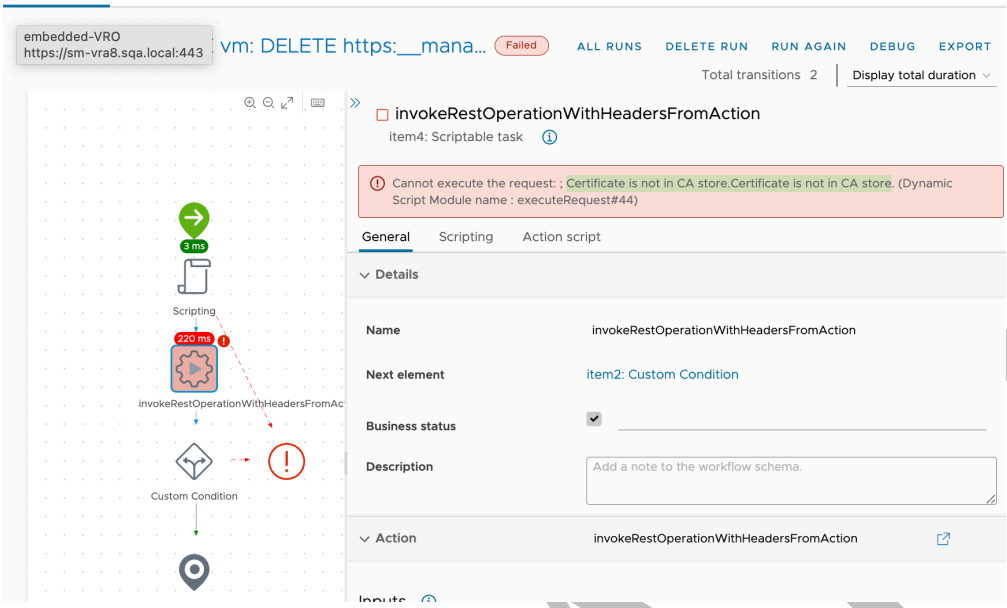
{"error":{"code":"AuthenticationFailed","message":"Authentication failed. The 'Authorization' header is missing."}}

Troubleshooting:

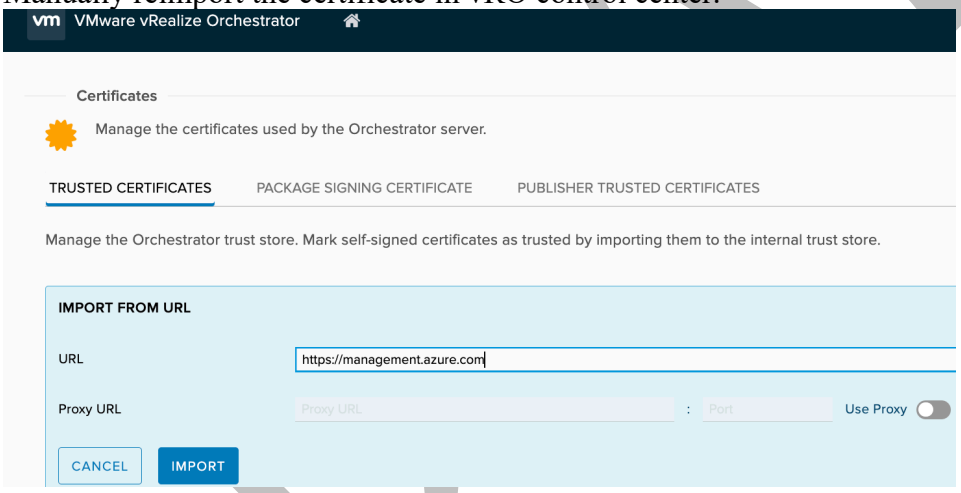
1) Test getAzureCustomHeaders action

Solution: run update rest operation workflow

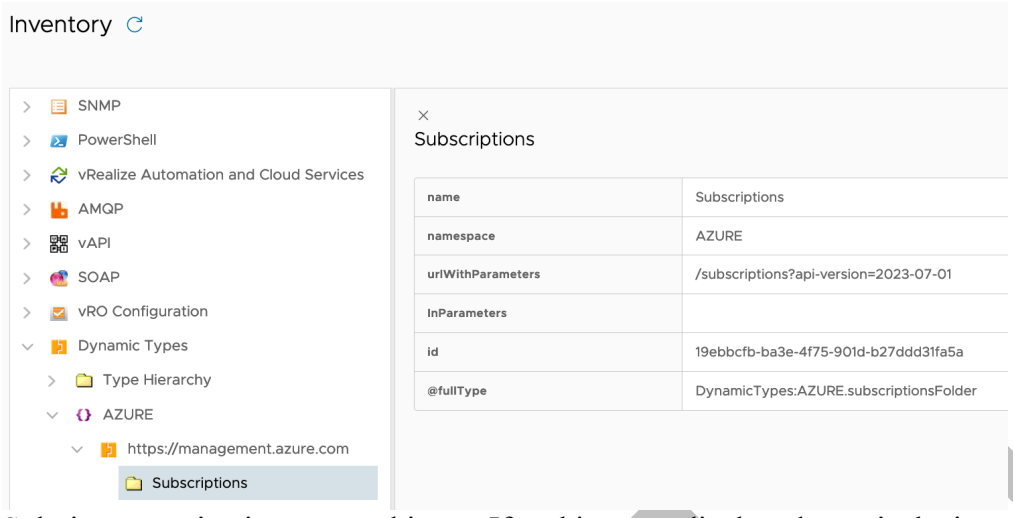
Cert Error



Solution: examine DT inventory objects. If nothing was displayed, below cert is the issue. Manually reimport the certificate in vRO control center.



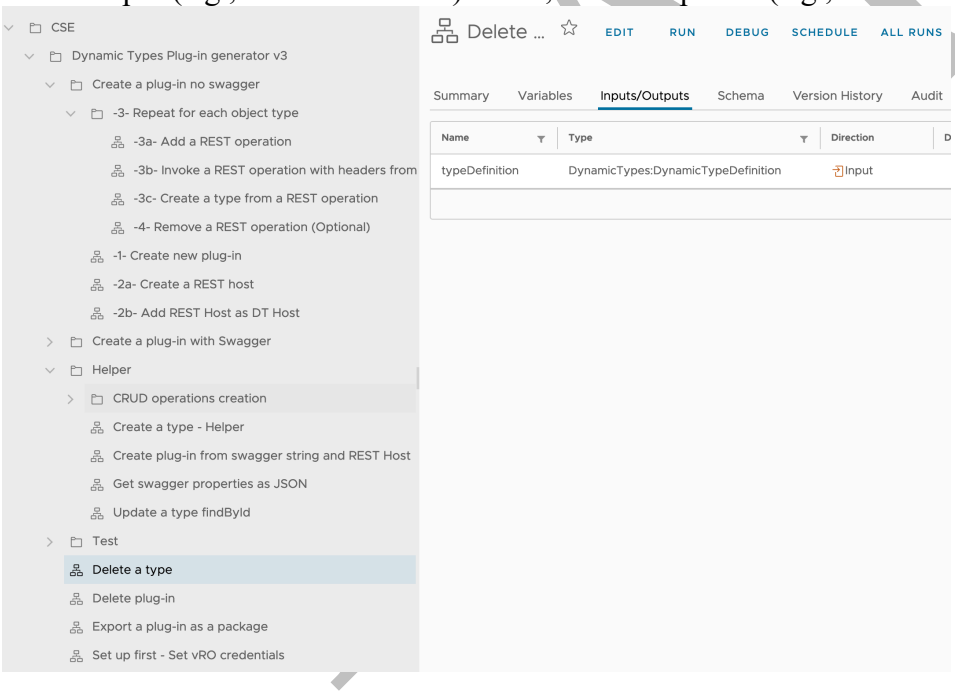
Inventory Issue



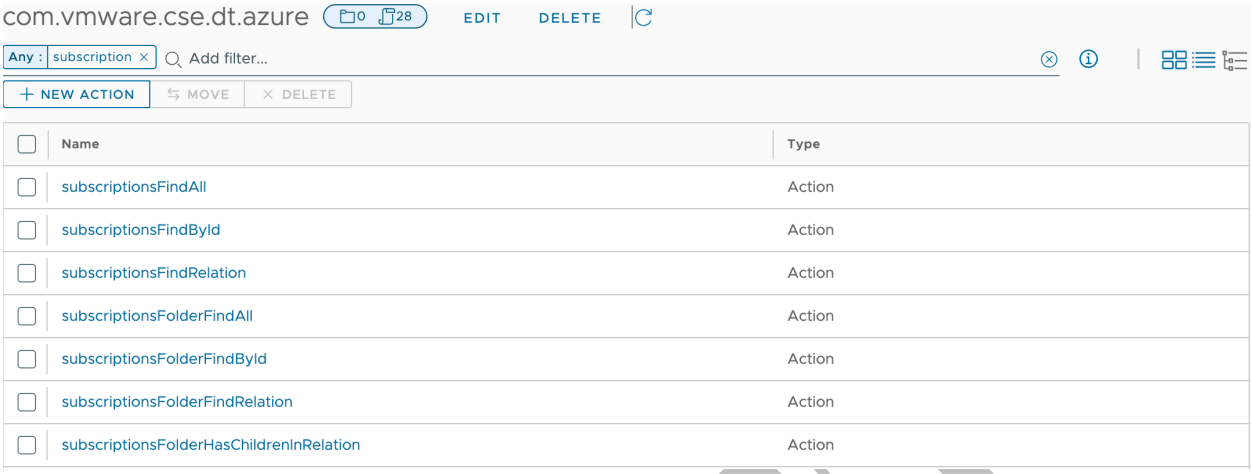
Solution: examine inventory objects. If nothing was displayed, cert is the issue. Manually reimport the management.azure.com certificate.

Delete a DT type

1) Run 'Delete a type' workflow two times. First, select a child type in Type Hierarchy as input (e.g., virtualMachines). Then, select its parent (e.g., virtualMachineFolder).

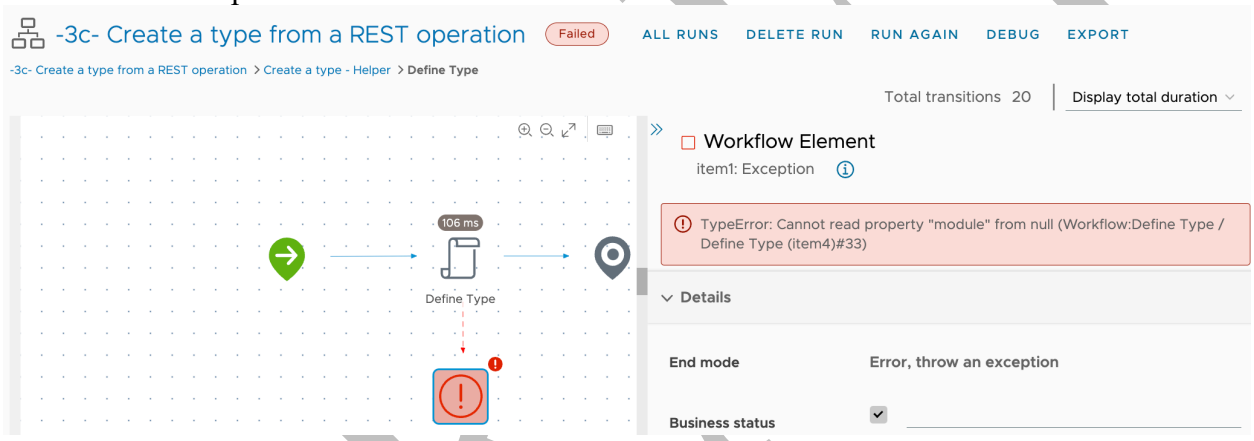


2) Delete all actions of both types from com.vwmare.cse.dt.azure module (e.g., subscriptions, subscriptionsFolder). See the list below.



Create a Type from REST Operation issue

Below is an example.



Solution: delete the type and its actions. Then recreate the type. If necessary, rollback and retry.

Rollback and Retry

If install didn't produce the expected result, it is necessary to delete the namespace and its dynamic types. To do so, reader deletes DT types in the reversed order and runs the 'remove namespace' workflow to delete namespace at the end. In addition, reader deletes all actions in com.vmware.cse.dt.azure module.

Azure Managed instances

Below are links about Azure Managed Instances. Each can be created via a REST call.
https://learn.microsoft.com/en-us/rest/api/sql/managed-instances?view=rest-sql-2021-11-01

https://learn.microsoft.com/en-us/rest/api/sql/managed-instances/update?view=rest-sql-2021-11-01&tabs=HTTP

https://learn.microsoft.com/en-us/azure/azure-sql/managed-instance/quickstart-content-reference-guide?view=azuresql

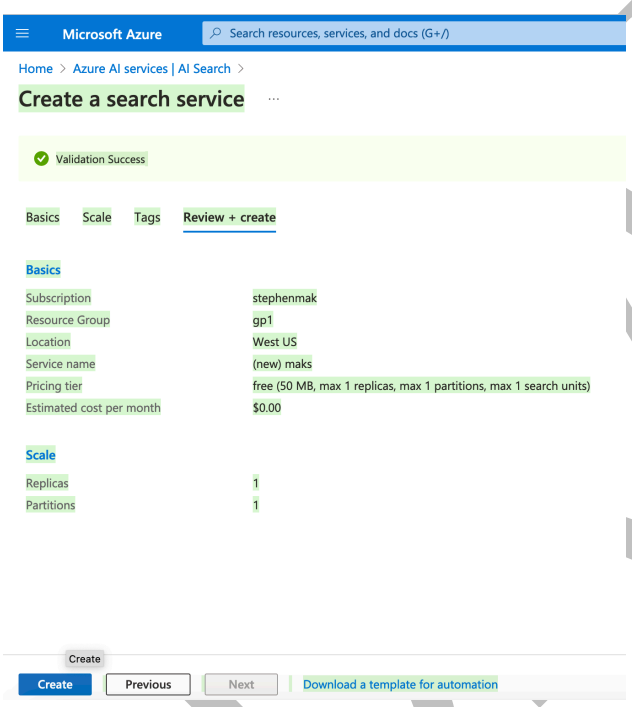
https://learn.microsoft.com/en-us/rest/api/sql/managed-instances/create-or-update?view=rest-sql-2021-11-01&tabs=HTTP

```
PUT
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Sql/managedInstances/{managedInstanceName}?api-version=2023-07-01
```

Azure REST Reference Page

https://learn.microsoft.com/en-us/rest/api/azure/

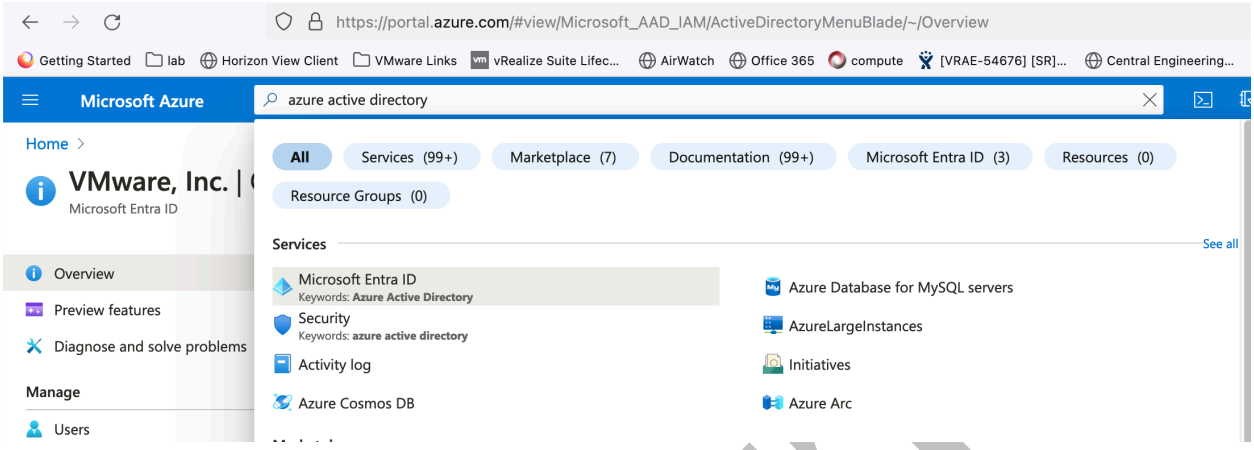
Azure search service Screenshot

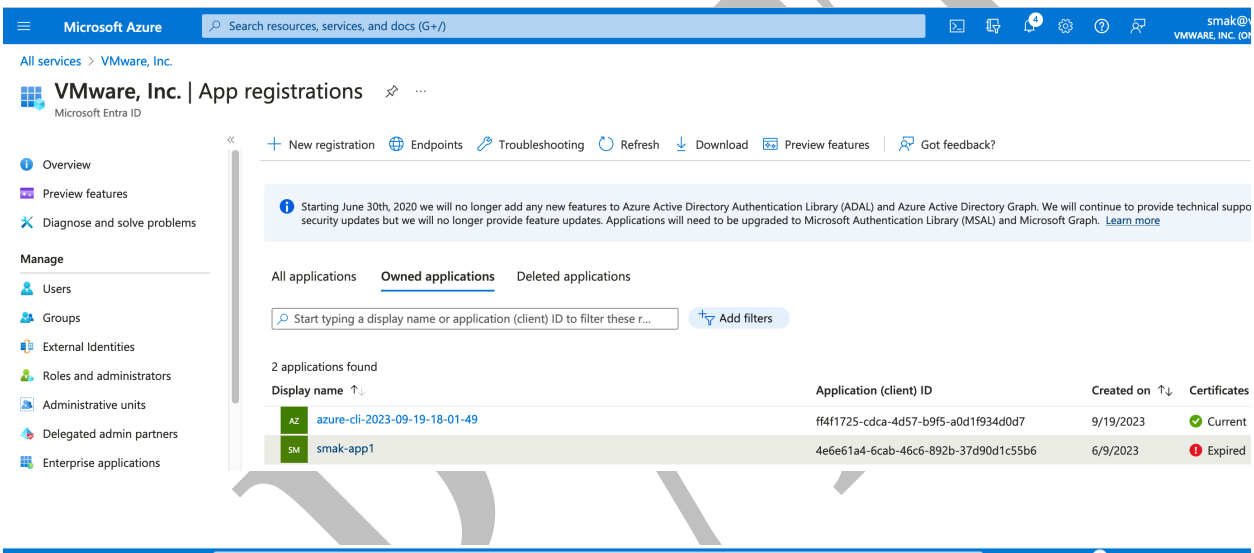


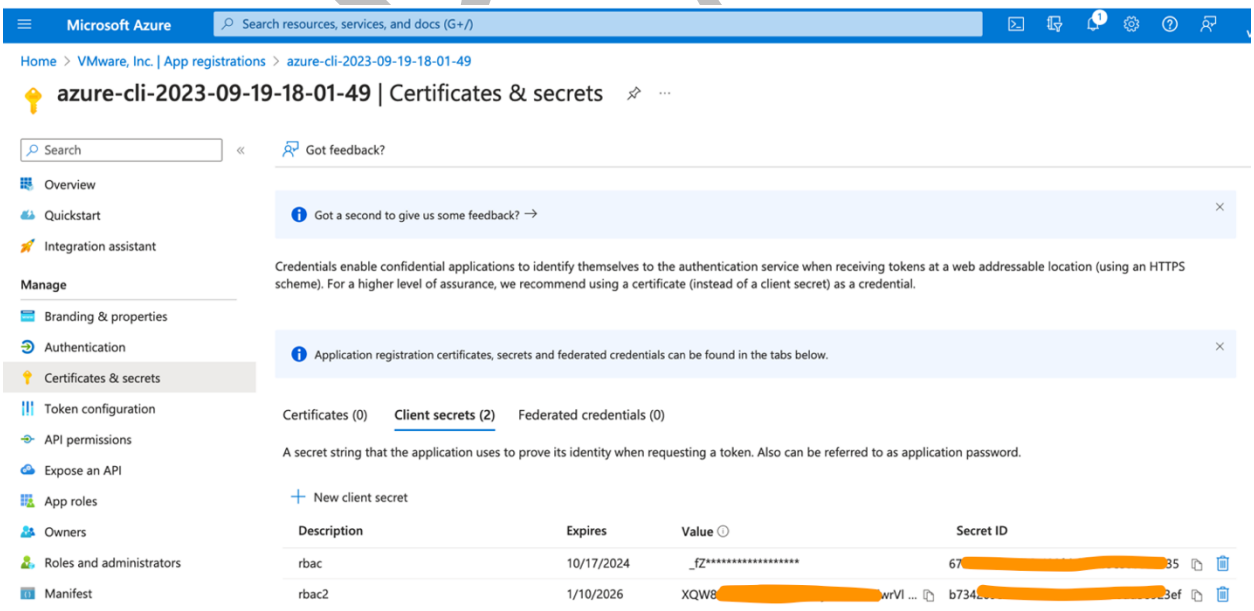
Create Application Secret

To access resources of a subscription, you create a secret for the application. If you don't have an application, see "Service principal" section to create one.

Below are screenshots to create a secret. https://learn.microsoft.com/en-us/rest/api/servicebus/get-azure-active-directory-token#use-postman-to-get-the-azure-ad-token is another example.


https://portal.azure.com/#view/Microsoft_AAD_IAM/ActiveDirectoryMenuBlade/~/Overview
Getting Started
lab
Horizon View Client
VMware Links
vRealize Suite Lifec...
AirWatch
Office 365
compute
[VRAE-54676] [SR]...
Central Engineering...
Microsoft Azure
azure active directory
Home
VMware, Inc. |
Microsoft Entra ID
Overview
Preview features
Diagnose and solve problems
Manage
Users
All
Services (99+)
Marketplace (7)
Documentation (99+)
Microsoft Entra ID (3)
Resources (0)
Resource Groups (0)
Services
See all
Microsoft Entra ID
Keywords: Azure Active Directory
Security
Keywords: azure active directory
Activity log
Azure Cosmos DB
Azure Database for MySQL servers
AzureLargeInstances
Initiatives
Azure Arc


Microsoft Azure
Search resources, services, and docs (G+/)
All services > VMware, Inc.
VMware, Inc. | App registrations
Microsoft Entra ID
Overview
Preview features
Diagnose and solve problems
Manage
Users
Groups
External Identities
Roles and administrators
Administrative units
Delegated admin partners
Enterprise applications
New registration
Endpoints
Troubleshooting
Refresh
Download
Preview features
Got feedback?
Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure Active Directory Graph. We will continue to provide technical suppo
security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. Learn more
All applications
Owned applications
Deleted applications
Start typing a display name or application (client) ID to filter these r...
Add filters
2 applications found
Display name
Application (client) ID
Created on
Certificates
azure-cli-2023-09-19-18-01-49
ff4f1725-cdca-4d57-b9f5-a0d1f934d0d7
9/19/2023
Current
smak-app1
4e6e61a4-6cab-46c6-892b-37d90d1c55b6
6/9/2023
Expired


Microsoft Azure
Search resources, services, and docs (G+/)
Home > VMware, Inc. | App registrations > azure-cli-2023-09-19-18-01-49
azure-cli-2023-09-19-18-01-49 | Certificates & secrets
Search
Overview
Quickstart
Integration assistant
Manage
Branding & properties
Authentication
Certificates & secrets
Token configuration
API permissions
Expose an API
App roles
Owners
Roles and administrators
Manifest
Got feedback?
Got a second to give us some feedback?
Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.
Application registration certificates, secrets and federated credentials can be found in the tabs below.
Certificates (0)
Client secrets (2)
Federated credentials (0)
A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.
New client secret
Description
Expires
Value
Secret ID
rbac
10/17/2024
rbac2
1/10/2026

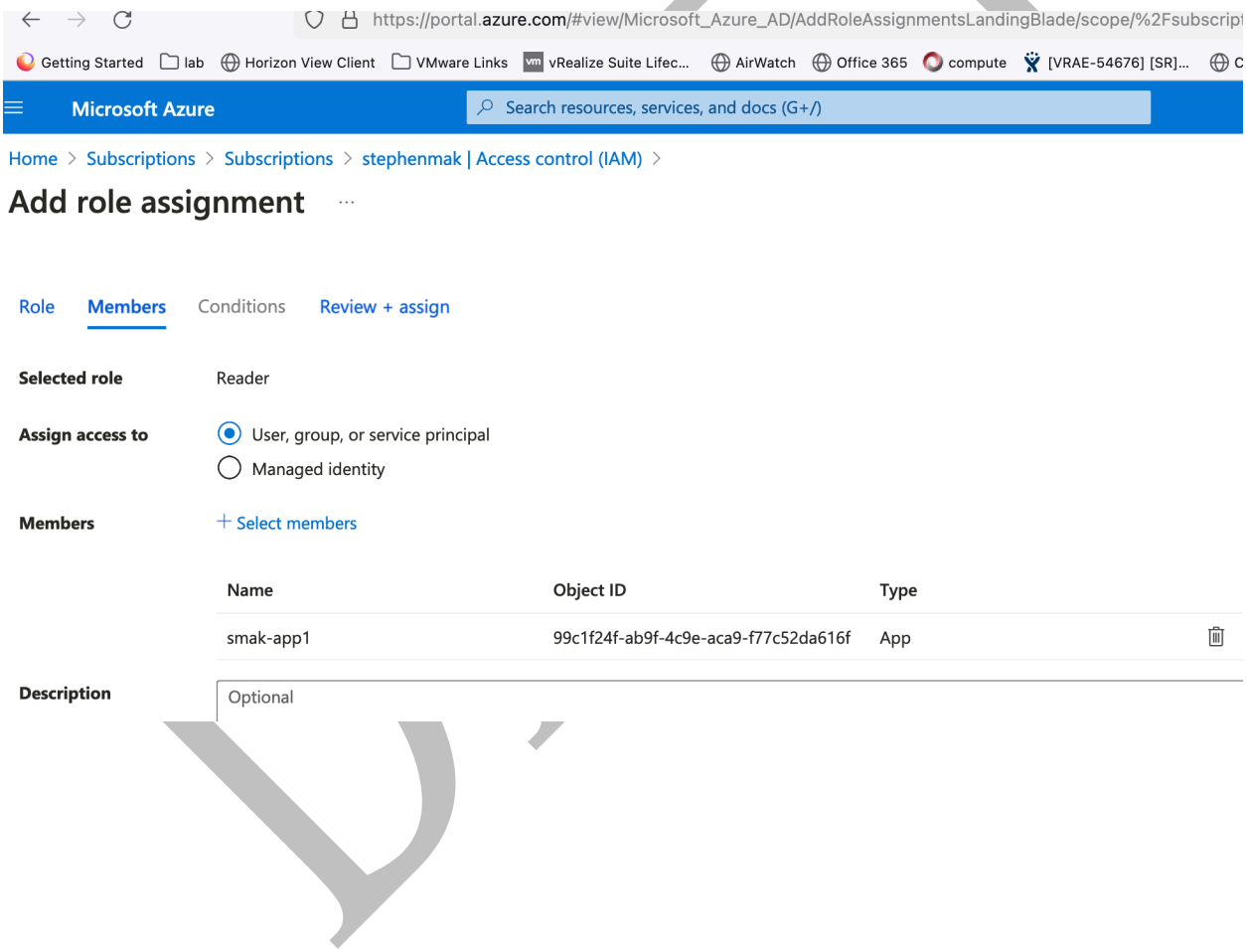
Service principal

Instead of having applications sign in as a fully privileged user, Azure offers service principals. An Azure service principal is an identity created for use with applications, hosted services, and automated tools. This identity is used to access resources.

See link for instructions to create a Microsoft Entra application and service principal with scope and role. https://learn.microsoft.com/en-us/entra/identity-platform/howto-create-service-principal-portal.

Azure offers four scope levels. A management group sees all subscriptions of the tenant. https://learn.microsoft.com/en-us/azure/role-based-access-control/scope-overview has the detail.

Below is an example of assigning a role to a service principal.



Dynamic Type Generator Acknowledgement

Dynamic Type Generator v3 was created by Chris Decanini of VMware/Broadcom. Version 3.1 has below updates over version 3 and has been regression tested with vRA8.14.1, vRA8.11.1 and cloud extensibility proxy.