Programming Guide

vRealize Automation 6.2

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see http://www.vmware.com/support/pubs.

EN-001636-00

vmware®

You can find the most up-to-date technical documentation on the VMware Web site at:

http://www.vmware.com/support/

The VMware Web site also provides the latest product updates.

If you have comments about this documentation, submit your feedback to:

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

vRealize Automation Programming Guide

The *Programming Guide* provides information about the vRealize Automation REST APIs, including how to use the REST API services and resources, create HTTP bearer tokens for authentication and authorization, and construct calls for the API Explorer.

Intended Audience

This information is intended for administrators and programmers who want to manage items in the vRealize Automation service catalog remotely.

VMware Technical Publications Glossary

VMware Technical Publications provides a glossary of terms that might be unfamiliar to you. For definitions of terms as they are used in VMware technical documentation, go to http://www.vmware.com/support/pubs.

Overview of the vRealize Automation REST API

1

The vRealize Automation REST API provides consumer, administrator, and provider-level access to the service catalog with the same services that support the console user interface. You can perform vRealize Automation functions programmatically using REST API service calls.

The REST API offers the following services and functions.

When a service request contains a resource URL, the first part of the URL identifies the service and the last part identifies the resource. For example, the following resource URL identifies the catalog service and the providers resource:

```
https://$host/component-registry/api/services
```

Table 1-1. REST API Services

Service	Description
Advanced Designer Service	Manage forms, endpoints, service blueprints, tenants, vRealize Orchestrator imports, workflows, and work items through the Advanced Designer Service.
Approval Service	Retrieve, create, update, and delete approval policies, policy types, policy instances, and policy requests.
Branding Service	Change the background and text colors, company logo, company name, product name, tenant name, and other resources in the console.
Catalog Service	Retrieve global and entitled catalog items, and entitlements for a catalog item and its service that the current user can review. A consumer can retrieve, edit, and submit a request form for a catalog item. A provider can retrieve, register, update, and delete catalog items. Provision and manage systems.
Catalog Registry	Access services from a single location.
EventLog Service	Query system events recorded by other services.
Files Service	Unused.
Identity Service	Manage tenants, business groups, SSO and custom groups, users, and identity stores.
Licensing Service	Retrieve permissions and post serial keys.
Management Service	Retrieve work item forms, callbacks, and tasks. Manage endpoint details including tenant, password, user name, and endpoint URL. Retrieve performance metrics. Retrieve and cancel reclamation requests.
Notification Service	Configure and send notifications for several types of events such as the successful completion of a catalog request or a required approval.

Table 1-1. REST API Services (Continued)

Service	Description
Plug-in Service	Retrieve, create, update, and delete a resource. Retrieve an extension. Retrieve license notifications.
Portal Service	Retrieve, create, update, and delete a portal resource.
Reservation Service	Retrieve, create, update, and delete a reservation or reservation policy.
vCO Service	Manage vRealize Orchestrator actions, tasks, packages, and workflows. Browse system and plug-in inventories.
WorkItem Service	Retrieve, create, update, complete, cancel, and delete a work item. Also retrieve form data, metadata, detail forms, and submission forms from service providers.

For a more terse description of the API services, see the *REST API Reference*, which is a collection of zipped resource files located on the VMware vRealize Automation Documentation page at https://www.vmware.com/support/pubs/vcac-pubs.html. For more information, see "Using the REST API Services Reference Documentation," on page 191.

REST API Authentication 2

In the REST API, vRealize Automation requires HTTP bearer tokens in request headers for authentication of consumer requests. A consumer request applies to tasks that you can perform in the vRealize Automation console, such as requesting a machine.

To acquire an HTTP bearer token, you authenticate with an identity service that manages the communication with the SSO server. The identity service returns an HTTP bearer token that you include in all request headers until the token expires, or you delete it. An HTTP bearer token expires in 24 hours by default, but you can configure the token with a different duration.

Using HTTP Bearer Tokens

You use HTTP bearer tokens for tasks that you can also perform in the vRealize Automation console. Whether you create a request header with the `curl` command or with some other utility,

You use POST, HEAD, and DELETE methods to manage HTTP bearer tokens.

Method	URL	Description
POST	/tokens	Authenticate the user with the identity service /tokens and generate a new token.
HEAD	/tokens/*tokenID*	Validate the token *tokenID*.
DELETE	/tokens/*tokenID*	Delete the token *tokenID*.

The root URL for HTTP bearer calls is `https://$vra_server/identity/api/tokens`.

Configure the Duration of an HTTP Bearer Token

You set the duration of HTTP bearer tokens in the `/etc/vcac/security.properties` file on the vRealize Automation appliance.

The effective duration or lifetime of an HTTP bearer token depends on the duration of its corresponding SAML token, which the SSO server creates at request time. An HTTP bearer token expires when it reaches the end of its configured duration, or at the end of the configured duration of the SAML token, whichever comes first. For example, if the configured duration is three days for the HTTP bearer token and two days for the SAML token, the HTTP bearer token expires in two days. A configuration setting on the SSO server determines the duration of SAML tokens.

Prerequisites

- Log in to the vRealize Automation appliance with SSH as root. The password is the one you specified when you deployed the appliance.
- The `/etc/vcac/security.properties` file on the appliance must be editable.

Procedure

1 Open the `/etc/vcac/security.properties` file for editing.

2 Add the following lines to the file, where *N* is an integer specifying the duration of the token in hours.

```
identity.basic.token.lifetime.hours=N
#The number is in hours.
```

3 Save and close the file.

4 Log out of the vRealize Automation appliance.

The new value applies the next time someone requests an HTTP bearer token.

Request an HTTP Bearer Token

A bearer token is required by the REST client to use the vRealize Automation REST API. You can obtain a bearer token by authenticating to the identity service.

Prerequisites

- Log in to vRealize Automation using the applicable credentials. For example, to assign a user to a role, log in as a **tenant administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.

For related information, see "Example: Requesting an HTTP Bearer Token," on page 13.

Input

You can use supported input parameters to control the command output.

A consumer request must specify the correct component registry service and resource. For example, the URL to obtain an HTTP bearer token must contain the identity service and token resource values.

Input	Description
host	*host name.domain name* of the vRealize Automation server, for example, **`mycompany.mktg.mydomain.com`**.
usrname	Specifies the tenant administrator user name.
passwd	Specifies the tenant administrator password.
tenantURLtoken	Specifies the tenant URL token determined by the system administrator when creating the tenant, for example, **`support`**.

Output

The following information is displayed as a result of your HTTP bearer token request.

Output	Description
expires	Contains the ISO 8601 timestamp indicating when the token expires.
id	Contains the HTTP bearer token to use in Authorization header in subsequent requests.
tenant	Displays the tenant ID associated with the token.

Response Status Codes

The following codes may be displayed as a result of your HTTP bearer token request.

Status Code	Description
200 OK	Your request succeeded and the resource was updated. The response body contains the full representation of the resource.
400 BAD REQUEST	The data you provided in the POST failed validation. Inspect the response body for details.
401 UNAUTHORIZED	The request could not authenticate the user or authentication credentials required.

Example: `curl` Command

You can use the following command line format to request an HTTP bearer token.

```
curl --insecure -H "Accept: application/json" -H 'Content-Type: application/json' --data
'{"username":"usrname",
"password":"passwd","tenant":"tenantURLtoken"}' https://$host/identity/api/tokens
```

After receiving the bearer token, you can include it in your request headers.

The HTTP bearer token expires in 24 hours by default. See "Configure the Duration of an HTTP Bearer Token," on page 9 for information on how to set the duration.

To delete a token, see "Delete an HTTP Bearer Token," on page 14.

Example: POST Request

You can use the POST method with your user name, password, and tenant ID to request an HTTP bearer token.

```
POST /tokens
Accept: application/json
Content-Type: application/json
{
"username":"tanteater@example.com",
"password":"password",
"tenant":"MYCOMPANY"
}
```

Example: HTTP Bearer Request Response

When your request succeeds, the system returns the `200 OK` status code, the expiration date and time of the token, and the HTTP bearer token.

```
HTTP/1.1 200 OK
  Server: Apache-Coyote/1.1
  Cache-Control: no-cache, no-store
  Pragma: no-cache
  Expires: Wed, 19 Nov 2014 23:59:59 GMT
  Content-Type: application/json;charset=UTF-8
  Content-Length: 324
  Date: Sat, 01 February 2015 13:04:50 GMT
{
   "expires":"2015-02-01T13:09:45.619Z",
   "id":"MTM5MTI1OTg5MDQwMzozNDQyZWMxZmQ5ZDliODUzMGFiMjp0ZW5hbnQ6cWV1c2VybmFtZTjYjY1ZjhiODI
   2OTg4ODU3M2UwOTUwOWRkMjlmYWRjNWQ4NjJkOTk1YmE3MTg1MWZhOTc2MjEyYjYxZmU3YTVhZDcwNzM3ZTg3ZDN
   jNDk2ZDlmNA==","tenant":"MYCOMPANY"
}
```

Procedure: Request an HTTP Bearer Token

You use an HTTP bearer token to authenticate a consumer request.

A consumer request must specify the correct component registry service and resource. For example, the URL to obtain an HTTP bearer token must specify the identity service and token resource.

The HTTP bearer token expires in 24 hours by default. See "Configure the Duration of an HTTP Bearer Token," on page 9 for information on how to set the duration.

See "Delete an HTTP Bearer Token," on page 14 to delete a token.

The variable *$vRA* used here represents the *host name.domain name* of the vRealize Automation server, for example, **`mycompany.mktg.mydomain.com`**.

Prerequisites

You must have the host name and fully qualified domain name of the vRealize Automation instance, represented here by the *$VCAC* variable.

Procedure

- Enter a `curl` command in the following format, replacing the variables with the correct values.

```
curl –-insecure –H "Accept: application/json" –H 'Content–Type: application/json'
–-data '{"username":"usrname","password":"passwd","tenant":"tenantURLtoken"}'
https://$vRA/identity/api/tokens
```

Variable	Description
usrname	Tenant administrator user name
passwd	Tenant administrator password
tenantURLtoken	Tenant URL token specified by the system administrator when creating the tenant, for example, **`support`**.

The system returns a response header with a status code and, if your request is successful, an HTTP bearer token.

Status Code	Description
200 OK	Your request succeeded and the resource was updated. The response body contains the full representation of the resource.
400 BAD REQUEST	The data you provided in the POST failed validation. Inspect the response body for details.
401 UNAUTHORIZED	Could not authenticate the user, or authentication credentials required.

Example: Sample Request and Response Headers

```
POST /tokens
Accept: application/json
Content–Type: application/json
{
```

```
"username":"tanteater@example.com",
"password":"password",
"tenant":"MYCOMPANY"
}
```

```
HTTP/1.1 200 OK
  Server: Apache-Coyote/1.1
  Cache-Control: no-cache, no-store
  Pragma: no-cache
  Expires: Sun, 02 Nov 2014 23:59:59 GMT
  Content-Type: application/json;charset=UTF-8
  Content-Length: 324
  Date: Wed, 01 Oct 2014 13:04:50 GMT
{
   "expires":"2014-02-01T13:09:45.619Z",
   "id":"MTM5MTI1OTg5MDQwMzozNDQyZWMxZmQ5ZDliODUzMGFiMjp0ZW5hbnQ6cWV1c2VybmFtZTjYjY1ZjhiODI2OTg
   4ODU3M2UwOTUwOWRkMjlmYWRjNWQ4NjJkOTk1YmE3MTg1MWZhOTc2MjEyYjYxZmU3YTVhZDcwNzM3ZTg3ZDNjNDk2ZDl
   mNA==","tenant":"MYCOMPANY"
}
```

What to do next

Include the HTTP bearer token in all of your consumer requests. You can store the token in a variable such as $AUTH and then use the variable in your requests.

Example: Requesting an HTTP Bearer Token

You can request an HTTP bearer token for a specific tenant when you are logged in as a tenant administrator.

Request the HTTP Bearer Token

You can use the following command to request an HTTP bearer token. For related information, see "Request an HTTP Bearer Token," on page 10.

```
curl --insecure -H "Accept: application/json" -H 'Content-Type: application/json' --data
'{"username":"tanteater @example.com","password":"password","tenant":"MYCOMPANY"}'
https://tanteater.eng.mycompany.com:4870/identity/api/tokens
```

Response to HTTP Bearer Token Request

The following sample represents the result of successfully submitting the HTTP bearer request. For related information about result codes and bearer token longevity, see "Request an HTTP Bearer Token," on page 10.

```
HTTP/1.1 200 OK
  Server: Apache-Coyote/1.1
  Cache-Control: no-cache, no-store
  Pragma: no-cache
  Expires: Wed, 19 Nov 2014 23:59:59 GMT
  Content-Type: application/json;charset=UTF-8
  Content-Length: 324
  Date: Sat, 01 Feb 2015 13:04:50 GMT
{
 "expires":"2015-02-01T13:09:45.619Z",
 "id":"MTM5MTI1OTg5MDQwMzozNDQyZWMxZmQ5ZDliODUzMGFiMjp0ZW5hbnQ6cWV1c2VybmFtZTpmcml0ekBjb2tlLmNvb
```

```
 TplMDViNGU0NGM2ZWU0MWQ1OWEwMTNmZGExNTQwZjNlNGM3YTBlM2I5MDhlYWZjYjY1ZjhiODI2OTg4ODU3M2UwOTUwOWRk
 MjlmYWRjNWQ4NjJkOTk1YmE3MTg1MWZhOTc2MjEyYjYxZmU3YTVhZDcwNzM3ZTg3ZDNjNDk2ZDlmNA==",
 "tenant":"MYCOMPANY"
}
```

For an example of when and how to use an HTTP bearer token, see "List All Identity Stores for the Tenant," on page 22.

Validate an HTTP Bearer Token

You can validate an existing token with a HEAD request.

Prerequisites

- Log in as a tenant administrator in the vRealize Automation console.
- You must have an HTTP bearer token.

Procedure

1 Create the HEAD request with your HTTP bearer token.

2 Submit the request.

The system returns a status code.

Status Code	Description
204 NO CONTENT	The request succeeded.
401 UNAUTHORIZED	You must have authentication credentials to access the resource. All requests must be authenticated.
403 FORBIDDEN	Your authentication credentials do not provide sufficient access to the resource.
404 NOT FOUND	Could not locate the resource based on the specified URI.
405 METHOD NOT ALLOWED	The HEAD method is not supported for the resource.
500 SERVER ERROR	Could not create or update the resource because of an internal server error.

Example: Sample Request and Response

```
HEAD
/tokens/MTM5MTI1OTg5MDQwMzozNDQyZWMxZmQ5ZDliODUzMGFiMjp0ZW5hbnQ6cWV1c2VybmFtZTjYjY1ZjhiODI2OTg4O
DU3M2UwOTUwOWRkMjlmYWRjNWQ4NjJkOTk1YmE3MTg1MWZhOTc2MjEyYjYxZmU3YTVhZDcwNzM3ZTg3ZDNjNDk2ZDlmNA==
Accept: application/json
```

```
HTTP/1.1 204 No Content
```

Delete an HTTP Bearer Token

You can delete a token with a DELETE request.

Prerequisites

- Log in as a tenant administrator in the vRealize Automation console.
- You must have an HTTP bearer token.

Procedure

1 Create the DELETE request with your HTTP bearer token.

2 Submit the request.

The system returns a status code.

Status Code	Description
204 NO CONTENT	The request succeeded. The resource has been deleted.
401 UNAUTHORIZED	You must have authentication credentials to access the resource. All requests must be authenticated.
403 FORBIDDEN	Your authentication credentials do not provide sufficient access to the resource.
404 NOT FOUND	Could not locate the resource based on the specified URI.
405 METHOD NOT ALLOWED	The DELETE method is not supported for the resource.
500 SERVER ERROR	Could not create or update the resource because of an internal server error.

Example: Sample Request and Response

```
DELETE
/tokens/MTM5MTI1OTg5MDQwMzozNDQyZWMxZmQ5ZDliODUzMGFiMjp0ZW5hbnQ6cWV1c2VybmFtZTjYjY1ZjhiODI2OTg4O
DU3M2UwOTUwOWRkMjlmYWRjNWQ4NjJkOTk1YmE3MTg1MWZhOTc2MjEyYjYxZmU3YTVhZDcwNzM3ZTg3ZDNjNDk2ZDlmNA==
Accept: application/json

HTTP/1.1 204 No Content
```

REST API Use Cases 3

Available use cases provide the prerequisite, command line options and format, and sample results to help you perform a variety of vRealize Automation functions, such as requesting a machine or creating a reservation.

This chapter includes the following topics:

Creating a Tenant

You can create a vRealize Automation tenant using the REST API.

The checklist provides the tasks required to create a tenant with the REST API. Perform the tasks in sequence.

Table 3-1. Creating a Tenant Checklist

Task	Details	Permissions
❑ Request an HTTP bearer token	See "Request an HTTP Bearer Token," on page 10.	system administrator
❑ Display your current tenants	See "Display Your Current Tenants," on page 18.	system administrator
❑ Request a new tenant	See "Request a New Tenant," on page 20.	system administrator
❑ List all identity stores for the tenant	See "List All Identity Stores for the Tenant," on page 22.	system administrator
❑ Link an identity store to the tenant	See "Link an Identity Store to the Tenant," on page 25.	system administrator and tenant administrator
❑ Search LDAP or Active Directory for a user	See "Search LDAP or Active Directory for a User," on page 28.	system administrator

Table 3-1. Creating a Tenant Checklist (Continued)

Task	Details	Permissions
❑ Assign a user to the tenant administrator role CSP_TENANT_ADMIN	See "Assign a User to a Role," on page 30.	system administrator
❑ Show all roles assigned to a user	See "Display all Roles Assigned to a User," on page 31.	system administrator
❑ Assign a user to the IaaS administrator role COM_VMWARE_IAAS_IAAS_ADMINISTRATOR	See "Assign a User to a Role," on page 30.	system administrator

Display Your Current Tenants

You can display the current tenants in the vRealize Automation system.

Prerequisites

- Log in to the vRealize Automation server as a **system administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/identity/api/tenants
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Species an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL which produces the result.
Content	Specifies an array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following information:
Id	Specifies the unique tenant identifier.
urlName	Specifies the name of the tenant as it appears in URLs.
Name	Specifies the name of the tenant for display purposes.
description	Specifies the long description of the tenant.

Property	Description
contactEmail	Specifies the primary contact email address.
Password	Unused.
defaultTenant	**True** if the corresponding tenant is the default tenant (vsphere.local).
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: curl Command

You can use the following command to display your current tenants.

```
curl –-insecure –H "Accept:text/xml" –H "Authorization: Bearer $token"
https://$host/identity/api/tenants
```

You can format XML output to improve its readability. See “Formatting Your JSON Output,” on page 190 for more information about formatting output.

Example: API Explorer

You can use the following command to display your current tenants.

```
vcac–cli>rest get –-service authentication –-uri tenants
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
{
"links" : [ ],
"content" : [ {
"@type" : "Tenant",
"id" : "vsphere.local",
"urlName" : "vsphere.local",
"name" : "vsphere.local",
"description" : null,
"contactEmail" : null,
"password" : null,
"defaultTenant" : true
}, {
"@type" : "Tenant",
"id" : "MYCOMPANY",
"urlName" : "MYCOMPANY",
"name" : "QETenant",
"description" : "Test tenant",
"contactEmail" : null,
"password" : "defaultPwd#1",
"defaultTenant" : false
```

```
} ],
"metadata" : {
"size" : 19,
"totalElements" : 2,
"totalPages" : 1,
"number" : 1,
"offset" : 0
}
}
```

Request a New Tenant

You can submit a request for a tenant with the identity service and a JSON input file.

Prerequisites

- Log in to the vRealize Automation server as a **system administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/identity/api/tenants/*$tenantId* --data @ *$inputFileName*.json
$token	Specifies a valid HTTP bearer token with necessary credentials.
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$tenantId	Specifies the ID of the tenant.
$tenantURL	Specifies the URL of the tenant.
$enantName	Specifies the name of the tenant.
$description	Specifies a description of the tenant.
$emailAddress	Specifies the contact email address for the tenant.

JSON Input File Template

Use the following template to create a JSON input file. Replace the italicized variables in the template with actual values in the file.

```
{
    "@type" : "Tenant",
    "id" : "$tenantId",
    "urlName" : "$tenantURL",
    "name" : "$tenantName",
    "description" : "$description",
    "contactEmail" : "$emailAddress",
    "defaultTenant" : false
}
```

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Species an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL which produces the result.
Content	Specifies an array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following information:
Id	Specifies the unique tenant identifier.
urlName	Specifies the name of the tenant as it appears in URLs.
Name	Specifies the name of the tenant for display purposes.
description	Specifies the long description of the tenant.
contactEmail	Specifies the primary contact email address.
Password	Unused.
defaultTenant	**`True`** if the corresponding tenant is the default tenant (vsphere.local).
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: curl Command

You can submit a request for a new tenant and either call a JSON file that contains tenant request parameters or specify those parameters using inline text. The first example uses a JSON file as input. The second example uses inline text as input.

The first example calls the following sample `newTenant.json` file.

```
{
    "@type" : "Tenant",
    "id" : "development",
    "urlName" : "development",
    "name" : "DevelopmentTenant",
    "description" : "Tenant for all developers",
    "contactEmail" : "admin@mycompany.com",
    "defaultTenant" : false
}
```

You can use the following example to call the above JSON text file, which contains parameters for the tenant request.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/identity/api/tenants/development --data @C:\Temp\newTenant.json
```

You can use the following example to specify parameters for the tenant request by using inline text.

```
curl --insecure -H "Accept: application/json" -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
--data '{"@type":"Tenant","id":"development","urlName":"development","name":
"DevelopmentTenant","description":"Tenant for all developers","contactEmail":
"admin@mycompany.com","defaultTenant":false}'
```

Examples: API Explorer

You can submit a request for a new tenant and either call a JSON file that contains tenant request parameters or specify those parameters using inline text. The first example uses a JSON file as input. The second example uses inline text as input.

The first example calls the following sample `newTenant.json` file.

```
{
    "@type" : "Tenant",
    "id" : "development",
    "urlName" : "development",
    "name" : "DevelopmentTenant",
    "description" : "Tenant for all developers",
    "contactEmail" : "admin@company.com",
    "defaultTenant" : false
}
```

You can use the following example to call the above JSON file, which contains parameters for the tenant request.

```
vcac-cli>rest post --headers --service catalog-service --uri consumer/requests
--data @newTenant.json
```

You can use the following example to specify parameters for the tenant request by using inline text.

```
vcac-cli>rest post --headers --service catalog-service --uri consumer/requests
--data '{"@type":"Tenant","id":"development","urlName":"development","name":
"DevelopmentTenant","description":"Tenant for all developers","contactEmail":
"admin@company.com","defaultTenant":false}'
```

List All Identity Stores for the Tenant

You can list all available identity stores for a named tenant, such as the default tenant vsphere.local.

Prerequisites

- Log in to vRealize Automation as a **system administrator** and a **tenant administrator**.
- Verify that you have access to a functional LDAP, Active Directory, or Native Active Directory identity server.
- Verify that you have the identity server details required for the JSON template.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.

- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/identity/api/tenants/*$tenantId*/directories
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$tenantId	Specifies the ID of the tenant.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Species an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL which produces the result.
Content	Specifies an array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following information:
Id	Specifies the unique tenant identifier.
urlName	Specifies the name of the tenant as it appears in URLs.
Name	Specifies the name of the tenant for display purposes.
description	Specifies the long description of the tenant.
contactEmail	Specifies the primary contact email address.
Password	Unused.
defaultTenant	**`True`** if the corresponding tenant is the default tenant (vsphere.local).
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: `curl` Command

You can use the following command to list identity stores by using variables, instead of the full token and host name.domain name.

```
curl –-insecure –H "Accept: application/json" –H 'Content-Type: application/json'
–H "Authorization: Bearer $token" https://$host/identity/api/tenants/MYCOMPANY/directories
```

Example: `wget` Command

You can use the following command to list identity stores by using variables, instead of the full token and host name.domain name.

```
wget –-no-check-certificate –S –q –-header "Accept: application/json"
–-header='Content-Type: application/json'
–-header="Authorization: Bearer MTM5MTI1OTg5MDQwMzozNDQyZWMxZmQ5ZDliODUzMGFiMjp0ZW5hbnQ6cWV1c2V
ybmFtZTpmcml0ekBjb2tlLmNvbTplMDViNGU0NGM2ZWU0MWQ1OWEwMTNmZGExNTQwZjNlNGM3YTBlM2I5MDhlYWZjYjY1Zj
hiODI2OTg4ODU3M2UwOTUwOWRkMjlmYWRjNWQ4NjJkOTk1YmE3MTg1MWZhOTc2MjEyYjYxZmU3YTVhZDcwNzM3ZTg3ZDNjN
Dk2ZDlmNA==" –O –  https://tanteater.eng.mycompany.com:
4870/identity/api/tenants/MYCOMPANY/directories
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
HTTP/1.1 200 OK
  Server: Apache-Beach/1.1
  Cache-Control: no-cache, no-store
  Pragma: no-cache
  Expires: Wed, 31 Dec 1969 23:59:59 GMT
  Content-Type: application/json;charset=UTF-8
  Content-Length: 830
  Date: Sat, 01 Feb 2014 13:07:54 GMT
{"links":[],
 "content":[
    {"@type":"IdentityStore",
     "domain":"vcac.mycompany.com",
     "name":"openLDAPPromocom",
     "description":null,
     "alias":"promocom.com",
     "type":"LDAP",
     "userNameDn":"cn=promocomadmin,ou=promocom,dc=vcac,dc=mycompany,dc=com",
     "password":null,
     "url":"ldap://10.000.00.000:389",
     "groupBaseSearchDn":"ou=promocom,dc=vcac,dc=mycompany,dc=com",
     "userBaseSearchDn":"ou=promocom,dc=vcac,dc=mycompany,dc=com"
    },
    {"@type":"IdentityStore",
     "domain":"example.mycompany.com",
     "name":"openLDAPDemo",
     "description":null,
     "alias":"example.com",
     "type":"LDAP",
     "userNameDn":"cn=demoadmin,ou=demo,dc=example,dc=mycompany,dc=com",
     "password":null,
     "url":"ldap://10.000.00.000:389",
```

```
     "groupBaseSearchDn":"ou=demo,dc=example,dc=mycompany,dc=com",
     "userBaseSearchDn":"ou=demo,dc=example,dc=mycompany,dc=com"
    }],
   "metadata":{
        "size":20,
        "totalElements":2,
        "totalPages":1,
         "number":1,
         "offset":0
    }
}
```

Link an Identity Store to the Tenant

You can link an LDAP, Active Directory, or Native Active Directory identity store to the tenant y using the identity service.

Prerequisites

- Log in to the vRealize Automation server as a **system administrator**.
- Verify that you have access to a functional LDAP, Active Directory, or Native Active Directory identity server.
- Verify that you have the identity server details required for the JSON template.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/identity/api/tenants/*$tenantId*/directories/*$domainName* --data @*$inputFileName*.json
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$tenantId	Specifies the ID of the tenant.
userId	Specifies the ID of the user in the form name@domain.
$domainAlias	Specifies the domain alias.
$domainName	Specifies the domain of the identity store.
$grpBaseSearchDn	Specifies the group search base Distinguished Name.
$identityStoreName	Specifies a description of the new tenant.
$password	Specifies the password.
$identityStoreType	Specifies the identity store type for the tenant. The following values are supported: ■ LDAP ■ AD ■ NATIVE_AD

Input	Description
$identityServerUrl	Specifies the URL of the identity server.
$usrBaseSearchDn	Specifies the user search base Distinguished Name.
$usrNameDn	Specifies the Distinguished Name for the login user.

JSON Input File Template

You can use the following template to create a JSON input file. Replace the variables in the template with actual values in the file.

```
{
    "alias": "$domainAlias",
    "domain": "$domainName",
    "groupBaseSearchDn": "$grpBaseSearchDn",
    "name": "$identityStoreName",
    "password": "$password",
    "type": "$identityStoreType",
    "url": "$identityServerUrl",
    "userBaseSearchDn": "$usrBaseSearchDn",
    "userNameDn": "$usrNameDn"
}
```

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Species an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL which produces the result.
Content	Specifies an array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following information:
Id	Specifies the unique tenant identifier.
urlName	Specifies the name of the tenant as it appears in URLs.
Name	Specifies the name of the tenant for display purposes.
description	Specifies the long description of the tenant.
contactEmail	Specifies the primary contact email address.
Password	Unused.
defaultTenant	**`True`** if the corresponding tenant is the default tenant (vsphere.local).
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.

Property	Description
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example JSON Input File

You can call the following sample `ldap.json.txt` input file from the command line to specify necessary parameters.

```
{
    "alias": "example.com",
    "domain": "example.mycompany.com",
    "groupBaseSearchDn": "ou=demo,dc=example,dc=mycompany,dc=com",
    "name": "openLDAPDemo",
    "password": "password",
    "type": "LDAP",
    "url": "ldap://10.000.00.000:389",
    "userBaseSearchDn": "ou=demo,dc=example,dc=mycompany,dc=com",
    "userNameDn": "cn=demoadmin,ou=demo,dc=example,dc=mycompany,dc=com"
}
```

Example: curl Command

You can use the following command to call the example JSON text file and link an identity store to a tenant The command also tests that vRealize Automation can connect to the identity store successfully. If the command finishes successfully,vRealize Automation succeeded in connecting to your identity store.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/identity/api/tenants/development/directories/example.mycompany.com
--data @C:\Temp\ldap.json.txt
```

Example: API Explorer

You can use the following command to call the example JSON text file and link an identity store to a tenant The command also tests that vRealize Automation can connect to the identity store successfully. If the command finishes successfully, vRealize Automation succeeded in connecting to your identity store.

```
vcac-cli>rest put --headers --service identity --uri tenants/development/directories/
example.mycompany.com --data @example_ldap.json.txt
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
Request Headers
{
       Content-Type = application/json
             Accept = application/json
     Content-Length = 413
     Accept-Charset = big5, big5-hkscs, euc-jp, euc-kr, gb18030, gb2312, gbk,
ibm-thai, ibm00858, ibm01140, ibm01141, ibm01142, ibm01143, ibm01144, ibm01145,
ibm01146, ibm01147, ibm01148, ibm01149, ibm037, ibm1026, ibm1047, ibm273, ibm277,
ibm278, ibm280, ibm284, ibm285, ibm290, ibm297, ibm420, ibm424, ibm437, ibm500,
ibm775, ibm850, ibm852, ibm855, ibm857, ibm860, ibm861, ibm862, ibm863, ibm864,
```

```
ibm865, ibm866, ibm868, ibm869, ibm870, ibm871, ibm918, iso-2022-cn, iso-2022-jp,
iso-2022-jp-2, iso-2022-kr, iso-8859-1, iso-8859-13, iso-8859-15, iso-8859-2,
iso-8859-3, iso-8859-4, iso-8859-5, iso-8859-6, iso-8859-7, iso-8859-8, iso-8859-9,
jis_x0201, jis_x0212-1990, koi8-r, koi8-u, shift_jis, tis-620, us-ascii, utf-16,
utf-16be, utf-16le, utf-32, utf-32be, utf-32le, utf-8, windows-1250, windows-1251,
windows-1252, windows-1253, windows-1254, windows-1255, windows-1256, windows-1257,
windows-1258, windows-31j, x-big5-hkscs-2001, x-big5-solaris, x-compound_text,
x-euc-jp-linux, x-euc-tw, x-eucjp-open, x-ibm1006, x-ibm1025, x-ibm1046, x-ibm1097,
x-ibm1098, x-ibm1112, x-ibm1122, x-ibm1123, x-ibm1124, x-ibm1364, x-ibm1381,
x-ibm1383, x-ibm300, x-ibm33722, x-ibm737, x-ibm833, x-ibm834, x-ibm856, x-ibm874,
x-ibm875, x-ibm921, x-ibm922, x-ibm930, x-ibm933, x-ibm935, x-ibm937, x-ibm939,
x-ibm942, x-ibm942c, x-ibm943, x-ibm943c, x-ibm948, x-ibm949, x-ibm949c, x-ibm950,
x-ibm964, x-ibm970, x-iscii91, x-iso-2022-cn-cns, x-iso-2022-cn-gb, x-iso-8859-11,
x-jis0208, x-jisautodetect, x-johab, x-macarabic, x-maccentraleurope, x-maccroatian,
x-maccyrillic, x-macdingbat, x-macgreek, x-machebrew, x-maciceland, x-macroman,
x-macromania, x-macsymbol, x-macthai, x-macturkish, x-macukraine, x-ms932_0213,
x-ms950-hkscs, x-ms950-hkscs-xp, x-mswin-936, x-pck, x-sjis_0213, x-utf-16le-bom,
x-utf-32be-bom, x-utf-32le-bom, x-windows-50220, x-windows-50221, x-windows-874,
x-windows-949, x-windows-950, x-windows-iso2022jp
}
Response Headers
{
               Date = Wed, 29 Oct 2014 22:41:57 GMT
       Content-Type = application/json;charset=UTF-8
     Content-Length = 0
               Vary = Accept-Encoding,User-Agent
         Keep-Alive = timeout=15, max=100
         Connection = Keep-Alive
}
Successful
```

NOTE The following, or similar, error indicates that there is a problem connecting to the identity store. Examine your JSON input file and ensure that your identity store and its connection details are correct.

```
Command failed [Rest Error]: {Status code: 400}, {Error code: 90027} , {Error
Source: null}, {Error Msg: Cannot connect to the directory service.}, {System
Msg: 90027-Connection to directory service can't be established}
```

Search LDAP or Active Directory for a User

You can search the configured LDAP directory, Active Directory, or Native Active Directory for a user.

Prerequisites

- Log in to the vRealize Automation server as a **system administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/identity/api/tenants/*$tenantId*/principals/*$userId*
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$tenantId	Specifies the ID of the tenant.
$userId	Specifies the ID of the user in the form name@domain.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Specifies an array of link objects, each of which contains the following information:
rel	The name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service determines the other names.
href	Specifies the URL that produces the result.
Content	Specifies an array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following:
@type	Specifies the user name.
firstName	Specifies the first name of the user.
lastName	Specifies the last name of the user.
description	Specifies the description of the user.
emailAddress	Specifies the email address of the user.
locked	Specifies the Boolean flag indicating if the user is locked out.
disabled	Specifies the Boolean flag indicating if the user is disabled.
principalId	Specifies the principal ID of the user in username@domain format.
tenantName	Specifies the name of tenant to which user belongs.
name	Specifies the first and last name concatenated.

Example: curl Command

You can use the following command format to query the configured LDAP directory for a specific user.

```
curl –-insecure –H "Accept:text/xml"
–H "Authorization: Bearer $token"
https://$host/identity/api/tenants/$tenantId/principals/$userId
```

Example: API Explorer

You can use the following command to query the configured LDAP directory for the *$userId* value of **`tony`** in the **`example.mycompany.com`** domain.

```
vcac–cli>rest get –-service authentication –-uri tenants/MYCOMPANY1/principals/?criteria=tony
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
{
  "links" : [ ],
  "content" : [ {
    "@type" : "User",
    "firstName" : "Tony",
    "lastName" : "Anteater",
    "emailAddress" : "tony@example.mycompany.com",
    "locked" : false,
    "disabled" : false,
    "principalId" : {
      "domain" : "example.mycompany.com",
      "name" : "susan"
    },
    "tenantName" : "MYCOMPANY1",
    "name" : "Tony Anteater"
  } ]
}
```

Assign a User to a Role

You can assign a user to a role with the REST API identity service.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/identity/api/authorization/tenants/*$tenantId*/principals/*$principalId*/roles/*roleId*
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$tenantId	Specifies the ID of the tenant.
$principalId	Specifies the ID of the user in name@domain format.
$roleId	Specifies the ID of the user role.

Example: `curl` Command

You can use the following command string to submit a request to assign the user **`tony`** in the domain **`example.mycompany.com`** to the tenant administrator role. It provides empty braces for the required JSON payload. See "Search LDAP or Active Directory for a User," on page 28 for more information about getting the user name and domain values.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
"https://$host/identity/api/authorization/tenants/development/principals/
susan@example.mycompany.com/roles/CSP_TENANT_ADMIN/" --data "{}"
```

Example: API Explorer

You can use the following command string to submit a request to assign the user **`tony`** in the domain **`example.mycompany.com`** to the tenant administrator role. It provides the dummy JSON file `test.json` as the required payload. See "Search LDAP or Active Directory for a User," on page 28 for more information about getting the user name and domain values.

```
vcac-cli> rest put --headers --service authorization --uri tenants/
development/principals/tony@example.mycompany.com/roles/CSP_TENANT_ADMIN/
--data @test.json
```

Example: JSON Output

If the command is successful, the HTTP response body is empty except for a `204 No Content` status statement.

Display all Roles Assigned to a User

You can display all of the roles assigned to a user with the identity service.

Prerequisites

- Log in to the vRealize Automation server as a **system administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/identity/api/authorization/tenants/*$tenantId*/principals/*$principalId*/roles
$token	Specifies a valid HTTP bearer token with necessary credentials.
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$tenantId	Specifies the ID of the tenant.
principalId	Specifies the ID of the user in the form name@domain.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
id	Specifies the role ID.
name	Specifies the role name.
description	Specifies the role description.
status	Specifies the status of this role.
assignedPermissions	Specifies the set of permissions that are implied by this role assignment.

Example: curl Command

You can use the following command to list all the roles that are assigned to **tony@example.mycompany.com**.

```
curl –-insecure –H "Content-Type: application/json"
–H "Authorization: Bearer $token"
https://$host/identity/api/authorization/tenants/development/principals/
tony@example.mycompany.com/roles
```

Example: API Explorer

You can use the following command to list all the roles that are assigned to **susan@example.mycompany.com**.

```
vcac-cli>rest get –-service authorization –-uri tenants/development/principals/
susan@example.mycompany.com/roles
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
{
"links" : [ ],
"content" : [ {
"@type" : "SystemRole",
"id" : "ABX_TENANT_ADMIN",
"name" : "Tenant Administrator",
"description" : "ABX Tenant Administrator",
"assignedPermissions" : [ {
"id" : "CATALOG_CONSUME_TENANT_MGMT",
"name" : "Catalog Consume Tenant Management",
"description" : "Consume services, resources and manage requests on
behalf of any user within a Tenant",
"prereqAdminPermissions" : null
}, {
"id" : "MY_TENANT_MANAGEMENT",
"name" : "My Tenant Management",
"description" : "Manage my tenant.",
"prereqAdminPermissions" : null
}, {
"id" : "CATALOG_AUTHOR_TENANT",
"name" : "Catalog Tenant-level Author",
"description" : "Create, update and publish services, catalog items and actions shared across a
Tenant.",
"prereqAdminPermissions" : null
```

```
}, {
"id" : "GUI_MY_TENANT_MANAGEMENT",
"name" : "My Tenant Administration User Interface",
"description" : "Access my tenant administration GUI.",
"prereqAdminPermissions" : null
}, {
"id" : "CATALOG_ENTITLE_TENANT",
"name" : "Catalog Tenant-level Entitlement Management",
"description" : "Entitle services, catalog items and actions to all users within a tenant.",
"prereqAdminPermissions" : null
}, {
"id" : "FILE_EDIT_TENANT",
"name" : "Manage Tenant Files",
"description" : "Upload and delete files belonging to this tenant.",
"prereqAdminPermissions" : null
}, {
"id" : "TENANT_USER_DATA_MANAGEMENT",
"name" : "Manage user data (requests, items, tasks etc) within a tenant.",
"description" : "Manage user created objects belonging to the tenant.",
"prereqAdminPermissions" : null
}, {
"id" : "TENANT_ADMIN_ROLE_ASSIGNMENT",
"name" : "Tenant Administrator Role Assignment",
"description" : "Assign the tenant administrator role to other users.",
"prereqAdminPermissions" : null
}, {
"id" : "GUI_MY_TENANT_TUG_MANAGEMENT",
"name" : "My Tenant Identity Stores, Groups and Users Administration User Interfaces",
"description" : "Access my tenant identity stores, groups and users administration GUIs.",
"prereqAdminPermissions" : null
} ]
} ],
"metadata" : {
"size" : 20,
"totalElements" : 1,
"totalPages" : 1,
"number" : 1,
"offset" : 0
```


Requesting a Machine By Type

Your vRealize Automation API calls vary slightly based on your intended machine type.

See the following information to request a machine by type.

- "Requesting a Machine," on page 33
- "Requesting a vApp," on page 57
- "Requesting an Amazon EC2 Machine," on page 77

Requesting a Machine

You can request a machine using the REST API.

The checklist provides the tasks required to request a machine with the REST API. Perform the tasks in sequence.

Table 3-2. Requesting a Machine Checklist

Task	Details	Permissions
❑ Request an HTTP bearer token	See Chapter 2, "REST API Authentication," on page 9.	
❑ View a list of entitled catalog items	See "List Shared and Private Catalog Items," on page 34.	consumer and current business group member
❑ Find a catalog item by name	See "Find a Catalog Item by Name," on page 36.	consumer and current business group member
❑ Locate the blueprint values required to complete the machine request	See "Locate the Blueprint Values Required to Construct a Machine Request," on page 39.	consumer and current business group member
❑ Construct a machine request	See "Construct a JSON File For a Machine Request," on page 40.	consumer and current business group member
❑ Submit the request	See "Submit a Machine Request," on page 44.	consumer and current business group member
❑ View all of your requests	See "View All Your Requests," on page 47.	consumer and current business group member
❑ Find a resource by its request ID	See "Find a Resource by its Request ID," on page 51.	consumer and current business group member
❑ View the details of the submitted request	See "View the Details of a Machine Request," on page 53.	consumer and current business group member

List Shared and Private Catalog Items

You can retrieve a list of all shared catalog items that you can view in the catalog. Shared catalog items do not belong to a specific business group. You can also retrieve a list of all shared and private catalog items you can view, including their business groups.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/catalogItems
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
version	
id	Specifies the UUID Identifier of the object. Specifies the property type is string.
outputResourceTypeRef	Specifies the type of the resource that results from requesting the catalog item.
name	Specifies the user friendly name of the catalog item. Specifies the property type is string.
description	Specifies a short description of the catalog item. Specifies the property type is string.
status	Specifies the life cycle stage of the catalog item.
statusName	Specifies the life cycle status name, such as Active.
catalogItemTypeRef	Specifies the type of the catalog item.
serviceRef	Specifies the catalog service that contains the catalog item.
iconId	Specifies the associated icon representing this item.
organization	Specifies the subtenant and/or tenant to which this item belongs
providerBinding	Specifies the provider side identifier of this item.
forms	Specifies the forms that are associated with catalog items of this type.
callbacks	Specifies the call-backs to the provider that are supported by this catalog item.
isNoteworthy	Specifies if the catalog item should be highlighted to users for a period of time.
dateCreated	Specifies the date that this item was created in the catalog.
lastUpdatedDate	Specifies the date that this item was last updated in the catalog.
entitledOrganizations	Specifies the organizations in which the catalog item can be consumed by the current user.
catalogItem	Specifies the catalog item value.

Example: curl Command

Use the following command to retrieve information about all your available shared catalog items.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/entitledCatalogItems
```

Example: API Explorer

Use the following command to retrieve information about all your available shared catalog items.

```
vcac-cli> rest get --service catalog-service --u consumer/entitledCatalogItems
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
{
  "links" : [ ],
  "content" : [ {
    "@type" : "entitledCatalogItem",
    "id" : "65fbca06-a28e-46f3-bced-c6e5fb3a66f9",
    "version" : 1,
    "name" : "RHEL 6-vsphere",
```

```
    "description" : "",
    "status" : "PUBLISHED",
    "organization" : {
      "tenantRef" : "MYCOMPANY",
      "tenantLabel" : "ABTenant",
      "subtenantRef" : "cccd7a7e-5283-416b-beb0-45eb4e924dcb",
      "subtenantLabel" : "MyTestAgentBusinessGroup"
    },
    "providerBinding" : {
      "bindingId" : "e16edcf9-6a10-4bc7-98e2-a33361aeb857",
      "providerRef" : {
        "id" : "c6fb1980-75b4-4adc-ac71-020d75f61978",
        "label" : "iaas-service"
      }
    },
    "forms" : null,
    "callbacks" : null,
    "isNoteworthy" : true,
    "dateCreated" : "2014-02-14T21:53:39.072Z",
    "lastUpdatedDate" : "2014-02-14T21:54:07.756Z",
    "iconId" : "cafe_default_icon_genericCatalogItem",
    "catalogItemTypeRef" : {
      "id" : "Infrastructure.Virtual",
      "label" : "Virtual Machine"
    },
    "serviceRef" : {
      "id" : "e90847d7-03e1-45a9-8377-be77be03af6f",
      "label" : "Tyler's Service"
    },
    "outputResourceTypeRef" : {
      "id" : "Infrastructure.Virtual",
      "label" : "Virtual Machine"
    }
  } ],
  "metadata" : {
    "size" : 20,
    "totalElements" : 1,
    "totalPages" : 1,
    "number" : 1,
    "offset" : 0
  }
}
```


Find a Catalog Item by Name

You can locate a catalog item in the service catalog.

Prerequisites

- Log in to vRealize Automation as a **business group manager**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- "List Shared and Private Catalog Items," on page 34.

To filter your output, add one of the following strings to the end of the URL. Replace *my blueprint* with the actual name of the catalog item.

- ?$filter=name eq '*my blueprintt*'
- ?$filter=name%20eq%20%27my%20blueprint%27
- ?$filter=name+eq+%27my+blueprint%27

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/entitledCatalogItems?$filter=name+eq+%27my+custom+blueprint%27
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$catalogItemId	Specifies the ID of a catalog item.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	An array of link objects, each of which contains the following parts:
rel	The name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service determines the other names.
href	The URL which produces the result.
Content	An array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following properties:
@type	entitledCatalogItem
Id	The unique tenant identifier
version	
name	The name of the tenant for display purposes
description	Bief description of the tenant
status	Life cycle stage of this catalog item
organization	Subtenant and/or tenant to which this item belongs
tenantRef	ID of tenant
tenantLabel	Name of tenant
subtenantRef	ID of business group
subtenantLabel	Name of business group
providerBinding	Provider side identifier of this item
bindingId	binding ID
providerRef	Provider
forms	A specification for the various forms associated with catalog items of this type

Property	Description
callbacks	A specification for the various call-backs to the provider supported by this catalog item
isNoteworthy	Flag indicating that this catalog item should be highlighted to users for a period of time
dateCreated	Date this item was created in Catalog
lastUpdatedDate	Date this item was last updated in Catalog
iconId	Associated icon representing this item
catalogItemTypeRef	Type of the catalog item
serviceRef	Catalog Service containing this catalog item
outputResourceTypeRef	Type of the resource resulting from requesting this catalog item
Metadata	The paging-related data
Size	Maximum number of rows per page
totalElements	Number of rows returned
totalPages	Total number of pages of data available
Number	Current page number
Offset	Number of rows skipped

Example: curl Command

You can use the following command to retrieve all shared catalog items that you are allowed to view.

```
curl --insecure -H "Content-Type: application/json" -H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/entitledCatalogItems
```

Example: API Explorer

You can use the following command to retrieve all shared catalog items that you are allowed to view.

```
vcac-cli>rest get --service catalog-service --u consumer/entitledCatalogItems
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
{
  "links" : [ ],
  "content" : [ {
    "@type" : "entitledCatalogItem",
    "id" : "65fbca06-a28e-46f3-bced-c6e5fb3a66f9",
    "version" : 1,
    "name" : "RHEL 6-vsphere",
    "description" : "",
    "status" : "PUBLISHED",
    "organization" : {
      "tenantRef" : "MYCOMPANY",
      "tenantLabel" : "ABTenant",
      "subtenantRef" : "cccd7a7e-5283-416b-beb0-45eb4e924dcb",
      "subtenantLabel" : "MyTestAgentBusinessGroup"
    },
    "providerBinding" : {
      "bindingId" : "e16edcf9-6a10-4bc7-98e2-a33361aeb857",
      "providerRef" : {
        "id" : "c6fb1980-75b4-4adc-ac71-020d75f61978",
        "label" : "iaas-service"
```

```
      }
    },
    "forms" : null,
    "callbacks" : null,
    "isNoteworthy" : true,
    "dateCreated" : "2014-02-14T21:53:39.072Z",
    "lastUpdatedDate" : "2014-02-14T21:54:07.756Z",
    "iconId" : "cafe_default_icon_genericCatalogItem",
    "catalogItemTypeRef" : {
      "id" : "Infrastructure.Virtual",
      "label" : "Virtual Machine"
    },
    "serviceRef" : {
      "id" : "e90847d7-03e1-45a9-8377-be77be03af6f",
      "label" : "Tyler's Service"
    },
    "outputResourceTypeRef" : {
      "id" : "Infrastructure.Virtual",
      "label" : "Virtual Machine"
    }
  } ],
  "metadata" : {
    "size" : 20,
    "totalElements" : 1,
    "totalPages" : 1,
    "number" : 1,
    "offset" : 0
  }
}
```


Locate the Blueprint Values Required to Construct a Machine Request

You can find the blueprint values you need to complete a machine request by listing your entitled catalog items, and then locating the catalog item that corresponds to the machine blueprint.

Prerequisites

"List Shared and Private Catalog Items," on page 34.

Procedure

1 In the list of catalog items you requested, locate the catalog item that corresponds to the machine blueprint.

You can refer to the sample catalog item output in the example.

2 Locate the following attributes and their values in the catalog item output.

The actual values are required for the machine request.

Attribute	Sample Value	Description
id	65fbca06-a28e-46f3-bced-c6e5fb3a66f9	Catalog item ID
tenantRef	MYCOMPANY	Tenant name
subtenantRef	cccd7a7e-5283-416b-beb0-45eb4e924dcb	Business group ID
bindingId	e16edcf9-6a10-4bc7-98e2-a33361aeb857	Machine blueprint ID

Example: Excerpt from Sample Catalog Item Output

```
...
    "@type" : "CatalogItem",
    "id" : "65fbca06-a28e-46f3-bced-c6e5fb3a66f9",
    "version" : 1,
    "name" : "RHEL 6-vsphere",
    "description" : "",
    "status" : "PUBLISHED",
    "organization" : {
      "tenantRef" : "MYCOMPANY",
      "tenantLabel" : "QETenant",
      "subtenantRef" : "cccd7a7e-5283-416b-beb0-45eb4e924dcb",
      "subtenantLabel" : "MyTestAgentBusinessGroup"
    },
    "providerBinding" : {
      "bindingId" : "e16edcf9-6a10-4bc7-98e2-a33361aeb857",
      "providerRef" : {
        "id" : "c6fb1980-75b4-4adc-ac71-020d75f61978",
        "label" : "iaas-service"
      }
    },
...
```

What to do next

Use the actual values to create a JSON file for use in a machine request.

Construct a JSON File For a Machine Request

You construct a JSON file for use in a command line machine request.

Prerequisites

"Locate the Blueprint Values Required to Construct a Machine Request," on page 39

Process Overview

You can create a JSON file and include that JSON file as part of your command line input.

Create a JSON file to be used in a machine request.

1 Copy the JSON input file template to a new text file.

2 Substitute values for the input variables in the template.

3 Save the file with a `.json` extension.

JSON File Template Parameters

The following table describes the IDs, machine resources, and other information that you must add to your JSON file to create the JSON input parameters you need to submit the machine request.

Input Variable	Description
$catalogItemId	Specifies the *CatalogItem ID* value for the blueprint.
$tenantName	Specifies the *tenantRef* value for the blueprint.
$businessGroupId	Specifies the *subtenantRef* value for the blueprint, which equates to the business group ID. This value appears twice in the JSON template.
$username@fqdn	Specifies the user name of the consumer and business group manager account and the fully qualified domain name.

Input Variable	Description
$blueprintId	Specifies the *bindingId* in the blueprint.
$cpuCount	Specifies the number of CPUs for the provisioned machine.
$memorySize	Specifies the memory size, in integer form, for the provisioned machine specified in MB.
$leaseDays	Specifies the number of lease days, in integer form, for the provisioned machine.
$notes	Displays notes.
$disk0Size	Specifies the storage capacity, in integer form, for volume 0 in GB.
$disk0Letter	Specifies the drive letter, such as A, B, C, and so on, for volume 0.
$disk0Label	Specifies the label, in string text form, for volume 0.

For information about how to obtain the necessary values for the following JSON file variables, see "List Shared and Private Catalog Items," on page 34 and "Locate the Blueprint Values Required to Construct a Machine Request," on page 39.

```
{
    "@type": "CatalogItemRequest",
    "catalogItemRef": {
        "id": "$catalogItemId"
    },
    "organization": {
        "tenantRef": "$tenantName",
        "subtenantRef": "$businessGroupId"
    },
    "requestedFor": "$username@fqdn",
    "state": "SUBMITTED",
    "requestNumber": 0,
    "requestData": {
        "entries": [{
            "key": "provider-blueprintId",
            "value": {
                "type": "string",
                "value": "$blueprintId"
            }
        },
        {
            "key": "provider-provisioningGroupId",
            "value": {
                "type": "string",
                "value": "$businessGroupId"
            }
        },
        {
            "key": "requestedFor",
            "value": {
                "type": "string",
                "value": "$username@fqdn"
            }
        },
        {
            "key": "provider-VirtualMachine.CPU.Count",
            "value": {
```

```
                    "type": "integer",
                    "value": $cpuCount
                }
            },
            {
                "key": "provider-VirtualMachine.Memory.Size",
                "value": {
                    "type": "integer",
                    "value": $memorySize
                }
            },
            {
                "key": "provider-VirtualMachine.LeaseDays",
                "value": {
                    "type": "integer",
                    "value": $leaseDays
                }
            },
            {
                "key": "provider-__Notes",
                "value": {
                    "type": "string",
                    "value": "$notes"
                }
            },
            {
                "key": "provider-VirtualMachine.Disk0.Size",
                "value": {
                    "type": "string",
                    "value": "$disk0Size"
                }
            },
            {
                "key": "provider-VirtualMachine.Disk0.Letter",
                "value": {
                    "type": "string",
                    "value": "$disk0Letter"
                }
            },
            {
                "key": "provider-VirtualMachine.Disk0.Label",
                "value": {
                    "type": "string",
                    "value": "$disk0Label"
                }
            }]
        }
    }
```

Example: JSON Input File

You can use the following JSON input file sample when constructing a file for your own requirements.

```
{
    "@type": "CatalogItemRequest",
    "catalogItemRef": {
        "id": "65fbca06-a28e-46f3-bced-c6e5fb3a66f9"
    },
    "organization": {
        "tenantRef": "MYCOMPANY",
        "subtenantRef": "cccd7a7e-5283-416b-beb0-45eb4e924dcb"
    },
    "requestedFor": "fritz@example.mycompany.com",
    "state": "SUBMITTED",
    "requestNumber": 0,
    "requestData": {
        "entries": [{
            "key": "provider-blueprintId",
            "value": {
                "type": "string",
                "value": "e16edcf9-6a10-4bc7-98e2-a33361aeb857"
            }
        },
        {
            "key": "provider-provisioningGroupId",
            "value": {
                "type": "string",
                "value": "cccd7a7e-5283-416b-beb0-45eb4e924dcb"
            }
        },
        {
            "key": "requestedFor",
            "value": {
                "type": "string",
                "value": "fritz@example.mycompany.com"
            }
        },
        {
            "key": "provider-VirtualMachine.CPU.Count",
            "value": {
                "type": "integer",
                "value": 1
            }
        },
        {
            "key": "provider-VirtualMachine.Memory.Size",
            "value": {
                "type": "integer",
                "value": 1024
            }
        },
        {
            "key": "provider-VirtualMachine.LeaseDays",
            "value": {
                "type": "integer",
```

```
                "value": 30
            }
        },
        {
            "key": "provider-__Notes",
            "value": {
                "type": "string",
                "value": "MYCOMPANY machine"
            }
        },
        {
            "key": "provider-VirtualMachine.Disk0.Size",
            "value": {
                "type": "string",
                "value": "1"
            }
        },
        {
            "key": "provider-VirtualMachine.Disk0.Letter",
            "value": {
                "type": "string",
                "value": "C"
            }
        },
        {
            "key": "provider-VirtualMachine.Disk0.Label",
            "value": {
                "type": "string",
                "value": "main"
            }
        }]
    }
}
```

Submit a Machine Request

You can submit a machine request with the catalog service.

Prerequisites

- Log in to vRealize Automation.
- You must have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, you must have a valid HTTP bearer token that matches your login credentials.
- "Construct a JSON File For a Machine Request," on page 40.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/requests
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
version	
state	Specifies the item state, such as SUBMITTED.
approvalStatus	Specifies a status indicating whether this request has been approved, rejected, or is still pending some form of approval.
waitingStatus	Specifies a status indicating whether this request is waiting on any external users or services before it is able to progress.
requestNumber	Specifies a more user-friendly identifier for this request.
executionStatus	Specifies the current execution status of the request.
stateName	Specifies the localized state name.
phase	Specifies the current phase of the request, which is more coarse grained and easier for users to understand.
id	Specifies the universally unique identifier of this object.
iconId	Retrieves icon of this request based on the type of the object requested.
description	Contains a brief description of this request.
reasons	Specifies the business reasons entered by the requestor or owner of this request.
requestedFor	Specifies the ID of the user for whom this request is logged.
requestedBy	Specifies the ID of the user who actually submitted the request
organization	Subtenant and/or tenant owner of this request.
requestorEntitlementId	Specified the value of the requestorEntitlement setting.
preApprovalId	Specifies the ID of the preApproval setting.
postApprovalId	Specifies the ID of the approval generated for the post-provisioning workflow step.
dateCreated	Specifies the date when this request was sent to the catalog.
lastUpdated	Specifies the date when this request was last updated.
dateSubmitted	Specifies the date when this request was first submitted.
dateApproved	Specifies the date when this request was approved.
dateCompleted	Specifies the date when this request was completed.
quote	Contains a quote made by the provider defining the estimated cost(s) associated with the request and/or any resources provisioned as a result of the request.
requestCompletion	Contains additional request completion information.
requestData	Contains a map of the provider-specific field-value pairs collected for this request.

Property	Description
retriesRemaning	Specifies the number of attempts remaining to move this request from its current state to the next state in the request workflow. Some state transitions require calls to external services. These calls may fail due to transient errors such as momentary network errors. In these cases, the catalog will retry the call a number of times before failing. This property defines the number of retries remaining for the current state transition. When it reaches 0, the catalog will stop retrying and mark the request as failed. This property is reset to the default number of retries for every new operation that is triggered.
requestedItemName	Specifies the item name.
requestedItemDescription	Specifies the item description.

Example: curl Command

You can use the following command to submit a machine request.

```
curl --insecure -H "Content-Type: application/json" -H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/requests --verbose --data @C:/Temp/requestMachine.json
```

Examples: API Explorer

You can use the following command to submit a machine request and display request and response headers with the output. Uses the indicated JSON file or inline text as input.

```
rest post --headers --service catalog-service --uri consumer/requests --data @filename.json

rest post --headers --service catalog-service --uri consumer/requests --data 'inline_json_text'
```

Example: Output with Request and Response Headers

Display request and response headers with the output. Use the indicated JSON text file or inline text as input.

```
{
Accept = application/json
Content-Type = application/json
Content-Length = 2806
}
Response Headers
{
Date = Wed, 19 Feb 2014 20:58:34 GMT
ETag = "0"
Location = https://vcac152-013-208.mycompany.com/catalog-service/api/consumer/
     requests/3a5d9697-e3c8-476f-9754-29e773af4aa8
Content-Type = application/json;charset=UTF-8
Content-Length = 0
Vary = Accept-Encoding,User-Agent
Keep-Alive = timeout=15, max=100
Connection = Keep-Alive
}
null
```

View All Your Requests

You can view all of your requests with the catalog service.

Prerequisites

- Log in to vRealize Automation as a **business group manager**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/requests
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	An array of link objects, each of which contains the following:
rel	The name of the link. ■ "Self" refers to the object which was returned or requested. ■ "First', "previous", "next", and "last" refer to corresponding pages of pageable lists. ■ The application or service determines the other names.
href	The URL which produces the result.
Content	An array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following:
version	
state	The state
approvalStatus	A status indicating whether this request has been approved, rejected, or is still pending some form of approval
waitingStatus	A status indicating whether this request is waiting on any external users or services before it is able to progress
requestNumber	A more user-friendly identifier for this request
executionStatus	The current execution status of the request. Is it running?
stateName	The localized state name
phase	The current phase of the request, which is more coarse grained and easier for users to understand
id	The UUID Identifier of this object
iconId	Retrieves icon of this request based on the type of the object requested
description	Brief description of this request

Property	Description
reasons	The business reasons entered by the requestor or owner of this Request
requestedFor	ID of the user for whom this request is logged
requestedBy	ID of the user who actually submitted the request
organization	Subtenant and/or tenant owner of this Request
requestorEntitlementId	requestorEntitlement
preApprovalId	ID of the preApproval
postApprovalId	ID of the approval generated for the post-provisioning workflow step
dateCreated	The date when this request was sent to the catalog
lastUpdated	The date when this request was last updated
dateSubmitted	The date when this request was first submitted
dateApproved	The date when this Request was Approved
dateCompleted	The date when this Request was completed
quote	A quote made by the provider defining the estimated cost(s) associated with the request and/or any resources provisioned as a result
requestCompletion	Request completion information
requestData	A map of the provider-specific field-value pairs collected for this request
retriesRemaning	The number of attempts remaining to move this request from its current state to the next state in the request workflow. Some state transitions require calls to external services. These calls may fail due to transient errors such as momentary network errors. In these cases, the catalog will retry the call a number of times before failing. This property defines the number of retries remaining for the current state transition. When it reaches 0, the catalog will stop retrying and mark the request as failed. This property is reset to the default number of retries for every new operation that is triggered.
requestedItemName	
requestedItemDescription	
Metadata	The paging-related data
size	Maximum number of rows per page
totalElements	Number of rows returned
totalPages	Total number of pages of data available
number	Current page number
offset	Number of rows skipped

Example: curl Command

You can use the following command to view all of your requests.

```
curl –-insecure –H "Content-Type: application/json" –H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/requests
```

Example: API Explorer

You can use the following command to view all of your requests.

```
vcac-cli>rest get –-service catalog-service –-u consumer/requests
```

Example: JSON Output

The following JSON output is returned based on your command input.


```
{
  "links" : [ ],
  "content" : [ {
    "@type" : "CatalogItemRequest",
    "id" : "ec813a12-68c3-40a2-9a33-7efa38e8e2c9",
    "iconId" : "Travel_100.png",
    "version" : 5,
    "requestNumber" : 1,
    "state" : "SUCCESSFUL",
    "description" : "Attending conference",
    "reasons" : "Cuz I wanna go to Austrailia",
    "requestedFor" : "tony@example.vmware.com",
    "requestedBy" : "tony@example.vmware.com",
    "organization" : {
      "tenantRef" : "MYCOMPANY",
      "tenantLabel" : "QETenant",
      "subtenantRef" : "27b85c29-2624-459d-91d6-09ad071c6eb1",
      "subtenantLabel" : "Finance"
    },
    "requestorEntitlementId" : "7840175e-08e8-4152-a3f9-c53a4dd10f38",
    "preApprovalId" : null,
    "postApprovalId" : null,
    "dateCreated" : "2014-02-14T19:45:28.361Z",
    "lastUpdated" : "2014-02-14T19:48:27.690Z",
    "dateSubmitted" : "2014-02-14T19:45:28.361Z",
    "dateApproved" : null,
    "dateCompleted" : "2014-02-14T19:48:27.683Z",
    "quote" : {
      "leasePeriod" : {
        "type" : "timeSpan",
        "unit" : "DAYS",
        "amount" : 5
      },
      "leaseRate" : {
        "type" : "moneyTimeRate",
        "cost" : {
          "type" : "money",
          "currencyCode" : null,
          "amount" : 213.0
        },
        "basis" : {
          "type" : "timeSpan",
          "unit" : "DAYS",
          "amount" : 1
        }
      },
      "totalLeaseCost" : {
        "type" : "money",
        "currencyCode" : null,
        "amount" : 1065.0
      }
    },
```

```
"requestCompletion" : {
  "requestCompletionState" : "SUCCESSFUL",
  "completionDetails" : "The request was successfully completed"
},
"requestData" : {
  "entries" : [ {
    "key" : "provider-roomType",
    "value" : {
      "type" : "entityRef",
      "componentId" : null,
      "classId" : "roomType",
      "id" : "2",
      "label" : "Deluxe"
    }
  }, {
    "key" : "provider-workspaceType",
    "value" : {
      "type" : "entityRef",
      "componentId" : null,
      "classId" : "workspaceType",
      "id" : "1",
      "label" : "Private Office"
    }
  }, {
    "key" : "provider-arrivalDate",
    "value" : {
      "type" : "dateTime",
      "value" : "2014-02-21T19:44:00.000Z"
    }
  }, {
    "key" : "provider-address",
    "value" : {
      "type" : "string",
      "value" : "25 McLaren Street\nNorth Sydney, NSW 2060\nAUS"
    }
  }, {
    "key" : "provider-hotel",
    "value" : {
      "type" : "entityRef",
      "componentId" : null,
      "classId" : "hotel",
      "id" : "8",
      "label" : "Racecar Hotel"
    }
  }, {
    "key" : "provider-location",
    "value" : {
      "type" : "entityRef",
      "componentId" : null,
      "classId" : "location",
      "id" : "3",
      "label" : "AUS-Sydney-Napier"
    }
  }, {
    "key" : "provider-duration",
```

```
        "value" : {
          "type" : "integer",
          "value" : 5
        }
      } ]
    },
...
```

Find a Resource by its Request ID

You can use a request ID to find its corresponding resource with the catalog service.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- "Submit a Machine Request," on page 44.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/resources?$filter=request/id+eq+%27*requestId*%27
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
requestId	Specifies the ID of the request used for the resource, for example, **`9e3e2e33-2361-4c0a-8dcf-ff0a347bb08e`**.

Add one of the following strings to the URL in the command line. Replace *requestId* with the actual request ID.

- ?$filter=request/id eq '*requestId*'
- ?$filter=request/id%20eq%20%27*requestId*%27
- ?$filter=request/id+eq+%27*requestId*%27

Output

You can use supported input parameters to control the command output.

Property	Description
Links	Specifies an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL which produces the result.
work itemNumber	Displays a reference number for the work item.

Property	Description
id	Displays the universally unique ID of the entity.
version	Displays the object version number, supports optimistic concurrency.
assignees	Displays the list of work item assignees.
subTenantId	Optionally associates the work item with a specific business group granting users with management responsibilities over that business group permission to see the approval.
tenantId	Specifies the tenant ID for the work item.
callbackEntityId	Specifies the callback entity ID for the work item.
work itemType	Specifies the work item type for the work item.
completedDate	Specifies the date when the work item was completed.
assignedDate	Specifies the date when the work item was assigned.
createdDate	Specifies the created date of this instance.
assignedOrCompletedDate	Specifies the date to be displayed on UI.
formUrl	Specifies the URL from which the layout for this work item can be retrieved.
serviceId	Specifies the service ID that generated this work item instance.
work itemRequest	Specifies the corresponding work item request object.
status	Specifies the status of the work item.
completedBy	Specifies the principal ID of user who completed the work item.
availableActions	Contains a list of relevant work item actions.
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: `curl` Command

You can use the following command to find a resource by using its resource ID.

```
curl ––insecure –H "Content–Type: application/json"
–H "Authorization: Bearer $token"
https://$host/catalog–service/api/consumer/resources/?$filter=request/id+eq+%279e3
e2e33–2361–4c0a–8dcf–ff0a347bb08e%27
```

Example: API Explorer

You can use the following command to find a resource by using its resource ID.

```
rest get ––service catalog–service ––u consumer/resources/?$filter=request/id+eq+
%279e3e2e33–2361–4c0a–8dcf–ff0a347bb08e%27
```

Example: JSON Output

View the Details of a Machine Request

You can view the details of a machine request by using the catalog service.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- "Submit a Machine Request," on page 44.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/requests/*$requestId*
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$requestId	Specifies the request ID. See "View All Your Requests," on page 47 to view all of your requests and search for a request ID. The required request ID is located at the end of the Location URL in the response header. The request ID is located in the `Location` field of the response header if you submitted the request with the **`-headers`** flag.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
version	
state	Specifies the item state, such as SUBMITTED.
approvalStatus	Specifies a status indicating whether this request has been approved, rejected, or is still pending some form of approval.
waitingStatus	Specifies a status indicating whether this request is waiting on any external users or services before it is able to progress.
requestNumber	Specifies a more user-friendly identifier for this request.
executionStatus	Specifies the current execution status of the request.
stateName	Specifies the localized state name.
phase	Specifies the current phase of the request, which is more coarse grained and easier for users to understand.
id	Specifies the universally unique identifier of this object.
iconId	Retrieves icon of this request based on the type of the object requested.
description	Contains a brief description of this request.
reasons	Specifies the business reasons entered by the requestor or owner of this request.

Property	Description
requestedFor	Specifies the ID of the user for whom this request is logged.
requestedBy	Specifies the ID of the user who actually submitted the request
organization	Subtenant and/or tenant owner of this request.
requestorEntitlementId	Specified the value of the requestorEntitlement setting.
preApprovalId	Specifies the ID of the preApproval setting.
postApprovalId	Specifies the ID of the approval generated for the post-provisioning workflow step.
dateCreated	Specifies the date when this request was sent to the catalog.
lastUpdated	Specifies the date when this request was last updated.
dateSubmitted	Specifies the date when this request was first submitted.
dateApproved	Specifies the date when this request was approved.
dateCompleted	Specifies the date when this request was completed.
quote	Contains a quote made by the provider defining the estimated cost(s) associated with the request and/or any resources provisioned as a result of the request.
requestCompletion	Contains additional request completion information.
requestData	Contains a map of the provider-specific field-value pairs collected for this request.
retriesRemaning	Specifies the number of attempts remaining to move this request from its current state to the next state in the request workflow. Some state transitions require calls to external services. These calls may fail due to transient errors such as momentary network errors. In these cases, the catalog will retry the call a number of times before failing. This property defines the number of retries remaining for the current state transition. When it reaches 0, the catalog will stop retrying and mark the request as failed. This property is reset to the default number of retries for every new operation that is triggered.
requestedItemName	Specifies the item name.
requestedItemDescription	Specifies the item description.

Example: curl Command

You can use the following command to display the details of a request.

```
curl --insecure -H "Content-Type: application/json" -H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/requests/3a5d9697-e3c8-476f-9754-29e773af
```

Example: API Explorer

You can use the following command to display the details of a request. You can find the request ID in the Location field of the response header if you submitted the request with the **-headers** flag.

```
rest get --service catalog-service --u consumer/requests/3a5d9697-e3c8-476f-9754-29e773af
```

You can use the following command to display all your requests that have not completed.

```
rest get --service catalog-service --u "consumer/requests?limit=1&$filter=not (state eq
'SUCCESSFUL')"
```

Example: JSON Output

The following sample output contains information about the catalog item request **3a5d9697-e3c8-476f-9754-29e773af4aa8**.


```
{
  "@type" : "CatalogItemRequest",
  "id" : "3a5d9697-e3c8-476f-9754-29e773af4aa8",
  "iconId" : "cafe_default_icon_genericCatalogItem",
  "version" : 5,
  "requestNumber" : 5,
  "state" : "SUCCESSFUL",
  "description" : "MYCOMPANY machine",
  "reasons" : "New QE hire",
  "requestedFor" : "fritz@example.mycompany.com",
  "requestedBy" : "fritz@example.mycompany.com",
  "organization" : {
    "tenantRef" : "MYCOMPANY",
    "tenantLabel" : "QETenant",
    "subtenantRef" : "cccd7a7e-5283-416b-beb0-45eb4e924dcb",
    "subtenantLabel" : "MyTestAgentBusinessGroup"
  },
  "requestorEntitlementId" : "1d896d03-96ad-4aae-9900-75f49b57a6bf",
  "preApprovalId" : null,
  "postApprovalId" : null,
  "dateCreated" : "2014-09-19T20:58:35.854Z",
  "lastUpdated" : "2014-09-19T20:59:14.014Z",
  "dateSubmitted" : "2014-09-19T20:58:35.854Z",
  "dateApproved" : null,
  "dateCompleted" : "2014-09-19T20:59:13.994Z",
  "quote" : null,
  "requestCompletion" : {
    "requestCompletionState" : "SUCCESSFUL",
    "completionDetails" : "Request succeeded. Created tyler-prefix04."
  },
  "requestData" : {
    "entries" : [ {
      "key" : "provider-blueprintId",
      "value" : {
        "type" : "string",
        "value" : "e16edcf9-6a10-4bc7-98e2-a33361aeb857"
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.MaxCost",
      "value" : {
        "type" : "string",
        "value" : "0.0000000000"
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.TotalStorageSize",
      "value" : {
        "type" : "decimal",
        "value" : 1.0
      }
    }, {
      "key" : "provider-provisioningGroupId",
```

```
    "value" : {
      "type" : "string",
      "value" : "cccd7a7e-5283-416b-beb0-45eb4e924dcb"
    }
  }, {
    "key" : "provider-Cafe.Shim.VirtualMachine.AssignToUser",
    "value" : {
      "type" : "string",
      "value" : "fritz@example.mycompany.com"
    }
  }, {
    "key" : "provider-VirtualMachine.LeaseDays",
    "value" : {
      "type" : "integer",
      "value" : 30
    }
  }, {
    "key" : "provider-Cafe.Shim.VirtualMachine.Description",
    "value" : {
      "type" : "string",
      "value" : "MYCOMPANY machine"
    }
  }, {
    "key" : "provider-Cafe.Shim.VirtualMachine.Reason",
    "value" : {
      "type" : "string",
      "value" : "New QE hire"
    }
  }, {
    "key" : "provider-VirtualMachine.CPU.Count",
    "value" : {
      "type" : "integer",
      "value" : 1
    }
  }, {
    "key" : "provider-Cafe.Shim.VirtualMachine.NumberOfInstances",
    "value" : {
      "type" : "integer",
      "value" : 1
    }
  }, {
    "key" : "provider-VirtualMachine.Disk0.Letter",
    "value" : {
      "type" : "string",
      "value" : "C"
    }
  }, {
    "key" : "provider-__Notes",
    "value" : {
      "type" : "string",
      "value" : "MYCOMPANY machine"
    }
  }, {
    "key" : "provider-Cafe.Shim.VirtualMachine.MinCost",
    "value" : {
```

```
        "type" : "string",
        "value" : "0.0000000000"
      }
    }, {
      "key" : "provider-VirtualMachine.Disk0.Label",
      "value" : {
        "type" : "string",
        "value" : "main"
      }
    }, {
      "key" : "provider-VirtualMachine.Disk0.Size",
      "value" : {
        "type" : "string",
        "value" : "1"
      }
    }, {
      "key" : "provider-VirtualMachine.Memory.Size",
      "value" : {
        "type" : "integer",
        "value" : 1
      }
    } ]
  },
  "retriesRemaining" : 3,
  "phase" : "SUCCESSFUL",
  "executionStatus" : "STOPPED",
  "waitingStatus" : "NOT_WAITING",
  "approvalStatus" : "POST_APPROVED",
  "catalogItemRef" : {
    "id" : "65fbca06-a28e-46f3-bced-c6e5fb3a66f9",
    "label" : "RHEL 6-vsphere"
  }
}
```


Requesting a vApp

You can request a vCloud Director vApp using the REST API.

The checklist provides the tasks required to request a vApp with the REST API. Perform the tasks in sequence.

In addition to the checklist sequence, verify that your environment meets the following prerequisite conditions:

- To request a vApp using the REST API, vCloud Director must be integrated into vRealize Automation as described in the vRealize Automation documentation.
- The vCloud endpoint in vRealize Automation must use the instance URL, for example https://vcd-instance, and not the full vCloud Director URL, for example https://vcd-instance/org/tenantId.
- The same user account must exist in vCloud Director and in vRealize Automation. The credentials for creating the vCloud endpoint are not sufficient for requesting vApps.

Table 3-3. Requesting a vApp Checklist

Task	Details	Permissions
❑ Request an HTTP bearer token	See Chapter 2, "REST API Authentication," on page 9.	
❑Find the published vApp (vCloud) blueprint	See "Find the Published Blueprint for a vApp Request," on page 58.	consumer and current business group member
❑ Construct a vApp request	See "Construct a JSON File for a vApp Request," on page 61.	consumer and current business group member
❑ Submit a vApp request	See "Submit a vApp Request," on page 70.	consumer and current business group member
❑ Check your vApp request status	See "View the Details of a vApp Request," on page 72.	consumer and current business group member

Find the Published Blueprint for a vApp Request

You can find the catalog item that corresponds to the vApp (vCloud) blueprint to use for a vApp request by retrieving a page of published blueprint catalog items.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- View a list of catalog items. See "List Shared and Private Catalog Items," on page 34.

Process Overview

You can use the following sequence to find a vApp blueprint for use in creating a vApp request.

1 From the list of your entitled catalog items, find the catalog item that corresponds to the vApp blueprint to use for the request. You can search on the catalog item ID `Infrastructure.vApp` to locate a published vApp blueprint.

2 In the catalog item output that contains a catalog item ID `Infrastructure.vApp` entry, locate the following entries that are required by the vApp request:

   - Catalog item ID, for example **`c2cacf7c-b3c8-47fb-a938-2c09910b6713`**
   - Tenant reference, for example **`sqa`**
   - Binding ID (blueprint), for example **`46548940-eb20-4368-9e73-c1685cda8c64`**
   - Subtenant reference (business group), for example `name1`

     If the subtenant reference value is null, you do need to enter a subtenant reference value in the vApp request

If you request information about a catalog item for which you are not entitled, or the blueprint catalog item is not published, your request is rejected.

Input

You can use supported input parameters to control the command output.

Property	Description
&page	Specifies a page number. Specifies the default value is 1.
?limit	Specifies the number of entries displayed on a page. Specifies the default value is 10.
$orderby	Specifies how the results are sorted and paginated.
$skip	Specifies how many results to skip before computing pagination.
$filter	Specifies a Boolean expression to define whether a particular entry be included in the response. Each API supports a different set of filterable fields.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
version	
id	Specifies the UUID Identifier of the object. Specifies the property type is string.
outputResourceTypeRef	Specifies the type of the resource that results from requesting the catalog item.
name	Specifies the user friendly name of the catalog item. Specifies the property type is string.
description	Specifies a short description of the catalog item. Specifies the property type is string.
status	Specifies the life cycle stage of the catalog item.
statusName	Specifies the life cycle status name, such as Active.
catalogItemTypeRef	Specifies the type of the catalog item.
serviceRef	Specifies the catalog service that contains the catalog item.
iconId	Specifies the associated icon representing this item.
organization	Specifies the subtenant and/or tenant to which this item belongs
providerBinding	Specifies the provider side identifier of this item.
forms	Specifies the forms that are associated with catalog items of this type.
callbacks	Specifies the callbacks to the provider that are supported by this catalog item.
isNoteworthy	Specifies if the catalog item should be highlighted to users for a period of time.
dateCreated	Specifies the date that this item was created in the catalog.
lastUpdatedDate	Specifies the date that this item was last updated in the catalog.

For sample curl and REST API calls, sample output is provided.

Example: `curl` Command

You can use the following command to display the catalog items that you are entitled to view, including published vApp (vCloud) blueprints, one page at a time with a maximum of 10 items on each page.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/catalogItems?limit=10&page=1
```

Example: API Explorer

You can use the following command to display the catalog items that you are entitled to view, including published vApp (vCloud) blueprints, one page at a time with a maximum of 10 items on each page.

```
rest get --service catalog-service --u consumer/catalogItems?limit=10&page=1
```

Example: JSON Output

The following JSON output is returned based on your command input.

The following highlighted items are required when you submit a request for a vApp.


```
{
    "@type" : "CatalogItem",
    "id" : "c2cacf7c-b3c8-47fb-a938-2c09910b6713",
    "version" : 1,
    "name" : "vApp",
    "description" : "",
    "status" : "PUBLISHED",
    "organization" : {
      "tenantRef" : "acx",
      "tenantLabel" : "ACX",
      "subtenantRef" : null,
      "subtenantLabel" : null
    },
    "providerBinding" : {
      "bindingId" : "46548940-eb20-4368-9e73-c1685cda8c64",
      "providerRef" : {
        "id" : "ba3b18dd-a891-48d2-a3e7-faed239990ed",
        "label" : "iaas-service"
      }
    },
    "forms" : null,
    "callbacks" : null,
    "isNoteworthy" : false,
    "dateCreated" : "2014-09-18T23:50:52.858Z",
    "lastUpdatedDate" : "2014-11-11T23:52:14.407Z",
    "iconId" : "cafe_default_icon_genericCatalogItem",
    "catalogItemTypeRef" : {
      "id" : "Infrastructure.vApp",
      "label" : "vCD vApp"
    },
    "serviceRef" : {
      "id" : "ca6b9988-fe07-4b25-b465-3e0c905b7aad",
      "label" : "vCD service"
    },
    "outputResourceTypeRef" : {
      "id" : "Infrastructure.vApp",
      "label" : "vCD vApp"
    }
  } ],
  "metadata" : {
    "size" : 10,
    "totalElements" : 3,
    "totalPages" : 1,
    "number" : 1,
    "offset" : 0
  }
```

Construct a JSON File for a vApp Request

You construct a JSON file for use in a command line vApp request.

Prerequisites

"Find the Published Blueprint for a vApp Request," on page 58

Process Overview

You can create a JSON file and include that JSON file as part of your command line input.

Create a JSON file to be used in a machine request.

1 Copy the JSON input file template to a new text file.

2 Substitute values for the input variables in the template.

3 Save the file with a `.json` extension.

JSON Template File Parameters

The following table describes the IDs, machine resources, and other information that you must add to your JSON file to create the JSON input parameters you need to submit the machine request.

Table 3-4. JSON Template Values

Value	Description
catalog_item_ID	Specifies the value of `CatalogItem` ID in the machine blueprint catalog item.
tenant_name	Specifies the value of `tenantRef` in the machine blueprint catalog item.
business_group_ID	Specifies the value of `subtenantRef` in the machine blueprint catalog item.
username@fqdn	Specifies the user name of the consumer and business group manager account and fully qualified domain name.
blueprint_ID	Specifies the value of `bindingId` in the machine blueprint catalog item.
provisioning_group_ID	Specifies the provisioning group ID of the blueprint.
cpu_count	Specifies the number of CPUs for the provisioned machine. The value type is Integer.
memory_size	Specifies the memory size of the provisioned machine in MB. The value type is Integer.
lease_days	Specifies the number of lease days for provisioned machine. The value type is Integer.
notes	Specifies the Notes string(s) content.
disk0_size	Specifies the storage capacity of volume 0 in GB. The value type is Integer.
disk0_letter	Specifies the drive letter (A, B, C, and so on) for volume 0. The value type is String
disk0_label	Specifies the label for volume 0.
key#	Specifies the key identifier in the map of key, value pairs.
value	Specifies the value that corresponds to the key.
allocation_type	Specifies the vApp allocation type using an enumeration of the following values. ■ 0 = Guaranteed ■ 1= Limited ■ 2 = Unlimited
storage_size	Specifies the storage size for the instance.
description	Specifies the description for the request.

Table 3-4. JSON Template Values (Continued)

Value	Description
numberOfInstances	Specifies the number of machines to be deployed as a part of this vApp.
reason	Specifies the reason for requesting the vApp blueprint.
assigned_user	Specifies the user to which the machine is assigned.
min_cost	Specifies the minimum cost.
max_cost	Specifies the maximum cost.
provision_into_network	Specifies the type of network to use for provisioning. ■ 0 = all ■ 1 = subnet ■ 2 = non VPC (Non-virtual private cloud)

JSON Input File Template

The highlighted items in the following sample correlate to the JSON Template Input Values table.

```
{
    "@type": "CatalogItemRequest",
    "catalogItemRef": {
        "id": "catalog_item_ID"
    },
    "organization": {
        "tenantRef": "tenant_name",
        "subtenantRef": "business_group_ID"
    },
    "requestedFor": "username@fqdn",
    "state": "SUBMITTED",
    "requestNumber": 0,
    "requestData": {
        "entries": [{
            "key": "provider-blueprintId",
            "value": {
                "type": "string",
                "value": "blueprint_ID"
            }
        },
        {
            "key": "provider-provisioningGroupId",
            "value": {
                "type": "string",
                "value": "subtenantRef"
            }
        },
        {
            "key": "requestedFor",
            "value": {
                "type": "string",
                "value": "username@fqdn"
            }
        },
        {
            "key": "provider-VirtualMachine.CPU.Count",
            "value": {
```

```
                "type": "integer",
                "value": cpu_count
            }
        },
        {
            "key": "provider-VirtualMachine.Memory.Size",
            "value": {
                "type": "integer",
                "value": memory_size
            }
        },
        {
            "key": "provider-VirtualMachine.LeaseDays",
            "value": {
                "type": "integer",
                "value": lease_days
            }
        },
        {
            "key": "provider-__Notes",
            "value": {
                "type": "string",
                "value": "notes"
            }
        },
        {
            "key": "provider-VirtualMachine.Disk0.Size",
            "value": {
                "type": "string",
                "value": "disk0_size"
            }
        },
        {
            "key": "provider-VirtualMachine.Disk0.Letter",
            "value": {
                "type": "string",
                "value": "disk0_letter"
            }
        },
        {
            "key": "provider-VirtualMachine.Disk0.Label",
            "value": {
                "type": "string",
                "value": "disk0_label"
            }
        }]
    }
}
{
   "@type": "CatalogItemRequest",
   "catalogItemRef": {
      "id": catalog_item_ID
   },
   "organization": {
      "tenantRef": tenant_name,
```

```
      "subtenantRef": business_group_ID
   },
   "requestedFor": username@fqdn,
   "state": "SUBMITTED",
   "requestNumber": 0,
   "requestData": {
      "entries": [
         {
            "key": "provider-blueprintId",
            "value": {
               "type": "string",
               "value": blueprint_ID
            }
         },
         {
            "key": "provider-provisioningGroupId",
            "value": {
               "type": "string",
               "value": provisioning_group_ID
            }
         },
         {
            "key": "requestedFor",
            "value": {
               "type": "string",
               "value": username@fqdn
            }
         },
         {
            "key": "provider-VirtualMachine.LeaseDays",
            "value": {
               "type": "integer",
               "value": lease_days
            }
         },
         {
            "key": "provider-__Notes",
            "value": {
               "type": "string",
               "value": notes
            }
         },
         {
            "key": "provider-ASCT-1.VirtualMachine.CPU.Count",
            "value": {
               "type": "string",
               "value": cpu_count
            }
         },
         {
            "key": "provider-ASCT-1.VirtualMachine.Memory.Size",
            "value": {
               "type": "string",
               "value": memory_size
            }
```

```
            },
            {
               "key": "provider-ASCT-1.__Notes",
               "value": {
                  "type": "string",
                  "value": notes
               }
            },
            {
               "key": "provider-ASCT-1.VirtualMachine.Disk0.Size",
               "value": {
                  "type": "string",
                  "value": disk0_size
               }
            },
            {
               "key": "provider-ASCT-1.VirtualMachine.Disk0.Letter",
               "value": {
                  "type": "string",
                  "value": disk0_letter
               }
            },
            {
               "key": "provider-__MultiMachine.Provision.NumberOfInstances",
               "value": {
                  "type": "string",
                  "value": "<ArrayOfKeyValueOfintint xmlns:i=\\\"http://www.w3.org/2001/XMLSchema-
instance\\\" xmlns=\\\"http://schemas.microsoft.com/2003/10/Serialization/Arrays\\\">\r\n
<KeyValueOfintint>\r\n    <Key>key#<\/Key>\r\n    <Value>value<\/Value>\r\n
<\/KeyValueOfintint>\r\n<\/ArrayOfKeyValueOfintint>"
               }
            },
            {
               "key": "provider-__requested_allocation_type",
               "value": {
                  "type": "string",
                  "value": allocation_type            }
            },
            {
               "key": "provider-Cafe.Shim.VirtualMachine.TotalStorageSize",
               "value": {
                  "type": "decimal",
                  "value": storage_size
               }
            },
            {
               "key": "provider-Cafe.Shim.VirtualMachine.Description",
               "value": {
                  "type": "string",
                  "value": description
               }
            },
            {
               "key": "provider-Cafe.Shim.VirtualMachine.NumberOfInstances",
               "value": {
```

```
      "type": "integer",
      "value": numberOfInstances
   }
},
{
   "key": "provider-Cafe.Shim.VirtualMachine.Reason",
   "value": {
      "type": "string",
      "value": reason
   }
},
{
   "key": "provider-Cafe.Shim.VirtualMachine.AssignToUser",
   "value": {
      "type": "string",
      "value": assigned_user
   }
},
{
   "key": "provider-Cafe.Shim.VirtualMachine.MinCost",
   "value": {
      "type": "string",
      "value": min_cost
   }
},
{
   "key": "provider-Cafe.Shim.VirtualMachine.MaxCost",
   "value": {
      "type": "string",
      "value": max_cost
   }
},
{
   "key": "provider-Cafe.Shim.VirtualMachine.ProvisionInto",
   "value": {
      "type": "string",
      "value": provision_into_network

   }
},
{
   "key": "description",
   "value": {
      "type": "string",
      "value": description

   }
},
{
   "key": "reasons",
   "value": {
      "type": "string",
      "value": reason
   }
```

```
         }
      ]
   }
}
```

Example: JSON Input File

You can use the following JSON input file sample when constructing a file for your own requirements.

You can omit `requestNumber` from the JSON input file.

```
{
   "@type": "CatalogItemRequest",
   "catalogItemRef": {
      "id": "c2cacf7c-b3c8-47fb-a938-2c09910b6713"
   },
   "organization": {
      "tenantRef": "abx",
      "subtenantRef": "43a2f89a-c04e-4941-abc5-b4dc68a2810d"
   },
   "requestedFor": "Auto.admin@abx.local",
   "state": "SUBMITTED",
   "requestNumber": 0,
   "requestData": {
      "entries": [
         {
            "key": "provider-blueprintId",
            "value": {
               "type": "string",
               "value": "46548940-eb20-4368-9e73-c1685cda8c64"
            }
         },
         {
            "key": "provider-provisioningGroupId",
            "value": {
               "type": "string",
               "value": "43a2f89a-c04e-4941-abc5-b4dc68a2810d"
            }
         },
         {
            "key": "requestedFor",
            "value": {
               "type": "string",
               "value": "Auto.admin@abx.local"
            }
         },
         {
            "key": "provider-VirtualMachine.LeaseDays",
            "value": {
               "type": "integer",
               "value": 2
            }
         },
         {
            "key": "provider-__Notes",
            "value": {
               "type": "string",
```

```
                  "value": "A simple vCD provisioning scenario."
               }
            },
            {
               "key": "provider-ASCT-1.VirtualMachine.CPU.Count",
               "value": {
                  "type": "string",
                  "value": "1"
               }
            },
            {
               "key": "provider-ASCT-1.VirtualMachine.Memory.Size",
               "value": {
                  "type": "string",
                  "value": "1"
               }
            },
            {
               "key": "provider-ASCT-1.__Notes",
               "value": {
                  "type": "string",
                  "value": ""
               }
            },
            {
               "key": "provider-ASCT-1.VirtualMachine.Disk0.Size",
               "value": {
                  "type": "string",
                  "value": "1"
               }
            },
            {
               "key": "provider-ASCT-1.VirtualMachine.Disk0.Letter",
               "value": {
                  "type": "string",
                  "value": "c"
               }
            },
            {
               "key": "provider-__MultiMachine.Provision.NumberOfInstances",
               "value": {
                  "type": "string",
                  "value": "<ArrayOfKeyValueOfintint xmlns:i=\\\"http://www.w3.org/2001/XMLSchema-
instance\\\" xmlns=\\\"http://schemas.microsoft.com/2003/10/Serialization/Arrays\\\">\r\n
<KeyValueOfintint>\r\n    <Key>1<\/Key>\r\n    <Value>1<\/Value>\r\n
<\/KeyValueOfintint>\r\n<\/ArrayOfKeyValueOfintint>"
               }
            },
            {
               "key": "provider-__requested_allocation_type",
               "value": {
                  "type": "string",
                  "value": "2"
               }
            },
```

```
            {
               "key": "provider-Cafe.Shim.VirtualMachine.TotalStorageSize",
               "value": {
                  "type": "decimal",
                  "value": 0
               }
            },
            {
               "key": "provider-Cafe.Shim.VirtualMachine.Description",
               "value": {
                  "type": "string",
                  "value": "A simple vApp provisioning scenario."
               }
            },
            {
               "key": "provider-Cafe.Shim.VirtualMachine.NumberOfInstances",
               "value": {
                  "type": "integer",
                  "value": 1
               }
            },
            {
               "key": "provider-Cafe.Shim.VirtualMachine.Reason",
               "value": {
                  "type": "string",
                  "value": "Requesting a vApp."
               }
            },
            {
               "key": "provider-Cafe.Shim.VirtualMachine.AssignToUser",
               "value": {
                  "type": "string",
                  "value": "Auto.admin@abx.local"
               }
            },
            {
               "key": "provider-Cafe.Shim.VirtualMachine.MinCost",
               "value": {
                  "type": "string",
                  "value": "4.0000000000"
               }
            },
            {
               "key": "provider-Cafe.Shim.VirtualMachine.MaxCost",
               "value": {
                  "type": "string",
                  "value": "4.0000000000"
               }
            },

      {
               "key": "provider-Cafe.Shim.VirtualMachine.ProvisionInto",
               "value": {
                  "type": "string",
                  "value": "2"
```

```
            }
         },
         {
            "key": "description",
            "value": {
               "type": "string",
               "value": "A simple vApp provisioning scenario."
            }
         },
         {
            "key": "reasons",
            "value": {
               "type": "string",
               "value": "Requesting a vApp."
            }
         }
      ]
   }
}
```

Submit a vApp Request

You can submit a vApp request with the catalog service that contains instruction either in he command line or in a specified JSON file.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- "Construct a JSON File for a vApp Request," on page 61.

Input

You can use supported input parameters to control the command output.

Inputs	Description
URL	Specifies the API resource URL in the following format. https://*$host*/catalog-service/api/consumer/requests
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
--headers	Optional. Specifies that request and response headers are displayed with the output. The response header includes the location of the newly created resource.
--uri *pathname*	Specifies that the request is submitted to the designated location.

Inputs	Description
--data @filename	Specifies that the named JSON file be used as input, as in the following example. `--data @MyRequest.json`
--data '*inline_json_text*'	Specifies that inline JSON text be used as input, as in the following example. `--data 'inline_json_text'`

Example: `curl` Command

You can use the following command to submit a vApp request, where `C:/Temp/requestMachine.json` is the file name and location of the JSON file that contains the necessary information for processing the request.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/requests --verbose --data @C:/Temp/requestMachine.json
```

Example: API Explorer

You can use the following command format to submit a vApp request using either a JSON file or JSON command line input.

```
rest post --headers --service catalog-service --uri consumer/requests --data @filename
```

```
rest post --headers --service catalog-service --uri consumer/requests --data 'inline_json_text'
```

Example: JSON Output with Request and Response Headers

You can display request and response headers with the output. Reference the following example to submit a vApp request by using JSON inline text that contains the necessary information, rather than by using a JSON file that contains the necessary information.

In this example, the **`@C:\vcd.txt`** entry calls a file a `vcd.txt` file that contains the request payload.

```
rest post --headers --service catalog-service --u consumer/requests --d @C:\vcd.txt
Request Headers
{
             Accept = application/json
       Content-Type = application/json
     Content-Length = 2721
     Accept-Charset = big5, big5-hkscs, euc-jp, euc-kr, gb18030, gb2312, gbk, ibm-thai,
ibm00858, ibm01140, ibm01141, ibm01142, ibm01143, ibm01144, ibm01145, ibm01146, ibm01147,
ibm01148, ibm01149, ibm
37, ibm500, ibm775, ibm850, ibm852, ibm855, ibm857, ibm860, ibm861, ibm862, ibm863, ibm864,
ibm865, ibm866, ibm868, ibm869, ibm870, ibm871, ibm918, iso-2022-cn, iso-2022-jp, iso-2022-jp-2,
iso-2022-kr,
iso-8859-7, iso-8859-8, iso-8859-9, jis_x0201, jis_x0212-1990, koi8-r, koi8-u, shift_jis,
tis-620, us-ascii, utf-16, utf-16be, utf-16le, utf-32, utf-32be, utf-32le, utf-8, windows-1250,
windows-1251, w
ndows-31j, x-big5-hkscs-2001, x-big5-solaris, x-euc-jp-linux, x-euc-tw, x-eucjp-open, x-ibm1006,
x-ibm1025, x-ibm1046, x-ibm1097, x-ibm1098, x-ibm1112, x-ibm1122, x-ibm1123, x-ibm1124, x-
ibm1364, x-ibm
x-ibm922, x-ibm930, x-ibm933, x-ibm935, x-ibm937, x-ibm939, x-ibm942, x-ibm942c, x-ibm943, x-
ibm943c, x-ibm948, x-ibm949, x-ibm949c, x-ibm950, x-ibm964, x-ibm970, x-iscii91, x-iso-2022-cn-
cns, x-iso-20
ccroatian, x-maccyrillic, x-macdingbat, x-macgreek, x-machebrew, x-maciceland, x-macroman, x-
macromania, x-macsymbol, x-macthai, x-macturkish, x-macukraine, x-ms932_0213, x-ms950-hkscs, x-
ms950-hkscs-x
```

```
, x-windows-50221, x-windows-874, x-windows-949, x-windows-950, x-windows-iso2022jp
        subtenantId =
}
Response Headers
{
               Date = Tue, 11 November 2014 23:57:43 GMT
               ETag = "0"
           Location = https://abx148-084-124.mycompany.com/catalog-
service/api/consumer/requests/510051b5-52ce-45db-8889-d4eeabf68da1
       Content-Type = application/json;charset=UTF-8
     Content-Length = 0
               Vary = Accept-Encoding,User-Agent
         Keep-Alive = timeout=15, max=100
         Connection = Keep-Alive
}
Null
```

View the Details of a vApp Request

You can view the details of your vApp request by using the catalog service.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the request ID (*$requestId*) of the request for which to view status. See "View All Your Requests," on page 47.
- "Submit a vApp Request," on page 70.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/requests/*$requestId*
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$requestId	Specifies a request ID. Specifies the UUID of the request. See "View All Your Requests," on page 47 to view all of your requests and search for the request ID. The required request ID is located at the end of the Location URL in the response header. The request ID is located in the `Location` field of the response header if you submitted the request with the **`-headers`** flag.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
version	
state	Specifies the item state, such as SUBMITTED.
approvalStatus	Specifies a status indicating whether this request has been approved, rejected, or is still pending some form of approval.
waitingStatus	Specifies a status indicating whether this request is waiting on any external users or services before it is able to progress.
requestNumber	Specifies a more user-friendly identifier for this request.
executionStatus	Specifies the current execution status of the request.
stateName	Specifies the localized state name.
phase	Specifies the current phase of the request, which is more coarse grained and easier for users to understand.
id	Specifies the universally unique identifier of this object.
iconId	Retrieves icon of this request based on the type of the object requested.
description	Contains a brief description of this request.
reasons	Specifies the business reasons entered by the requestor or owner of this request.
requestedFor	Specifies the ID of the user for whom this request is logged.
requestedBy	Specifies the ID of the user who actually submitted the request
organization	Subtenant and/or tenant owner of this request.
requestorEntitlementId	Specified the value of the requestorEntitlement setting.
preApprovalId	Specifies the ID of the preApproval setting.
postApprovalId	Specifies the ID of the approval generated for the post-provisioning workflow step.
dateCreated	Specifies the date when this request was sent to the catalog.
lastUpdated	Specifies the date when this request was last updated.
dateSubmitted	Specifies the date when this request was first submitted.
dateApproved	Specifies the date when this request was approved.
dateCompleted	Specifies the date when this request was completed.
quote	Contains a quote made by the provider defining the estimated cost(s) associated with the request and/or any resources provisioned as a result of the request.
requestCompletion	Contains additional request completion information.
requestData	Contains a map of the provider-specific field-value pairs collected for this request.
retriesRemaning	Specifies the number of attempts remaining to move this request from its current state to the next state in the request workflow. Some state transitions require calls to external services. These calls may fail due to transient errors such as momentary network errors. In these cases, the catalog will retry the call a number of times before failing. This property defines the number of retries remaining for the current state transition. When it reaches 0, the catalog will stop retrying and mark the request as failed. This property is reset to the default number of retries for every new operation that is triggered.
requestedItemName	Specifies the item name.
requestedItemDescription	Specifies the item description.

Example: `curl` Command

You can display the status of a vApp request, where **`510051b5-52ce-45db-8889-d4eeabf68da1`** is the value of the request ID.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/requests/510051b5-52ce-45db-8889-d4eeabf68da1
```

Example: API Explorer

You can display the status of a vApp request by using the request ID, where **`510051b5-52ce-45db-8889-d4eeabf68da1`** is the value of the request ID.

```
rest get --service catalog-service --uri consumer/requests/510051b5-52ce-45db-8889-d4eeabf68da1
```

Example: JSON Output

The following JSON output is returned based on your command input.

The follows sample illustrates example output for a request to query the status of a vApp, where `510051b5-52ce-45db-8889-d4eeabf68da1` is the value of the request ID.

```
{
  "@type" : "CatalogItemRequest",
  "id" : "510051b5-52ce-45db-8889-d4eeabf68da1",
  "iconId" : "cafe_default_icon_genericCatalogItem",
  "version" : 3,
  "requestNumber" : 16,
"state" : "PROVIDER_FAILED",
  "description" : "A simple vApp provisioning scenario.",
  "reasons" : "Requesting a vApp.",
  "requestedFor" : "Auto.admin@abx.local",
  "requestedBy" : "Auto.admin@abx.local",
  "organization" : {
    "tenantRef" : "abx",
    "tenantLabel" : "ABX",
    "subtenantRef" : "43a2f89a-c04e-4941-abc5-b4dc68a2810d",
    "subtenantLabel" : "vCD business group"
  },
  "requestorEntitlementId" : "3391b550-fd41-413a-8b45-5ae94e34f36a",
  "preApprovalId" : null,
  "postApprovalId" : null,
  "dateCreated" : "2014-08-11T23:58:06.445Z",
  "lastUpdated" : "2014-08-11T23:59:30.151Z",
  "dateSubmitted" : "2014-08-11T23:58:06.445Z",
  "dateApproved" : null,
  "dateCompleted" : null,
  "quote" : {
    "leasePeriod" : {
      "type" : "timeSpan",
      "unit" : "DAYS",
      "amount" : 2
    },
    "leaseRate" : {
      "type" : "moneyTimeRate",
      "cost" : {
        "type" : "money",
        "currencyCode" : null,
```

```
        "amount" : 4.0
      },
      "basis" : {
        "type" : "timeSpan",
        "unit" : "DAYS",
        "amount" : 1
      }
    },
    "totalLeaseCost" : {
      "type" : "money",
      "currencyCode" : null,
      "amount" : 8.0
    }
  },
  "requestCompletion" : {
    "requestCompletionState" : "FAILED",
    "completionDetails" : "Request failed: Machine vcd4: an error occurred while creating the
virtual machine.."
  },
  "requestData" : {
    "entries" : [ {
      "key" : "provider-ASCT-1.VirtualMachine.Memory.Size",
      "value" : {
        "type" : "string",
        "value" : "1"
      }
    }, {
      "key" : "provider-blueprintId",
      "value" : {
        "type" : "string",
        "value" : "46548940-eb20-4368-9e73-c1685cda8c64"
      }
    }, {
      "key" : "provider-ASCT-1.VirtualMachine.Disk0.Letter",
      "value" : {
        "type" : "string",
        "value" : "c"
      }
    }, {
      "key" : "provider-__requested_allocation_type",
      "value" : {
        "type" : "string",
        "value" : "2"
      }
    }, {
      "key" : "provider-ASCT-1.__Notes",
      "value" : {
        "type" : "string",
        "value" : ""
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.MaxCost",
      "value" : {
        "type" : "string",
        "value" : "4.0000000000"
```

```
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.TotalStorageSize",
      "value" : {
        "type" : "decimal",
        "value" : 0.0
      }
    }, {
      "key" : "provider-provisioningGroupId",
      "value" : {
        "type" : "string",
        "value" : "43a2f89a-c04e-4941-abc5-b4dc68a2810d"
      }
    }, {
      "key" : "provider-__MultiMachine.Provision.NumberOfInstances",
      "value" : {
        "type" : "string",
        "value" : "<ArrayOfKeyValueOfintint xmlns:i=\\\"http://www.w3.org/2001/XMLSchema-
instance\\\" xmlns=\\\"http://schemas.microsoft.com/2003/10/Serialization/Arrays\\\">\r\n
                                                <KeyValueOfintint>\r\n    <Key>1</Key>\r\n
<Value>1</Value>\r\n
</KeyValueOfintint>\r\n</ArrayOfKeyValueOfintint>"
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.AssignToUser",
      "value" : {
        "type" : "string",
        "value" : "Auto.admin@sqa.local"
      }
    }, {
      "key" : "provider-VirtualMachine.LeaseDays",
      "value" : {
        "type" : "integer",
        "value" : 2
      }
    }, {
      "key" : "provider-ASCT-1.VirtualMachine.Disk0.Size",
      "value" : {
        "type" : "string",
        "value" : "1"
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.ProvisionInto",
      "value" : {
        "type" : "string",
        "value" : "2"
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.Description",
      "value" : {
        "type" : "string",
        "value" : "A simple vApp provisioning scenario."
      }
    }, {
      "key" : "provider-ASCT-1.VirtualMachine.CPU.Count",
```

```
      "value" : {
        "type" : "string",
        "value" : "1"
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.Reason",
      "value" : {
        "type" : "string",
        "value" : "Requesting a vApp."
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.NumberOfInstances",
      "value" : {
        "type" : "integer",
        "value" : 1
      }
    }, {
      "key" : "provider-__Notes",
      "value" : {
        "type" : "string",
        "value" : "A simple vApp provisioning scenario."
      }
    }, {
      "key" : "provider-Cafe.Shim.VirtualMachine.MinCost",
      "value" : {
        "type" : "string",
        "value" : "4.0000000000"
      }
    } ]
  },
  "retriesRemaining" : 3,
  "phase" : "FAILED",
  "executionStatus" : "STOPPED",
  "waitingStatus" : "NOT_WAITING",
  "approvalStatus" : "PRE_APPROVED",
  "catalogItemRef" : {
    "id" : "c2cacf7c-b3c8-47fb-a938-2c09910b6713",
    "label" : "vApp"
  }
}
```


Requesting an Amazon EC2 Machine

You can request an Amazon EC2 machine using the REST API.

The checklist provides the tasks required to request a machine with the REST API. Perform the tasks in sequence.

Table 3-5. Requesting an Amazon EC2 Machine Checklist

Task	Details	Permissions
❑ Request an HTTP bearer token	See Chapter 2, "REST API Authentication," on page 9.	
❑ Find the published EC2 blueprint for your request	See "Find the Published Amazon EC2 Blueprint for Your Request," on page 78.	consumer and current business group member
❑ Request an EC2 machine with the published EC2 blueprint using its defaut values	See "Create a JSON File for an Amazon Machine Request," on page 81.	consumer and current business group member
❑ Check your EC2 request status	See "View the Details of an Amazon Machine Request," on page 87.	consumer and current business group member
❑ Request an EC2 machine with the published EC2 blueprint by overriding the default values	See "Request an Amazon Machine and Override Default Values," on page 90.	consumer and current business group member

Find the Published Amazon EC2 Blueprint for Your Request

You can locate the published Amazon EC2 blueprint to use for your machine request by displaying your entitled catalog items in the service catalog.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Generate a list of catalog items from which to obtain the Amazon EC2 blueprint ID. "List Shared and Private Catalog Items," on page 34.

Process Overview

1. From the list of your entitled catalog items, find the catalog item that corresponds to the Amazon EC2 blueprint to use for the request. You can search on the catalog item ID `Infrastructure.Cloud` to locate a published Amazon EC2 blueprint.
2. In the catalog item output, locate the following values which are required for the machine request:
   - catalog item ID
   - tenant ref
   - subtenant ref (business group)
   - binding ID (blueprint)

If you are unsure which blueprint to use, contact your vRealize Automation fabric administrator.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/catalogItems?limit=10&page=1
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	An array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service determines the other names.
href	The URL which produces the result.
Content	An array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following properties:
@type	CatalogItem
Id	The unique tenant identifier
version	
name	The name of the tenant for display purposes
description	Bief description of the tenant
status	Life cycle stage of this catalog item
organization	Subtenant and/or tenant to which this item belongs
tenantRef	ID of tenant
tenantLabel	Name of tenant
subtenantRef	ID of business group
subtenantLabel	Name of business group
providerBinding	Provider side identifier of this item
bindingId	binding ID
providerRef	Provider
forms	A specification for the various forms associated with catalog items of this type
callbacks	A specification for the various call-backs to the provider supported by this catalog item
isNoteworthy	Flag indicating that this catalog item should be highlighted to users for a period of time
dateCreated	Date this item was created in Catalog
lastUpdatedDate	Date this item was last updated in Catalog
iconId	Associated icon representing this item
catalogItemTypeRef	Type of the catalog item
serviceRef	Catalog Service containing this catalog item
outputResourceTypeRef	Type of the resource resulting from requesting this catalog item
Metadata	The paging-related data

Property	Description
Size	Maximum number of rows per page
totalElements	Number of rows returned
totalPages	Total number of pages of data available
Number	Current page number
Offset	Number of rows skipped

Example: `curl` Command

You can use the following command to display all the catalog items you have permission to view.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/catalogItems?limit=10&page
```

Example: API Explorer

You can use the following command to display all the catalog items you have permission to view.

```
vcac-cli>rest get --service catalog-service --u consumer/catalogItems?limit=10&page=1
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
{
  "links" : [ ],
  "content" : [{
    "@type" : "CatalogItem",
    "id" : "6cca9fd9-83b7-4f5d-8884-fb8a005fc656",
    "version" : 1,
    "name" : "EC2 Blueprint",
    "description" : "EC2 blueprint for AMI: amzn-ami-pv-2013.09.2.x86_64-ebs",
    "status" : "PUBLISHED",
    "organization" : {
      "tenantRef" : "sqa",
      "tenantLabel" : "SQA",
      "subtenantRef" : "b475039a-94dd-4bf3-97f6-8596f8cf8818",
      "subtenantLabel" : "Business Group"
    },
    "providerBinding" : {
      "bindingId" : "1701645d-7e43-479f-930c-fbef58d13d50",
      "providerRef" : {
        "id" : "ba3b18dd-a891-48d2-a3e7-faed239990ed",
        "label" : "iaas-service"
      }
    },
    "forms" : null,
    "callbacks" : null,
    "isNoteworthy" : false,
    "dateCreated" : "2014-09-11T18:53:44.474Z",
    "lastUpdatedDate" : "2014-09-11T18:55:11.957Z",
    "iconId" : "cafe_default_icon_genericCatalogItem",
    "catalogItemTypeRef" : {
      "id" : "Infrastructure.Cloud",
      "label" : "Cloud Machine"
```

```
    },
    "serviceRef" : {
      "id" : "5d4ce014-1ee5-41fa-aecd-ec8734f5317a",
      "label" : "CLI Service"
    },
    "outputResourceTypeRef" : {
      "id" : "Infrastructure.Cloud",
      "label" : "Cloud Machine"
    }
  } ],
  "metadata" : {
    "size" : 10,
    "totalElements" : 1,
    "totalPages" : 1,
    "number" : 1,
    "offset" : 0
  }
}
```

Create a JSON File for an Amazon Machine Request

You can request a machine by using a published Amazon EC2 blueprint, the resource values specified in the blueprint, and a JSON input file containing request data such as your user name and business group ID.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- "Find the Published Amazon EC2 Blueprint for Your Request," on page 78.

For information about overriding the default values, see "Request an Amazon Machine and Override Default Values," on page 90.

Process Overview

You can create a JSON file and include that JSON file as part of your command line input.

Create a JSON file to be used in a machine request.

1 Copy the JSON input file template to a new text file.

2 Substitute values for the input variables in the template.

3 Save the file with a `.json` extension.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/requests
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

JSON Template File Parameters

The following table describes the IDs, machine resources, and other information that you must add to your JSON file to create the JSON input parameters you need to submit the machine request.

Value	Description
catalog_item_ID	Specifies the value of CatalogItem ID in the machine blueprint catalog item.
tenant_name	Specifies the value of tenantRef in the machine blueprint catalog item.
business_group_ID	Specifies the value of subtenantRef in the machine blueprint catalog item.
username@fqdn	Specifies the user name of the consumer and business group manager account and fully qualified domain name.
blueprint_ID	Specifies the value of bindingId in the machine blueprint catalog item.
notes	Specifies notes that help to describe the request.
description	Contains a description of the request.
reasons	Contains a general reason for the request.
provider_reason	Contains a general provider reason for the request.
Amazon_ins_type	Specifies an Amazon instance type. Request only Amazon instance types that are supported by your blueprint. If necessary, consult your fabric administrator for details on what your blueprint supports. For information about Amazon instance types, see Amazon EC2 product documentation.

```
{
    "@type": "CatalogItemRequest",
    "catalogItemRef": {
        "id": "catalog_item_ID"
    },
    "organization": {
        "tenantRef": "tenant_name",
        "subtenantRef": "business_group_ID"
    },
    "requestedFor": "username@fqdn",
    "state": "SUBMITTED",
    "requestData": {
        "entries": [{
            "key": "provider-blueprintId",
            "value": {
                "type": "string",
                "value": "blueprint_ID"
            }
        },
        {
            "key": "provider-provisioningGroupId",
            "value": {
                "type": "string",
                "value": "business_group_ID"
            }
        },
```

```
        {
            "key": "requestedFor",
            "value": {
                "type": "string",
                "value": "username@fqdn"
            }
        },
        {
            "key": "provider-__Notes",
            "value": {
                "type": "string",
                "value": "notes"
            }
        },
        {
            "key": "description",
            "value": {
                "type": "string",
                "value": "description"
            }
        },
        {
            "key": "reasons",
            "value": {
                "type": "string",
                "value": "reasons"
            }
        },
        {
            "key": "provider-Cafe.Shim.VirtualMachine.Reason",
            "value": {
                "type": "string",
                "value": "provider_reason"
            }
        },
        {
            "key": "provider-__amazon.instanceType",
            "value": {
                "type": "string",
                "value": "Amazon_ins_type"
            }
        }]
    }
}
```

Example: JSON Input File

You can use the following JSON input file sample when constructing a file for your own requirements.

You should populate all the highlighted value equivalents from the following example JSON file when you create your own JSON input file.

```
{
    "@type": "CatalogItemRequest",
    "catalogItemRef": {
        "id": "6cca9fd9-83b7-4f5d-8884-fb8a005fc656"
    },
```

```
"organization": {
    "tenantRef": "abx",
    "subtenantRef": "b475039a-94dd-4bf3-97f6-8596f8cf8818"
},
"requestedFor": "Auto.admin@abx.local",
"state": "SUBMITTED",
"requestData": {
    "entries": [{
        "key": "provider-blueprintId",
        "value": {
            "type": "string",
            "value": "1701645d-7e43-479f-930c-fbef58d13d50"
        }
    },
    {
        "key": "provider-provisioningGroupId",
        "value": {
            "type": "string",
            "value": " b475039a-94dd-4bf3-97f6-8596f8cf8818"
        }
    },
    {
        "key": "requestedFor",
        "value": {
            "type": "string",
            "value": "Auto.admin@abx.local"
        }
    },
    {
        "key": "provider-__Notes",
        "value": {
            "type": "string",
            "value": "CLI EC2 description"
        }
    },
    {
        "key": "description",
        "value": {
            "type": "string",
            "value": "CLI EC2 description"
        }
    },
    {
        "key": "reasons",
        "value": {
            "type": "string",
            "value": "CLI EC2 reason"
        }
    },
    {
        "key": "provider-Cafe.Shim.VirtualMachine.Reason",
        "value": {
            "type": "string",
            "value": "CLI EC2 reason"
        }
```

```
        },
        {
            "key": "provider-__amazon.instanceType",
            "value": {
                "type": "string",
                "value": "t1.micro"
            }
        }]
    }
}
```

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
version	
state	Specifies the item state, such as SUBMITTED.
approvalStatus	Specifies a status indicating whether this request has been approved, rejected, or is still pending some form of approval.
waitingStatus	Specifies a status indicating whether this request is waiting on any external users or services before it is able to progress.
requestNumber	Specifies a more user-friendly identifier for this request.
executionStatus	Specifies the current execution status of the request.
stateName	Specifies the localized state name.
phase	Specifies the current phase of the request, which is more coarse grained and easier for users to understand.
id	Specifies the universally unique identifier of this object.
iconId	Retrieves icon of this request based on the type of the object requested.
description	Contains a brief description of this request.
reasons	Specifies the business reasons entered by the requestor or owner of this request.
requestedFor	Specifies the ID of the user for whom this request is logged.
requestedBy	Specifies the ID of the user who actually submitted the request
organization	Subtenant and/or tenant owner of this request.
requestorEntitlementId	Specified the value of the requestorEntitlement setting.
preApprovalId	Specifies the ID of the preApproval setting.
postApprovalId	Specifies the ID of the approval generated for the post-provisioning workflow step.
dateCreated	Specifies the date when this request was sent to the catalog.
lastUpdated	Specifies the date when this request was last updated.
dateSubmitted	Specifies the date when this request was first submitted.
dateApproved	Specifies the date when this request was approved.
dateCompleted	Specifies the date when this request was completed.
quote	Contains a quote made by the provider defining the estimated cost(s) associated with the request and/or any resources provisioned as a result of the request.
requestCompletion	Contains additional request completion information.
requestData	Contains a map of the provider-specific field-value pairs collected for this request.

Property	Description
retriesRemaning	Specifies the number of attempts remaining to move this request from its current state to the next state in the request workflow. Some state transitions require calls to external services. These calls may fail due to transient errors such as momentary network errors. In these cases, the catalog will retry the call a number of times before failing. This property defines the number of retries remaining for the current state transition. When it reaches 0, the catalog will stop retrying and mark the request as failed. This property is reset to the default number of retries for every new operation that is triggered.
requestedItemName	Specifies the item name.
requestedItemDescription	Specifies the item description.

Example: `curl` Command

You can use the following command to submit a machine request that includes the specifications in an `EC2request.json` input file.

```
curl --insecure -H "Content-Type: application/json" -H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/requests --data @EC2request.json
```

Example: API Explorer

You can use the following command to submit a machine request that includes the specifications in an `EC2request.json` input file.

```
rest post --headers --service catalog-service --uri consumer/requests --data @EC2request.json
```

Example: JSON Output

The following JSON output is returned based on your command input.

The highlighted URL in the following sample indicates the location and ID of the vRealize Automation request.

```
Request Headers
{
             Accept = application/json
       Content-Type = application/json
     Content-Length = 1347
     Accept-Charset = big5, big5-hkscs, ...
}
Response Headers
{
               Date = Tue, 11 Nov 2014 22:28:35 GMT
               ETag = "0"
           Location =
https://abx148-084-124.eng.mycompany.com/catalog-service/api/consumer/requests/25211c6c-
f09d-4e2b-
                      9be4-7b09c47c9f6c
       Content-Type = application/json;charset=UTF-8
     Content-Length = 0
               Vary = Accept-Encoding,User-Agent
         Keep-Alive = timeout=15, max=100
         Connection = Keep-Alive
}
null
```

View the Details of an Amazon Machine Request

You can check the status of your Amazon EC2 Machine request by using the catalog service.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- "Create a JSON File for an Amazon Machine Request," on page 81 or "Request an Amazon Machine and Override Default Values," on page 90.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/requests/*requestId*
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
requestId	Specifies the ID of the request to check.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
version	
state	Specifies the item state, such as SUBMITTED.
approvalStatus	Specifies a status indicating whether this request has been approved, rejected, or is still pending some form of approval.
waitingStatus	Specifies a status indicating whether this request is waiting on any external users or services before it is able to progress.
requestNumber	Specifies a more user-friendly identifier for this request.
executionStatus	Specifies the current execution status of the request.
stateName	Specifies the localized state name.
phase	Specifies the current phase of the request, which is more coarse grained and easier for users to understand.
id	Specifies the universally unique identifier of this object.
iconId	Retrieves icon of this request based on the type of the object requested.
description	Contains a brief description of this request.
reasons	Specifies the business reasons entered by the requestor or owner of this request.
requestedFor	Specifies the ID of the user for whom this request is logged.
requestedBy	Specifies the ID of the user who actually submitted the request
organization	Subtenant and/or tenant owner of this request.

Property	Description
requestorEntitlementId	Specified the value of the requestorEntitlement setting.
preApprovalId	Specifies the ID of the preApproval setting.
postApprovalId	Specifies the ID of the approval generated for the post-provisioning workflow step.
dateCreated	Specifies the date when this request was sent to the catalog.
lastUpdated	Specifies the date when this request was last updated.
dateSubmitted	Specifies the date when this request was first submitted.
dateApproved	Specifies the date when this request was approved.
dateCompleted	Specifies the date when this request was completed.
quote	Contains a quote made by the provider defining the estimated cost(s) associated with the request and/or any resources provisioned as a result of the request.
requestCompletion	Contains additional request completion information.
requestData	Contains a map of the provider-specific field-value pairs collected for this request.
retriesRemaning	Specifies the number of attempts remaining to move this request from its current state to the next state in the request workflow. Some state transitions require calls to external services. These calls may fail due to transient errors such as momentary network errors. In these cases, the catalog will retry the call a number of times before failing. This property defines the number of retries remaining for the current state transition. When it reaches 0, the catalog will stop retrying and mark the request as failed. This property is reset to the default number of retries for every new operation that is triggered.
requestedItemName	Specifies the item name.
requestedItemDescription	Specifies the item description.

Example: `curl` Command

You can use the following command to check on the status of your Amazon machine request.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/requests/25211c6c-f09d-4e2b-9be4-7b09c47c9f6c
```

Example: API Explorer

You can use the following command to check on the status of your Amazon machine request.

```
rest get --service catalog-service --uri consumer/requests/25211c6c-f09d-4e2b-9be4-7b09c47c9f6c
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
{
  "@type" : "CatalogItemRequest",
  "id" : "25211c6c-f09d-4e2b-9be4-7b09c47c9f6c",
  "iconId" : "cafe_default_icon_genericCatalogItem",
  "version" : 5,
  "requestNumber" : 14,
  "state" : "SUCCESSFUL",
  "description" : "CLI EC2 description",
  "reasons" : "CLI EC2 reason",
  "requestedFor" : "Auto.admin@abx.local",
  "requestedBy" : "Auto.admin@abx.local",
```

```
"organization" : {
  "tenantRef" : "sqa",
  "tenantLabel" : "SQA",
  "subtenantRef" : "b475039a-94dd-4bf3-97f6-8596f8cf8818",
  "subtenantLabel" : "Business Group"
},
"requestorEntitlementId" : "04f4588f-548a-4bc6-baf8-c22241918322",
"preApprovalId" : null,
"postApprovalId" : null,
"dateCreated" : "2014-09-11T22:29:02.190Z",
"lastUpdated" : "2014-09-11T22:31:05.780Z",
"dateSubmitted" : "2014-09-11T22:29:02.190Z",
"dateApproved" : null,
"dateCompleted" : "2014-09-11T22:31:05.779Z",
"quote" : {
  "leaseRate" : {
    "type" : "moneyTimeRate",
    "cost" : {
      "type" : "money",
      "currencyCode" : null,
      "amount" : 3.0
    },
    "basis" : {
      "type" : "timeSpan",
      "unit" : "DAYS",
      "amount" : 1
    }
  }
},
"requestCompletion" : {
  "requestCompletionState" : "SUCCESSFUL",
  "completionDetails" : "Request succeeded. Created mp108."
},
"requestData" : {
  "entries" : [ {
    "key" : "provider-blueprintId",
    "value" : {
      "type" : "string",
      "value" : "1701645d-7e43-479f-930c-fbef58d13d50"
    }
  }, {
    "key" : "provider-provisioningGroupId",
    "value" : {
      "type" : "string",
      "value" : "b475039a-94dd-4bf3-97f6-8596f8cf8818"
    }
  }, {
    "key" : "provider-__Notes",
    "value" : {
      "type" : "string",
      "value" : "CLI EC2 description"
    }
  }, {
    "key" : "provider-Cafe.Shim.VirtualMachine.Reason",
    "value" : {
```

```
        "type" : "string",
        "value" : "CLI EC2 reason"
      }
    }, {
      "key" : "provider-__amazon.instanceType",
      "value" : {
        "type" : "string",
        "value" : "t1.micro"
      }
    } ]
  },
  "retriesRemaining" : 3,
  "phase" : "SUCCESSFUL",
  "executionStatus" : "STOPPED",
  "waitingStatus" : "NOT_WAITING",
  "approvalStatus" : "POST_APPROVED",
  "catalogItemRef" : {
    "id" : "6cca9fd9-83b7-4f5d-8884-fb8a005fc656",
    "label" : "EC2 Blueprint"
  }
}
```

Request an Amazon Machine and Override Default Values

You can override the default values of an Amazon EC2 blueprint by adding properties to the default JSON input file.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/requests/*requestId*
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

JSON Template

Use following JSON template sample to create a JSON input file.

```
{
    "@type": "CatalogItemRequest",
    "catalogItemRef": {
        "id": "catalog_item_ID"
    },
    "organization": {
        "tenantRef": "tenant_name,
```

```
    "subtenantRef": "business_group_ID"
},
"requestedFor": "username@fqdn",
"state": "SUBMITTED",
"requestData": {
    "entries": [{
        "key": "provider-blueprintId",
        "value": {
            "type": "string",
            "value": "blueprint_ID"
        }
    },
    {
        "key": "provider-provisioningGroupId",
        "value": {
            "type": "string",
            "value": "business_group_ID"
        }
    },
    {
        "key": "requestedFor",
        "value": {
            "type": "string",
            "value": "username@fqdn"
        }
    },
    {
        "key": "provider-__Notes",
        "value": {
            "type": "string",
            "value": "notes"
        }
    },
    {
        "key": "description",
        "value": {
            "type": "string",
            "value": "description"
        }
    },
    {
        "key": "reasons",
        "value": {
            "type": "string",
            "value": "reasons"
        }
    },
    {
        "key": "provider-Cafe.Shim.VirtualMachine.Reason",
        "value": {
            "type": "string",
            "value": "provider_reason"
        }
    },
    {
```

```
            "key": "provider-__amazon.instanceType",
            "value": {
                "type": "string",
                "value": "Amazon_ins_type"
            }
},
        {
            "key": "key_name1",
            "value": {
                "type": "type1",
                "value": "key_value1"
            }
        },
        {
            "key": "key_namen",
            "value": {
                "type": "typen",
                "value": "key_valuen"
            }
        }]
    }
}
```

The following table describes the IDs, machine resources, and other information that you must add to your JSON file to create the JSON input parameters you need to submit the machine request.

You should populate all the highlighted value equivalents from the following example JSON file when you create your own JSON input file.

Value	Description
catalog_item_ID	Specifies the value of CatalogItem ID in the machine blueprint catalog item.
tenant_name	Specifies the value of tenantRef in the machine blueprint catalog item.
business_group_ID	Specifies the value of subtenantRef in the machine blueprint catalog item.
username@fqdn	Specifies the user name of the consumer and business group manager account and fully qualified domain name.
blueprint_ID	Specifies the value of bindingId in the machine blueprint catalog item.
notes	Specifies notes that help to describe the request.
description	Contains a description of the request.
reasons	Contains a general reason for the request.
provider_reason	Contains a general provider reason for the request.
Amazon_ins_type	Specifies an Amazon instance type. Request only Amazon instance types that are supported by your blueprint. If necessary, consult your fabric administrator for details on what your blueprint supports. For information about Amazon instance types, see Amazon EC2 product documentation.

Example: JSON Input File

The following example requests a small Amazon instance type, which overrides the default location to us-west-1a. It also creates an EBS storage volume named Backup and mounts it to `/dev/sdf`.

```
{
    "@type": "CatalogItemRequest",
    "catalogItemRef": {
        "id": "6cca9fd9-83b7-4f5d-8884-fb8a005fc656"
    },
    "organization": {
        "tenantRef": "sqa",
        "subtenantRef": "b475039a-94dd-4bf3-97f6-8596f8cf8818"
    },
    "requestedFor": "Auto.admin@sqa.local",
    "state": "SUBMITTED",
    "requestData": {
        "entries": [{
            "key": "provider-blueprintId",
            "value": {
                "type": "string",
                "value": "1701645d-7e43-479f-930c-fbef58d13d50"
            }
        },
        {
            "key": "provider-provisioningGroupId",
            "value": {
                "type": "string",
                "value": "b475039a-94dd-4bf3-97f6-8596f8cf8818"
            }
        },
        {
            "key": "requestedFor",
            "value": {
                "type": "string",
                "value": "Auto.admin@sqa.local"
            }
        },
        {
            "key": "provider-__Notes",
            "value": {
                "type": "string",
                "value": "Adding extra EBS storage"
            }
        },
        {
            "key": "description",
            "value": {
                "type": "string",
                "value": "Adding extra EBS storage"
            }
        },
        {
            "key": "reasons",
            "value": {
                "type": "string",
```

```
                "value": "Writing documentation"
            }
        },
        {
            "key": "provider-Cafe.Shim.VirtualMachine.Reason",
            "value": {
                "type": "string",
                "value": "Writing documentation"
            }
        },
        {
            "key": "provider-__amazon.instanceType",
            "value": {
                "type": "string",
                "value": "m1.small"
            }
        },
        {
            "key": "provider-__amazon.volumeAction0",
            "value": {
                "type": "string",
                "value": "<StorageVolume xmlns:i=\"http://www.w3.org/2001/XMLSchema-instance\"
xmlns=\"http://schemas.datacontract.org/2004/07/DynamicOps.AmazonWSModel\">\r\n
<_x003C_Description_x003E_k__BackingField>Extra EBS for
backup</_x003C_Description_x003E_k__BackingField>\r\n
<_x003C_Device_x003E_k__BackingField>/dev/sdf</_x003C_Device_x003E_k__BackingField>\r\n
<_x003C_Id_x003E_k__BackingField>0</_x003C_Id_x003E_k__BackingField>\r\n
<_x003C_Location_x003E_k__BackingField i:nil=\"true\" />\r\n
<_x003C_Name_x003E_k__BackingField>Backup</_x003C_Name_x003E_k__BackingField>\r\n
<_x003C_Owner_x003E_k__BackingField>Auto.admin@sqa.local</_x003C_Owner_x003E_k__BackingField>\r\n
  <_x003C_Size_x003E_k__BackingField>2</_x003C_Size_x003E_k__BackingField>\r\n
<_x003C_ToBeAttached_x003E_k__BackingField>true</_x003C_ToBeAttached_x003E_k__BackingField>\r\n
<_x003C_ToBeDeleted_x003E_k__BackingField>false</_x003C_ToBeDeleted_x003E_k__BackingField>\r\n
<_x003C_VolumeId_x003E_k__BackingField i:nil=\"true\" />\r\n</StorageVolume>"
            }
        },
        {
            "key": "provider-Vrm.DataCenter.Location",
            "value": {
                "type": "string",
                "value": "us-west-1a"
            }
        }]
    }
}
```


Output

The command output contains property names and values based on the API command input parameters.

Property	Description
version	
state	Specifies the item state, such as SUBMITTED.
approvalStatus	Specifies a status indicating whether this request has been approved, rejected, or is still pending some form of approval.

Property	Description
waitingStatus	Specifies a status indicating whether this request is waiting on any external users or services before it is able to progress.
requestNumber	Specifies a more user-friendly identifier for this request.
executionStatus	Specifies the current execution status of the request.
stateName	Specifies the localized state name.
phase	Specifies the current phase of the request, which is more coarse grained and easier for users to understand.
id	Specifies the universally unique identifier of this object.
iconId	Retrieves icon of this request based on the type of the object requested.
description	Contains a brief description of this request.
reasons	Specifies the business reasons entered by the requestor or owner of this request.
requestedFor	Specifies the ID of the user for whom this request is logged.
requestedBy	Specifies the ID of the user who actually submitted the request
organization	Subtenant and/or tenant owner of this request.
requestorEntitlementId	Specified the value of the requestorEntitlement setting.
preApprovalId	Specifies the ID of the preApproval setting.
postApprovalId	Specifies the ID of the approval generated for the post-provisioning workflow step.
dateCreated	Specifies the date when this request was sent to the catalog.
lastUpdated	Specifies the date when this request was last updated.
dateSubmitted	Specifies the date when this request was first submitted.
dateApproved	Specifies the date when this request was approved.
dateCompleted	Specifies the date when this request was completed.
quote	Contains a quote made by the provider defining the estimated cost(s) associated with the request and/or any resources provisioned as a result of the request.
requestCompletion	Contains additional request completion information.
requestData	Contains a map of the provider-specific field-value pairs collected for this request.
retriesRemaning	Specifies the number of attempts remaining to move this request from its current state to the next state in the request workflow. Some state transitions require calls to external services. These calls may fail due to transient errors such as momentary network errors. In these cases, the catalog will retry the call a number of times before failing. This property defines the number of retries remaining for the current state transition. When it reaches 0, the catalog will stop retrying and mark the request as failed. This property is reset to the default number of retries for every new operation that is triggered.
requestedItemName	Specifies the item name.
requestedItemDescription	Specifies the item description.

Example: `curl` Command

You can use the following command to submit a request that includes the specifications in an `ec2machine_specific.json` input file.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/requests --data @ec2machine_specific.json
```

Example: API Explorer

You can use the following command to submit a request that includes the specifications in an `ec2machine_specific.json` input file.

```
rest post --headers --service catalog-service --uri consumer/requests --data
@ec2machine_specific.json
```

Example: JSON Output

The following JSON output is returned based on your command input.

The highlighted URL in the following sample indicates the location and ID of the vRealize Automation request.

```
Request Headers
{
             Accept = application/json
       Content-Type = application/json
     Content-Length = 1347
     Accept-Charset = big5, big5-hkscs, ...
}
Response Headers
{
               Date = Tue, 11 Oct 2014 22:28:35 GMT
               ETag = "0"
           Location =
https://abx148-084-124.eng.mycompany.com/catalog-service/api/consumer/requests/25211c6c-
f09d-4e2b-
                      9be4-7b09c47c9f6c
       Content-Type = application/json;charset=UTF-8
     Content-Length = 0
               Vary = Accept-Encoding,User-Agent
         Keep-Alive = timeout=15, max=100
         Connection = Keep-Alive
}
null
```

Approving a Machine Request

You can approve a machine request using the REST API.

The checklist provides the tasks required to request a machine with the REST API. Perform the tasks in sequence.

Table 3-6. Approving a Machine Request Checklist

Task	Details	Permissions
❑ Request an HTTP bearer token	See Chapter 2, "REST API Authentication," on page 9.	
❑ List the work items	See "List Work Items," on page 97.	consumer and current business group member

Table 3-6. Approving a Machine Request Checklist (Continued)

Task	Details	Permissions
❑ Get the work item details	See "Get Work Item Details," on page 103.	consumer and current business group member
❑ Approve the request by completing the work item	See "Approve the Request by Completing the Work Item," on page 108.	consumer and current business group member

List Work Items

You can list the unique IDs of all work items by using the work item service.

Prerequisites

- Log in to vRealize Automation as an approver with at least one of the following qualifications:
  - You are designated as an approver in an approval policy.
  - You belong to a group which has been designated as an approval group in an approval policy.
  - You are designated as a delegate for someone who is an approver.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Inputs

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/workitem-service/api/workitems
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Specifies an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL which produces the result.
work itemNumber	Displays a reference number for the work item.
id	Displays the universally unique ID of the entity.
version	Displays the object version number, supports optimistic concurrency.
assignees	Displays the list of work item assignees.

Property	Description
subTenantId	Optionally associates the work item with a specific business group granting users with management responsibilities over that business group permission to see the approval.
tenantId	Specifies the tenant ID for the work item.
callbackEntityId	Specifies the callback entity ID for the work item.
work itemType	Specifies the work item type for the work item.
completedDate	Specifies the date when the work item was completed.
assignedDate	Specifies the date when the work item was assigned.
createdDate	Specifies the created date of this instance.
assignedOrCompletedDate	Specifies the date to be displayed on UI.
formUrl	Specifies the URL from which the layout for this work item can be retrieved.
serviceId	Specifies the service ID that generated this work item instance.
work itemRequest	Specifies the corresponding work item request object.
status	Specifies the status of the work item.
completedBy	Specifies the principal ID of user who completed the work item.
availableActions	Contains a list of relevant work item actions.
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: `curl` Command

You can use the following command to retrieve all your available work item IDs.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/workitem-service/api/workitems
```

Example: API Explorer

You can use the following command to retrieve all your available work item IDs.

```
rest get --service workitem-service --uri workitems
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
{
  "links" : [ ],
  "content" : [ {
    "@type" : "WorkItem",
    "id" : "1755ef1a-d6f0-4901-9ecd-d03352ae4a05",
    "version" : 1,
    "workItemNumber" : 1,
    "assignees" : [ {
      "principalId" : "tony@example.mycompany.com",
```

```
  "principalType" : "USER"
} ],
"tenantId" : "MYCOMPANY",
"callbackEntityId" : "1",
"workItemType" : {
  "id" : "com.mycompany.cafe.samples.travel.workItem",
  "name" : "Workspace Assignment",
  "pluralizedName" : "Workspace Assignments",
  "description" : "Location Specific Workspace Assignment",
  "serviceTypeId" : "com.mycompany.cafe.samples.travel.api",
  "actions" : [ {
    "id" : "com.mycompany.cafe.samples.travel.workItem.complete",
    "name" : "Reserve Workspace",
    "stateName" : "Completed",
    "icon" : {
      "id" : "baa623db-0ca0-4db7-af41-9a301bc9e152",
      "name" : "Complete Action Icon",
      "contentType" : "image/png",
      "image" : null
    }
  }, {
    "id" : "com.mycompany.cafe.samples.travel.workItem.cancel",
    "name" : "Workspace Unavailable",
    "stateName" : "Cancelled",
    "icon" : {
      "id" : "b03f994a-e1ec-4aae-8fae-e747ed680a5e",
      "name" : "Cancel Action Icon",
      "contentType" : "image/png",
      "image" : null
    }
  } ],
  "completeByEmail" : true,
  "commentsField" : null,
  "listView" : {
    "columns" : [ {
      "id" : "duration",
      "label" : "Duration",
      "description" : "The length of stay, measured in days.",
      "dataType" : {
        "type" : "primitive",
        "typeId" : "INTEGER"
      },
      "displayAdvice" : null,
      "state" : {
        "dependencies" : [ ],
        "facets" : [ ]
      },
      "filterable" : false,
      "sortable" : false,
      "isMultiValued" : false
    }, {
      "id" : "location",
      "label" : "Destination",
      "description" : "The destination to which travel is being requested.",
      "dataType" : {
```

```
          "type" : "ref",
          "componentTypeId" : null,
          "componentId" : null,
          "classId" : "location",
          "typeFilter" : null,
          "label" : null
        },
        "displayAdvice" : null,
        "state" : {
          "dependencies" : [ ],
          "facets" : [ ]
        },
        "filterable" : false,
        "sortable" : false,
        "isMultiValued" : false
      }, {
        "id" : "arrivalDate",
        "label" : "Arrival Date",
        "description" : "The date of arrival at the destination",
        "dataType" : {
          "type" : "primitive",
          "typeId" : "DATE_TIME"
        },
        "displayAdvice" : null,
        "state" : {
          "dependencies" : [ ],
          "facets" : [ ]
        },
        "filterable" : false,
        "sortable" : false,
        "isMultiValued" : false
      } ],
      "defaultSequence" : [ "location", "arrivalDate", "duration" ]
    },
    "version" : 3,
    "forms" : {
      "workItemDetails" : {
        "type" : "external",
        "formId" : "travel.seating.task"
      },
      "workItemSubmission" : {
        "type" : "external",
        "formId" : "travel.seating.task"
      },
      "workItemNotification" : {
        "type" : "external",
        "formId" : "travel.itinerary.details"
      }
    }
  },

          .
          .
          .
```

```
"completedDate" : null,
"assignedDate" : "2014-02-20T23:55:31.600Z",
"createdDate" : "2014-02-20T23:55:31.600Z",
"assignedOrCompletedDate" : "2014-02-20T23:55:31.600Z",
"serviceId" : "2af18227-6a00-49e9-a76b-96de3ee767d2",
"workItemRequest" : {
  "itemId" : "531660fd-b540-4946-9917-38c023b61c02",
  "itemName" : "test travel 1",
  "itemDescription" : "test travel 1",
  "itemRequestor" : "tony@example.mycompany.com",
  "itemCost" : 0.0,
  "itemData" : {
    "entries" : [ {
      "key" : "requestLeaseTotal",
      "value" : {
        "type" : "money",
        "currencyCode" : null,
        "amount" : 1065.0
      }
    }, {
      "key" : "approvalId",
      "value" : {
        "type" : "string",
        "value" : "7a8b6054-1922-4f82-9266-245dffaa957c"
      }
    }, {
      "key" : "requestClassId",
      "value" : {
        "type" : "string",
        "value" : "request"
      }
    }, {
      "key" : "requestedFor",
      "value" : {
        "type" : "string",
        "value" : "tony@example.mycompany.com"
      }
    }, {
      "key" : "requestReasons"
    }, {
      "key" : "requestedItemName",
      "value" : {
        "type" : "string",
        "value" : "test travel 1"
      }
    }, {
      "key" : "requestInstanceId",
      "value" : {
        "type" : "string",
        "value" : "1cfe7177-74e3-4d68-a559-ea17587022ca"
      }
    }, {
      "key" : "requestRef",
      "value" : {
```

```
      "type" : "string",
      "value" : "15"
    }
  }, {
    "key" : "requestedItemDescription",
    "value" : {
      "type" : "string",
      "value" : "test travel 1"
    }
  }, {
    "key" : "requestLeaseRate",
    "value" : {
      "type" : "moneyTimeRate",
      "cost" : {
        "type" : "money",
        "currencyCode" : null,
        "amount" : 213.0
      },
      "basis" : {
        "type" : "timeSpan",
        "unit" : "DAYS",
        "amount" : 1
      }
    }
  }, {
    "key" : "requestingServiceId",
    "value" : {
      "type" : "string",
      "value" : "f91d044a-04f9-4b96-8542-375e3e4e1dc1"
    }
  }, {
    "key" : "policy",
    "value" : {
      "type" : "string",
      "value" : "test travel approval policy"
    }
  }, {
    "key" : "phase",
    "value" : {
      "type" : "string",
      "value" : "Pre Approval"
    }
  }, {
    "key" : "requestDescription",
    "value" : {
      "type" : "string",
      "value" : "t"
    }
  }, {
    "key" : "requestLease",
    "value" : {
      "type" : "timeSpan",
      "unit" : "DAYS",
      "amount" : 5
    }
```

```
        }, {
          "key" : "requestedBy",
          "value" : {
            "type" : "string",
            "value" : "tony@example.mycompany.com"
          }
        } ]
      }
    },
    "status" : "Active",
    "availableActions" : [ ]
  } ],
  "metadata" : {
    "size" : 20,
    "totalElements" : 7,
    "totalPages" : 1,
    "number" : 1,
    "offset" : 0
  }
}
```


Get Work Item Details

You can display the details of a pending work item. You need these details to submit a completion request.

Prerequisites

- Log in to vRealize Automation as an approver with at least one of the following qualifications:
  - You are designated as an approver in an approval policy.
  - You belong to a group which has been designated as an approval group in an approval policy.
  - You are designated as a delegate for someone who is an approver.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the necessary work item identifier. See "List Work Items," on page 97.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/workitem-service/api/workitems/*workitem_ID*
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
workitem_ID	Specifies the unique identifier of a work item.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Specifies an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL which produces the result.
work itemNumber	Displays a reference number for the work item.
id	Displays the universally unique ID of the entity.
version	Displays the object version number, supports optimistic concurrency.
assignees	Displays the list of work item assignees.
subTenantId	Optionally associates the work item with a specific business group granting users with management responsibilities over that business group permission to see the approval.
tenantId	Specifies the tenant ID for the work item.
callbackEntityId	Specifies the callback entity ID for the work item.
work itemType	Specifies the work item type for the work item.
completedDate	Specifies the date when the work item was completed.
assignedDate	Specifies the date when the work item was assigned.
createdDate	Specifies the created date of this instance.
assignedOrCompletedDate	Specifies the date to be displayed on UI.
formUrl	Specifies the URL from which the layout for this work item can be retrieved.
serviceId	Specifies the service ID that generated this work item instance.
work itemRequest	Specifies the corresponding work item request object.
status	Specifies the status of the work item.
completedBy	Specifies the principal ID of user who completed the work item.
availableActions	Contains a list of relevant work item actions.
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: `curl` Command

You can use this command to retrieve the necessary details for the specified work item.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/workitem-service/api/workitems/5e3e9519-78ea-4409-a52c-e4aa3bc56511
```

Example: API Explorer

You can use this command to retrieve the necessary details for the specified work item and redirect the JSON output to the `workItemDetails.json` file.

```
vcac-shell>rest get --service workitem-service --u workitems/5e3e9519-78ea-4409-a52c-e4aa3bc56511
2> workItemDetails.json
```

Example: JSON Output

The following JSON output is returned based on your command input.

To view the contents of a JSON output file, for example `workItemDetails.json`, use the `!` command with `more` in UNIX or `type` in Windows.

- (UNIX) vcac-shell>**`! more workItemDetails.json`**
- (Windows) vcac-shell> **`! CMD /C type workItemDetails.json`**


```
vcac-shell>  ! more workItemDetails.json
{
  "id" : "5e3e9519-78ea-4409-a52c-e4aa3bc56511",
  "version" : 0,
  "workItemNumber" : 8,
  "assignees" : [ {
    "principalId" : "tony@example.mycompany.com",
    "principalType" : "USER"
  } ],
  "subTenantId" : "eab762cb-6e75-4379-83ef-171a71c9f00e",
  "tenantId" : "MYCOMPANY",
  "callbackEntityId" : "069dc3ce-a260-4d6a-b191-683141c994c0",
  "workItemType" : {
    "id" : "com.mycompany.csp.core.approval.workitem.request",
    "name" : "Approval",
    "pluralizedName" : "Approvals",
    "description" : "",
    "serviceTypeId" : "com.mycompany.csp.core.cafe.approvals",
    "actions" : [ {
      "id" : "com.mycompany.csp.core.approval.action.approve",
      "name" : "Approve",
      "stateName" : "Approved",
      "icon" : {
        "id" : "c192b6a7-5b35-4a3b-8593-107ffcf8c3a8",
        "name" : "approved.png",
        "contentType" : "image/png",
        "image" : null
      }
    }, {
      "id" : "com.mycompany.csp.core.approval.action.reject",
      "name" : "Reject",
      "stateName" : "Rejected",
      "icon" : {
        "id" : "61c6da67-1164-421d-b575-10a245c89e10",
        "name" : "rejected.png",
        "contentType" : "image/png",
        "image" : null
      }
```

```
    } ],
    "completeByEmail" : true,
    "commentsField" : "businessJustification",
    "listView" : {
      "columns" : [ {
        "id" : "requestedItemName",
        "label" : "Requested Item",
        "description" : "",
        "dataType" : {
          "type" : "primitive",
          "typeId" : "STRING"
        },
        "displayAdvice" : null,
        "state" : {
          "dependencies" : [ ],
          "facets" : [ ]
        },
        "filterable" : false,
        "sortable" : false,
        "isMultiValued" : false
      },

            .
            .
            .

       {
        "id" : "requestLease",
        "label" : "Lease",
        "description" : "",
        "dataType" : {
          "type" : "primitive",
          "typeId" : "TIME_SPAN"
        },
        "displayAdvice" : null,
        "state" : {
          "dependencies" : [ ],
          "facets" : [ ]
        },
        "filterable" : false,
        "sortable" : false,
        "isMultiValued" : false
      } ],
      "defaultSequence" : [ "requestRef", "requestedItemName", "requestedFor", "requestLease",
"requestLeaseRate", "requestLeaseTotal" ]
    },
    "version" : 1,
    "forms" : {
      "workItemDetails" : {
        "type" : "external",
        "formId" : "approval.details"
      },
      "workItemSubmission" : {
        "type" : "external",
        "formId" : "approval.submission"
```

```
    },
    "workItemNotification" : {
      "type" : "external",
      "formId" : "approval.notification"
    }
  }
},
"completedDate" : null,
"assignedDate" : "2014-02-25T01:26:07.153Z",
"createdDate" : "2014-02-25T01:26:07.153Z",
"assignedOrCompletedDate" : "2014-02-25T01:26:07.153Z",
"serviceId" : "2af18227-6a00-49e9-a76b-96de3ee767d2",
"workItemRequest" : {
  "itemId" : "069dc3ce-a260-4d6a-b191-683141c994c0",
  "itemName" : "test-blueprint",
  "itemDescription" : "",
  "itemRequestor" : "fritz@example.mycompany.com",
  "itemCost" : 0.0,
  "itemData" : {
    "entries" : [ {
      "key" : "requestLeaseTotal"
    }, {
      "key" : "approvalId",
      "value" : {
        "type" : "string",
        "value" : "469c11ae-ed27-4790-baf1-c6839f35d474"
      }
    }, {
      "key" : "requestClassId",
      "value" : {
        "type" : "string",
        "value" : "request"
      }
    }, {
      "key" : "requestedFor",
      "value" : {
        "type" : "string",
        "value" : "fritz@example.mycompany.com"
      }
    }, {
      "key" : "requestReasons",
      "value" : {
        "type" : "string",
        "value" : ""
      }
    }, {
      "key" : "requestedItemName",
      "value" : {
        "type" : "string",
        "value" : "test-blueprint"
      }

          .
          .
          .
```

```
      }, {
        "key" : "requestLease"
      }, {
        "key" : "requestedBy",
        "value" : {
          "type" : "string",
          "value" : "fritz@example.mycompany.com"
        }
      } ]
    }
  },
  "status" : "Active",
  "availableActions" : [ ]
}
```

Approve the Request by Completing the Work Item

You can submit a CompleteRequest request to complete a work item and approve the request. To construct the request, add work item and work item form details to a template in a JSON file.

Prerequisites

- Log in to vRealize Automation as an approver. One of the following conditions must apply:
  - You are designated as an approver in an approval policy.
  - You belong to a group which has been designated as an approval group in an approval policy.
  - You are designated as a delegate for someone who is an approver.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the necessary work item detail values to include in the request. See "Get Work Item Details," on page 103.

Process Overview

You can create a JSON file and include that JSON file as part of your command line input.

1 Open a text editor and create a file, for example, `approve.json`.

2 Copy the JSON template in this section to the file.

3 Replace the highlighted text in the template with actual values from the work item details and work item form details.

4 Save the file with any valid file name and file extension, for example, `approve.json`.

5 Submit the CompleteRequest request.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/workitem-service/api/workitems/*workitem_ID*
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
workitem_ID	Specifies the unique identifier of a work item.

Template JSON File Values

You can specify a JSON file in your command line input. Include the information necessary to approve the request and complete the work item. Copy this template to a text file and replace the highlighted values with your obtained values.

```
{
   "formData": {
      "entries": [
         {
            "key": "source-source-provider-Cafe.Shim.VirtualMachine.NumberOfInstances",
            "value": {
               "type": "integer",
               "value": 1
            }
         },
         {
            "key": "source-source-provider-VirtualMachine.Memory.Size",
            "value": {
               "type": "integer",
               "value": 512
            }
         },
         {
            "key": "source-source-provider-VirtualMachine.CPU.Count",
            "value": {
               "type": "integer",
               "value": 1
            }
         },
         {
            "key": "source-businessJustification",
            "value": {
               "type": "string",
               "value": "solves abx request"
            }
         },
         {
            "key": "source-source-provider-VirtualMachine.LeaseDays",
            "value": {
               "type": "integer",
               "value": 0
            }
         }
      ]
```

```
    },
    "workItemId": "5e3e9519-78ea-4409-a52c-e4aa3bc56511",
    "workItemActionId": "com.mycompany.csp.core.approval.action.approve"
}
```

Table 3-7. JSON Template Value Table

JSON File Parameter Name	Description of Value
workItemId	Specifies the value of the corresponding work item ID obtained from the work item list.
source-source-provider-Cafe.Shim.VirtualMachine.NumberOfInstances value	Specifies the number of instances requested.
source-source-provider-VirtualMachine.Memory.Size	Specifies the amount of memory requested in GB.
source-source-provider-VirtualMachine.CPU.Count	Specifies the number of CPUs requested.
source-businessJustification	Specifies the text description of reason for request.
source-source-provider-VirtualMachine.LeaseDays	Specifies the number of days to lease.
workItemActionId	To reject the request, use **`com.mycompany.csp.core.approval.action.reject`**. Also use this in the URL.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Specifies an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL which produces the result.
work itemNumber	Displays a reference number for the work item.
id	Displays the universally unique ID of the entity.
version	Displays the object version number, supports optimistic concurrency.
assignees	Displays the list of work item assignees.
subTenantId	Optionally associates the work item with a specific business group granting users with management responsibilities over that business group permission to see the approval.
tenantId	Specifies the tenant ID for the work item.
callbackEntityId	Specifies the callback entity ID for the work item.
work itemType	Specifies the work item type for the work item.
completedDate	Specifies the date when the work item was completed.
assignedDate	Specifies the date when the work item was assigned.
createdDate	Specifies the created date of this instance.
assignedOrCompletedDate	Specifies the date to be displayed on UI.
formUrl	Specifies the URL from which the layout for this work item can be retrieved.
serviceId	Specifies the service ID that generated this work item instance.

Property	Description
work itemRequest	Specifies the corresponding work item request object.
status	Specifies the status of the work item.
completedBy	Specifies the principal ID of user who completed the work item.
availableActions	Contains a list of relevant work item actions.
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: `curl` Command

You can use the following command to submit the CompleteRequest request as approved.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/workitem-service/api/workitems/5e3e9519-78ea-4409-
a52c-e4aa3bc56511/actions/com.mycompany.csp.core.approval.action.approve
--d @approve.json
```

Example: API Explorer

You can use the following command to submit the CompleteRequest request as approved.

```
vcac-cli>rest post --service workitem-service --u workitems/5e3e9519-78ea-4409-
a52c-e4aa3bc56511/actions/com.mycompany.csp.core.approval.action.approve
--d @approve.json  --h
```

Error Conditions

If you submit the same request a second time, you receive the following error response:

```
Command failed [Rest Error]: {Status code: 400}, {Error code: 12005} ,
{Error Source: null}, {Error Msg: Work item 5e3e9519-78ea-4409-a52c-e4aa3bc56511
is in COMPLETED state. Requested operation cannot be performed.}, {System Msg:
Work item 5e3e9519-78ea-4409-a52c-e4aa3bc56511 is in COMPLETED state. Requested
operation cannot be performed.}
```

If you submit the request logged in as a user who is not authorized to approve the request, you receive the following error response:

```
Command failed [Rest Error]: {Status code: 400}, {Error code: 12017} ,
{Error Source: null}, {Error Msg: User fritz@example.mycompany.com not authorized to
complete work item with ID 5e3e9519-78ea-4409-a52c-e4aa3bc56511.}, {System Msg:
User fritz@example.mycompany.com not authorized to complete Work item with id
5e3e9519-78ea-4409-a52c-e4aa3bc56511.}
```

Listing Your Provisioned Resources

You can log in to vRealize Automation and display a full or filtered list of your provisioned resources using the REST API.

The checklist provides the tasks required to display a list of provisioned resources with the REST API. Perform the tasks in sequence.

Table 3-8. Viewing a List of Resources Checklist

Task	Details	Permissions
❑ Request an HTTP bearer token	See Chapter 2, "REST API Authentication," on page 9.	
❑ Log in as a business group manager	See "Log in as a Business Group Manager," on page 112.	consumer and current business group member
❑ Display a list of provisioned resources	See "Display Your Provisioned Resources," on page 113.	consumer and current business group member
❑ Display provisioned resources filtered by their resource type	See "Display Provisioned Resources by Resource Type," on page 115 .	consumer and current business group member
❑ Display a list of available resource types	See "Display All Available Resource Types," on page 118.	consumer and current business group member
❑ Display a list or provisioned resource for the business groups that you manage	See "Display Provisioned Resources by Business Groups You Manage," on page 120.	consumer and current business group member
❑Display machine details for a provisioned machine	See "View Machine Details," on page 127.	consumer and current business group member

Log in as a Business Group Manager

You can log in to vRealize Automation as a business group manager and access the provisioned resources that are owned by business groups that you manage.

Prerequisites

- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- You must have business manager credentials for the named tenant.

Inputs

The following REST API command line format is used.

```
login --url vra_url --user username@fqdn --password my_pwd --tenant tname
```

Table 3-9. Command Line Input Variables and Descriptions

Input	Description
url	Specifies the vRealize Automation system URL.
user	Specifies the business group manager account user name and fully qualified domain name
password	Specifies the business group manager account password.
tenant	Specifies the tenant name. If you do not specify a tenant name, the default tenant (vsphere.local) is used.
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

A successful login returns the command line prompt with no error message.

Example: `curl` Command

The following command logs you in to the named tenant as a business group manager using sample variable values.

```
curl –-insecure –H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://my.example.com –-user jdoe@mycompany.com –-password dpwg5sQam –-tenant MYCOMPANY
```

Example: API Explorer

The following command logs you in to the named tenant as a business group manager using sample variable values.

```
login –-url https://my.example.com –-user jdoe@mycompany.com –-password dpwg5sQam –-tenant
MYCOMPANY
```

Display Your Provisioned Resources

You can display a list of all the provisioned resources that you own.

Prerequisites

- Log in to vRealize Automation as a **business group manager**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/resources
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
id	Specifies the unique identifier of this resource.
iconId	
resourceTypeRef	Specifies the resource type.
name	Specifies the resource name.
description	Specifies the resource description.
status	Specifies the resource status.
catalogItem	Specifies the catalog item that defines the service this resource is based on.
requestId	Specifies the request ID that provisioned this resource.
providerBinding	Specifies the provider binding.
owners	Species the owners of this resource.
organization	Specifies the subtenant or tenant that owns this resource.
dateCreated	Specifies the data and time at which the resource was created.
lastUpdated	Specifies the date and time at which the resource was most recently modified.
hasLease	Returns true if the resource is subject to a lease.
lease	Displays the resource's current lease as start and end time stamps.
leaseForDisplay	Specifies the resource's current lease, #getLease, with time units synchronized with #getCosts.
hasCosts	Returns true if the resource is subject to per-time costs.
costs	Displays an optional rate of the cost charges for the resource.
costToDate	Displays an optional rate of the current cost charges for the resource.
totalCost	Displays an optional rate of the cost charges for the entire lease period.
parentResourceRef	Displays the parent of this resource.
childResources	Displays the children of this resource.
operations	Specifies the sequence of available operations that you can perform on this resource.
forms	Specifies the forms used to render this resource.
resourceData	Displays the extended provider-defined properties of the resource.

Example: `curl` Command

You can use the following command to display all your provisioned resources.

```
curl –-insecure –H "Content–Type: application/json"
–H "Authorization: Bearer $token"
https://$host/catalog–service/api/consumer/resources/?page=n&limit=n
```

Example: API Explorer

You can use the following command to display all your provisioned resources.

```
rest get --service catalog-service --u /consumer/resources/?page=n&limit=n
```

Example: JSON Output

The following JSON output is returned based on your command input.


```
{
  "links" : [ {
    "@type" : "link",
    "rel" : "next",
    "href" : "https://vra152-009-067.mycompany.com/catalog-service/api/consumer/resources/?
page=2&limit=1"
  } ],
  "content" : [ {
    "@type" : "ConsumerResource",
    "id" : "c24e8c75-c201-489c-b51c-8d7009c23563",
    "iconId" : "Travel_100.png",
    "resourceTypeRef" : {
      "id" : "com.mycompany.mystuff.samples.travel.packageType",
      "label" : "Reservation"
    },
    "name" : "example",
    "description" : "asd",
    "status" : "ACTIVE",
    "catalogResource" : {
      "id" : "6fddafcd-bc3d-4753-8a2a-5fa3f78a5a90",
      "label" : "example"
    },
    "requestId" : "55e7fcf3-4c77-4b11-a442-1f282333ac91",
    "providerBinding" : {
      "bindingId" : "1",
      "providerRef" : {
        "id" : "f60f5d1e-d6e9-4d98-9c48-f70a3e405346",
        "label" : "travel-service"
      }
    },
…
}
```


Display Provisioned Resources by Resource Type

You can display a list of the provisioned resources that you own filtered by machine resource type.

Prerequisites

- Log in to vRealize Automation as a **business group manager**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/resourceType
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

You can filter by the following resource types:

- Infrastructure.Machine
- Infrastructure.AppServic
- Infrastructure.Cloud
- Infrastructure.Physical
- Infrastructure.vApp
- Infrastructure.Virtual

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
id	Specifies the unique identifier of this resource.
iconId	
resourceTypeRef	Specifies the resource type.
name	Specifies the resource name.
description	Specifies the resource description.
status	Specifies the resource status.
catalogItem	Specifies the catalog item that defines the service this resource is based on.
requestId	Specifies the request ID that provisioned this resource.
providerBinding	Specifies the provider binding.
owners	Species the owners of this resource.
organization	Specifies the subtenant or tenant that owns this resource.
dateCreated	Specifies the data and time at which the resource was created.
lastUpdated	Specifies the date and time at which the resource was most recently modified.
hasLease	Returns true if the resource is subject to a lease.
lease	Displays the resource's current lease as start and end time stamps.
leaseForDisplay	Specifies the resource's current lease, #getLease, with time units synchronized with #getCosts.
hasCosts	Returns true if the resource is subject to per-time costs.
costs	Displays an optional rate of the cost charges for the resource.
costToDate	Displays an optional rate of the current cost charges for the resource.
totalCost	Displays an optional rate of the cost charges for the entire lease period.

Property	Description
parentResourceRef	Displays the parent of this resource.
childResources	Displays the children of this resource.
operations	Specifies the sequence of available operations that you can perform on this resource.
forms	Specifies the forms used to render this resource.
resourceData	Displays the extended provider-defined properties of the resource.

Example: `curl` Command

You can use the following command to display your provisioned resources by resource type.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/resourceTypes/Infrastructure.Machine/?page=1&limit=1
```

Example: API Explorer

You can use the following command to display your provisioned resources by resource type.

```
rest get --service catalog-service --u /consumer/resourceTypes/Infrastructure.Machine/?
page=1&limit=1
```

Example: JSON Output

In this example, the highlighted resource ID (**`3bfde906-81b9-44c3-8c2d-07d2c9768168`**) corresponds to a provisioned machine owned by the logged-in user. You can use resource IDs in requests to retrieve details for the corresponding machines.

Also in this example, the subtenantRef ID (**`eab762cb-6e75-4379-83ef-171a71c9f00e`**) corresponds to the business group of the logged-in user. If the logged-in user is also the manager of the business group, you can use the subtenantRef ID to get resources from all business groups that the user manages.

The following JSON output is returned based on your command input.


```
{
  "links" : [ ],
  "content" : [ {
    "@type" : "ConsumerResource",
    "id" : "3bfde906-81b9-44c3-8c2d-07d2c9768168",
    "iconId" : "cafe_default_icon_genericCatalogResource",
    "resourceTypeRef" : {
      "id" : "Infrastructure.Virtual",
      "label" : "Virtual Machine"
    },
    "name" : "test2",
    "description" : null,
    "status" : "ACTIVE",
    "catalogResource" : {
      "id" : "e2f397be-72ad-4ec4-a688-c017560fa1a3",
      "label" : "test-blueprint"
    },
    "requestId" : "b013d2fa-4ba4-416c-b46b-98bb8cc7b076",
    "providerBinding" : {
      "bindingId" : "8a4581a0-84f9-4e80-9af6-75d79633e382",
      "providerRef" : {
        "id" : "6918cd49-b737-467f-94bf-d14d52c78fba",
```

```
        "label" : "iaas-service"
      }
    },
    "owners" : [ {
      "tenantName" : "MYCOMPANY",
      "ref" : "fritz@example.mycompany.com",
      "type" : "USER",
      "value" : "Fritz Arbeiter"
    } ],
    "organization" : {
      "tenantRef" : "MYCOMPANY",
      "tenantLabel" : "QETenant",
      "subtenantRef" : "eab762cb-6e75-4379-83ef-171a71c9f00e",
      "subtenantLabel" : "MyTestAgentBusinessGroup"
    },
…
}
```

Display All Available Resource Types

You can display all the resource types that are available on the system.

Prerequisites

- Log in to vRealize Automation as a **business group manager**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/resourceType
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
id	Specifies the unique identifier of this resource.
iconId	
resourceTypeRef	Specifies the resource type.
name	Specifies the resource name.
description	Specifies the resource description.
status	Specifies the resource status.

Property	Description
catalogItem	Specifies the catalog item that defines the service this resource is based on.
requestId	Specifies the request ID that provisioned this resource.
providerBinding	Specifies the provider binding.
owners	Species the owners of this resource.
organization	Specifies the subtenant or tenant that owns this resource.
dateCreated	Specifies the data and time at which the resource was created.
lastUpdated	Specifies the date and time at which the resource was most recently modified.
hasLease	Returns true if the resource is subject to a lease.
lease	Displays the resource's current lease as start and end time stamps.
leaseForDisplay	Specifies the resource's current lease, #getLease, with time units synchronized with #getCosts.
hasCosts	Returns true if the resource is subject to per-time costs.
costs	Displays an optional rate of the cost charges for the resource.
costToDate	Displays an optional rate of the current cost charges for the resource.
totalCost	Displays an optional rate of the cost charges for the entire lease period.
parentResourceRef	Displays the parent of this resource.
childResources	Displays the children of this resource.
operations	Specifies the sequence of available operations that you can perform on this resource.
forms	Specifies the forms used to render this resource.
resourceData	Displays the extended provider-defined properties of the resource.

Example: `curl` Command

You can use the following command to display all available resource types.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/resourceTypes
```

Example: API Explorer

You can use the following command to display all available resource types.

```
rest get --service catalog-service --u /consumer/resourceTypes
```

Example: JSON Output

The following JSON output is returned based on your command input.

```
{
  "links" : [ ],
  "content" : [ {
    "@type" : "ResourceType",
    "id" : "Infrastructure.Machine",
    "name" : "Machine",
    "pluralizedName" : "Machines",
    "description" : "The common parent type for all types of machines",
    "primary" : true,
    "schema" : {
```

```
    "classId" : "Infrastructure.Machine.Schema",
    "typeFilter" : null
  },
  "forms" : {
    "catalogResourceInfoHidden" : true,
    "details" : {
      "type" : "extension",
      "extensionId" : "csp.places.iaas.resource.details",
      "extensionPointId" : null
    }
```

Display Provisioned Resources by Business Groups You Manage

You can display all of the provisioned resources that are owned by the business groups that you manage. You can optionally filter the list by business group name.

Prerequisites

- Log in to vRealize Automation as a **business group manager**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the business group subtenant ID values to specify on the command line. See "Display Your Provisioned Resources," on page 113.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/resources/type
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
id	Specifies the unique identifier of this resource.
iconId	
resourceTypeRef	Specifies the resource type.
name	Specifies the resource name.
description	Specifies the resource description.
status	Specifies the resource status.
catalogItem	Specifies the catalog item that defines the service this resource is based on.
requestId	Specifies the request ID that provisioned this resource.
providerBinding	Specifies the provider binding.

Property	Description
owners	Species the owners of this resource.
organization	Specifies the subtenant or tenant that owns this resource.
dateCreated	Specifies the data and time at which the resource was created.
lastUpdated	Specifies the date and time at which the resource was most recently modified.
hasLease	Returns true if the resource is subject to a lease.
lease	Displays the resource's current lease as start and end time stamps.
leaseForDisplay	Specifies the resource's current lease, #getLease, with time units synchronized with #getCosts.
hasCosts	Returns true if the resource is subject to per-time costs.
costs	Displays an optional rate of the cost charges for the resource.
costToDate	Displays an optional rate of the current cost charges for the resource.
totalCost	Displays an optional rate of the cost charges for the entire lease period.
parentResourceRef	Displays the parent of this resource.
childResources	Displays the children of this resource.
operations	Specifies the sequence of available operations that you can perform on this resource.
forms	Specifies the forms used to render this resource.
resourceData	Displays the extended provider-defined properties of the resource.

Example: `curl` Command

You can use the following command to display the provisioned resources of one or more business groups.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/resources/types/Infrastructure.Machine/?
page=1&limit=2&$orderby=dateCreated desc&$filter=((organization/subTenant/id eq
'subtenantID_group1') or (organization/subTenant/id eq ''subtenantID_group2') … )"
```

Example: API Explorer

You can use the following command to display the provisioned resources of one or more business groups.

```
rest get catalog-service --u  "consumer/resources/types/Infrastructure.Machine/?page=1&limit=2&
$orderby=dateCreated desc&$filter=((organization/subTenant/id eq 'subtenantID_group1') or
(organization/subTenant/id eq ''subtenantID_group2') … )"
```

Example: JSON Output

The following JSON output is returned based on your command input.

For the following command line, the specified subtenant IDs correspond to business groups that are managed by the logged-in user.

```
rest get catalog-service --u  "consumer/resources/types/Infrastructure.Machine/?page=1&limit=2&
$orderby=dateCreated desc&$filter=((organization/subTenant/id eq
'eab762cb-6e75-4379-83ef-171a71c9f00e') or (organization/subTenant/id eq 'fa995528-e289-455e-
a0e6-c2da8b0e1bf9') or (organization/subTenant/id eq '699efe66-fe6e-4e34-96e8-52a34f338d20') or
(organization/subTenant/id eq '4d949784-e93e-4538-accb-6a0a464e4a4b'))"
```

The following JSON output is returned based on your command input.


```
{
  "links" : [ ],
  "content" : [ {
    "@type" : "ConsumerResource",
    "id" : "3bfde906-81b9-44c3-8c2d-07d2c9768168",
    "iconId" : "cafe_default_icon_genericCatalogResource",
    "resourceTypeRef" : {
      "id" : "Infrastructure.Virtual",
      "label" : "Virtual Machine"
    },
    "name" : "test2",
    "description" : null,
    "status" : "ACTIVE",
    "catalogResource" : {
      "id" : "e2f397be-72ad-4ec4-a688-c017560fa1a3",
      "label" : "test-blueprint"
    },
    "requestId" : "b013d2fa-4ba4-416c-b46b-98bb8cc7b076",
    "providerBinding" : {
      "bindingId" : "8a4581a0-84f9-4e80-9af6-75d79633e382",
      "providerRef" : {
        "id" : "6918cd49-b737-467f-94bf-d14d52c78fba",
        "label" : "iaas-service"
      }
    },
    "owners" : [ {
      "tenantName" : "MYCOMPANY",
      "ref" : "fritz@example.mycompany.com",
      "type" : "USER",
      "value" : "Fritz Arbeiter"
    } ],
    "organization" : {
      "tenantRef" : "MYCOMPANY",
      "tenantLabel" : "QETenant",
      "subtenantRef" : "eab762cb-6e75-4379-83ef-171a71c9f00e",
      "subtenantLabel" : "MyTestAgentBusinessGroup"
    },
    "dateCreated" : "2014-09-19T21:19:37.541Z",
    "lastUpdated" : "2014-09-19T21:19:40.888Z",
    "hasLease" : true,
    "lease" : {
      "start" : "2014-09-19T21:18:57.000Z"
    },
    "leaseForDisplay" : null,
    "hasCosts" : true,
    "costs" : {
      "leaseRate" : {
        "type" : "moneyTimeRate",
        "cost" : {
          "type" : "money",
          "currencyCode" : "USD",
          "amount" : 0.0
        },
        "basis" : {
```

```
      "type" : "timeSpan",
      "unit" : "DAYS",
      "amount" : 1
    }
  }
},
"costToDate" : {
  "type" : "money",
  "currencyCode" : "USD",
  "amount" : 0.0
},
"totalCost" : null,
"childResources" : [ ],
"operations" : [ {
  "name" : "Reprovision",
  "description" : "Reprovision a machine.",
  "iconId" : "machineReprovision.png",
  "type" : "ACTION",
  "id" : "a1caee9b-d67f-41e8-a7b3-131616a0f6ac",
  "extensionId" : null,
  "providerTypeId" : "com.mycompany.csp.iaas.blueprint.service",
  "bindingId" : "Infrastructure.Machine.Action.Reprovision",
  "hasForm" : false,
  "formScale" : null
} ],
"forms" : {
  "catalogResourceInfoHidden" : true,
  "details" : {
    "type" : "extension",
    "extensionId" : "csp.places.iaas.resource.details",
    "extensionPointId" : null
  }
},
"resourceData" : {
  "entries" : [ {
    "key" : "Expire",
    "value" : {
      "type" : "boolean",
      "value" : true
    }
  }, {
    "key" : "MachineGroupName",
    "value" : {
      "type" : "string",
      "value" : "MyTestAgentBusinessGroup"
    }
  }, {
    "key" : "NETWORK_LIST",
    "value" : {
      "type" : "multiple",
      "elementTypeId" : "COMPLEX",
      "resources" : [ {
        "type" : "complex",
        "componentTypeId" : "com.mycompany.csp.component.iaas.proxy.provider",
        "componentId" : null,
```

```
      "classId" : "vra.api.model.NetworkViewModel",
      "typeFilter" : null,
      "values" : {
        "entries" : [ {
          "key" : "NETWORK_MAC_ADDRESS",
          "value" : {
            "type" : "string",
            "value" : "56:52:4d:e7:46:d4"
          }
        }, {
          "key" : "NETWORK_NAME",
          "value" : {
            "type" : "string",
            "value" : "Test Agent-network-1"
          }
        } ]
      }
    } ]
  }
}, {
  "key" : "SNAPSHOT_LIST",
  "value" : {
    "type" : "multiple",
    "elementTypeId" : "COMPLEX",
    "resources" : [ ]
  }
}, {
  "key" : "ConnectViaRdp",
  "value" : {
    "type" : "boolean",
    "value" : true
  }
}, {
  "key" : "MachineStatus",
  "value" : {
    "type" : "string",
    "value" : "On"
  }
}, {
  "key" : "PowerOff",
  "value" : {
    "type" : "boolean",
    "value" : true
  }
}, {
  "key" : "DISK_VOLUMES",
  "value" : {
    "type" : "multiple",
    "elementTypeId" : "COMPLEX",
    "resources" : [ {
      "type" : "complex",
      "componentTypeId" : "com.mycompany.csp.component.iaas.proxy.provider",
      "componentId" : null,
      "classId" : "vra.api.model.DiskInputModel",
      "typeFilter" : null,
```

```
      "values" : {
        "entries" : [ {
          "key" : "DISK_CAPACITY",
          "value" : {
            "type" : "integer",
            "value" : 1
          }
        }, {
          "key" : "DISK_DRIVE",
          "value" : {
            "type" : "string",
            "value" : "c"
          }
        }, {
          "key" : "DISK_INPUT_ID",
          "value" : {
            "type" : "string",
            "value" : "DISK_INPUT_ID1"
          }
        } ]
      }
    } ]
  }
}, {
  "key" : "MachineBlueprintName",
  "value" : {
    "type" : "string",
    "value" : "test-blueprint"
  }
}, {
  "key" : "Suspend",
  "value" : {
    "type" : "boolean",
    "value" : true
  }
}, {
  "key" : "Reboot",
  "value" : {
    "type" : "boolean",
    "value" : true
  }
}, {
  "key" : "Reprovision",
  "value" : {
    "type" : "boolean",
    "value" : true
  }
}, {
  "key" : "MachineStorage",
  "value" : {
    "type" : "integer",
    "value" : 1
  }
}, {
  "key" : "MachineDailyCost",
```

```
    "value" : {
      "type" : "decimal",
      "value" : 0.0
    }
  }, {
    "key" : "Destroy",
    "value" : {
      "type" : "boolean",
      "value" : true
    }
  }, {
    "key" : "MachineType",
    "value" : {
      "type" : "string",
      "value" : "Virtual"
    }
  }, {
    "key" : "InstallTools",
    "value" : {
      "type" : "boolean",
      "value" : true
    }
  }, {
    "key" : "Shutdown",
    "value" : {
      "type" : "boolean",
      "value" : true
    }
  }, {
    "key" : "ChangeLease",
    "value" : {
      "type" : "boolean",
      "value" : true
    }
  }, {
    "key" : "machineId",
    "value" : {
      "type" : "string",
      "value" : "8a4581a0-84f9-4e80-9af6-75d79633e382"
    }
  }, {
    "key" : "MachineMemory",
    "value" : {
      "type" : "integer",
      "value" : 0
    }
  }, {
    "key" : "MachineGuestOperatingSystem"
  }, {
    "key" : "MachineName",
    "value" : {
      "type" : "string",
      "value" : "test2"
    }
  }, {
```

```
        "key" : "MachineDestructionDate"
      }, {
        "key" : "MachineCPU",
        "value" : {
          "type" : "integer",
          "value" : 1
        }
      }, {
        "key" : "MachineInterfaceType",
        "value" : {
          "type" : "string",
          "value" : "Test"
        }
      }, {
        "key" : "MachineReservationName",
        "value" : {
          "type" : "string",
          "value" : "Test Agent-Res-1"
        }
      }, {
        "key" : "Reconfigure",
        "value" : {
          "type" : "boolean",
          "value" : true
        }
      }, {
        "key" : "EXTERNAL_REFERENCE_ID"
      }, {
        "key" : "MachineExpirationDate"
      }, {
        "key" : "Reset",
        "value" : {
          "type" : "boolean",
          "value" : true
        }
      } ]
    }
  } ],
  "metadata" : {
    "size" : 2,
    "totalElements" : 1,
    "totalPages" : 1,
    "number" : 1,
    "offset" : 0
  }
}
```


View Machine Details

You can display the machine details for a provisioned machine.

Prerequisites

- Log in to vRealize Automation as a **business group manager**.

- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the resource ID of the provisioned machine to query. See "Display Your Provisioned Resources," on page 113.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/catalog-service/api/consumer/resources/$resourceId
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$resourceID	Specifies a resource ID. See "Display Your Provisioned Resources," on page 113 to view all of your requests and search for a request ID.
managedOnly	If true, the returned requests are from the user's managed subtenants.
page	Specifies a page number.
limit	Specifies the number of entries to display on a page.
$orderby	Specifies how to order multiple comma-separated properties sorted in ascending or descending order.
$top	Specifies the number of returned entries from the top of the response (total number per page in relation to skip).
$skip	Specifies the number of entries to skip.
filter	Contains a Boolean expression to determine if a particular entry is included in the response.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
id	Specifies the unique identifier of this resource.
iconId	
resourceTypeRef	Specifies the resource type.
name	Specifies the resource name.
description	Specifies the resource description.
status	Specifies the resource status.
catalogItem	Specifies the catalog item that defines the service this resource is based on.
requestId	Specifies the request ID that provisioned this resource.
providerBinding	Specifies the provider binding.
owners	Species the owners of this resource.
organization	Specifies the subtenant or tenant that owns this resource.
dateCreated	Specifies the data and time at which the resource was created.
lastUpdated	Specifies the date and time at which the resource was most recently modified.

Property	Description
hasLease	Returns true if the resource is subject to a lease.
lease	Displays the resource's current lease as start and end time stamps.
leaseForDisplay	Specifies the resource's current lease, #getLease, with time units synchronized with #getCosts.
hasCosts	Returns true if the resource is subject to per-time costs.
costs	Displays an optional rate of the cost charges for the resource.
costToDate	Displays an optional rate of the current cost charges for the resource.
totalCost	Displays an optional rate of the cost charges for the entire lease period.
parentResourceRef	Displays the parent of this resource.
childResources	Displays the children of this resource.
operations	Specifies the sequence of available operations that you can perform on this resource.
forms	Specifies the forms used to render this resource.
resourceData	Displays the extended provider-defined properties of the resource.

Example: `curl` Command

You can use the following command to display machine details for a provisioned machine, where *resourceID* is the ID of the provisioned machine.

```
curl –-insecure –H "Content–Type: application/json"
–H "Authorization: Bearer $token"
https://$host/catalog–service/api/consumer/resources/resourceID
```

Example: API Explorer

You can use the following command to display machine details for a provisioned machine, where *resourceID* is the ID of the provisioned machine.

```
rest get –-service catalog–service –-u /consumer/resources/resourceID
```

Example: JSON Output

In the following example, the provisioned machine *resourceID* value specified in the command line was **`3bfde906–81b9–44c3–8c2d–07d2c9768168`**.

```
{
  "id" : "3bfde906–81b9–44c3–8c2d–07d2c9768168",
  "iconId" : "cafe_default_icon_genericCatalogResource",
  "resourceTypeRef" : {
    "id" : "Infrastructure.Virtual",
    "label" : "Virtual Machine"
  },
  "name" : "test2",
  "description" : null,
  "status" : "DELETED",
  "catalogResource" : {
    "id" : "e2f397be–72ad–4ec4–a688–c017560fa1a3",
    "label" : "test–blueprint"
  },
  "requestId" : "b013d2fa–4ba4–416c–b46b–98bb8cc7b076",
  "providerBinding" : {
    "bindingId" : "8a4581a0–84f9–4e80–9af6–75d79633e382",
```

```
  "providerRef" : {
    "id" : "6918cd49-b737-467f-94bf-d14d52c78fba",
    "label" : "iaas-service"
  }
},
"owners" : [ {
  "tenantName" : "MYCOMPANY",
  "ref" : "fritz@example.mycompany.com",
  "type" : "USER",
  "value" : "Fritz Arbeiter"
} ],
"organization" : {
  "tenantRef" : "MYCOMPANY",
  "tenantLabel" : "QETenant",
  "subtenantRef" : "eab762cb-6e75-4379-83ef-171a71c9f00e",
  "subtenantLabel" : "MyTestAgentBusinessGroup"
},
"dateCreated" : "2014-02-19T21:19:37.541Z",
"lastUpdated" : "2014-02-20T21:41:08.478Z",
"hasLease" : true,
"lease" : {
  "start" : "2014-02-19T21:18:57.000Z"
},
"leaseForDisplay" : null,
"hasCosts" : true,
"costs" : {
  "leaseRate" : {
    "type" : "moneyTimeRate",
    "cost" : {
      "type" : "money",
      "currencyCode" : "USD",
      "amount" : 0.0
    },
    "basis" : {
      "type" : "timeSpan",
      "unit" : "DAYS",
      "amount" : 1
    }
  }
},
"costToDate" : {
  "type" : "money",
  "currencyCode" : "USD",
  "amount" : 0.0
},
"totalCost" : null,
"childResources" : [ ],
"operations" : [ ],
"forms" : {
  "catalogResourceInfoHidden" : true,
  "details" : {
    "type" : "extension",
    "extensionId" : "csp.places.iaas.resource.details",
    "extensionPointId" : null
  }
```

```
  },
  "resourceData" : {
    "entries" : [ ]
  }
}
```

Reprovisioning a Machine Resource

You can list the available actions for a provisioned machine and then reprovision the machine using the REST API.

The checklist provides the tasks required to list the available actions for a provisioned machine and reprovision the machine with the REST API. Perform the tasks in sequence.

Table 3-10. Reprovisioning a Machine Resource Checklist

Task	Details	Permissions
❑ Request an HTTP bearer token	Chapter 2, "REST API Authentication," on page 9	
❑ View available actions on a provisioned machine	"View Available Actions for a Provisioned Machine," on page 131	consumer and current business group member
❑ Reprovision a machine	"Reprovision a Provisioned Machine," on page 133	consumer and current business group member

View Available Actions for a Provisioned Machine

You can display a list of actions enabled on the blueprint used to provision the specified machine, entitled to the logged-in user, and available in the current state of the provisioned machine.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the request ID (*$requestId*) of the request for which to view status. See "View All Your Requests," on page 47.

Input

You can use supported input parameters to control the command output.

Table 3-11. Inputs for Viewing the Available Actions for a Provisioned Machine

Input	Description
URL	https://*$host*/catalog-service/api/consumer/requests/*$requestId*
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$resourceId	Specifies the resource ID for the request.

Output

The command output contains property names and values based on the API command input parameters.

Table 3-12. Outputs for Viewing the Available Actions for a Provisioned Machine

Property	Description
type	Specifies the operation type. The property type is resourceOperationType.
id	Specifies the identifier for the resource operation. The property type is string.
extensionId	Specifies the unique ID of the UI extension that is associated with the operation if `#getType()` is set to `ResourceOperationType#EXTENSION`.
providerTypeId	Specifies the ID type for providers that support the action if `#getType()` is set to `ResourceOperationType#ACTION`. The property type is string.
bindingId	Specifies the unique ID of the action that the external provider that published it recognizes if `#getType()` is set to `ResourceOperationType#ACTION`. The property type is string.
hasForm	Indicates if the action has a request form to complete if `#getType()` is set to `ResourceOperationType#ACTION`. The property type is Boolean.
formScale	Specifies the form scale value of the request form for the action, if applicable. The property type is formScale.

Example: `curl` Command

You can use the following command to display the available actions for a provisioned machine by using its resource ID.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/resources/resourceID/actions
```

Example: API Explorer

You can use the following command to display the available actions for a provisioned machine by using its resource ID.

```
rest get --service catalog-service --u /consumer/resources/resourceID/actions
```

Example: JSON Output

The highlighted resource action ID corresponds to the reprovisioning actions that are available for the specified machine.

```
{
  "links" : [ ],
  "content" : [ {
    "@type" : "ConsumerResourceOperation",
    "name" : "Reprovision",
    "description" : "Reprovision a machine.",
    "iconId" : "machineReprovision.png",
    "type" : "ACTION",
    "id" : "a1caee9b-d67f-41e8-a7b3-131616a0f6ac",
```

```
    "extensionId" : null,
    "providerTypeId" : "com.mycompany.csp.iaas.blueprint.service",
    "bindingId" : "Infrastructure.Machine.Action.Reprovision",
    "hasForm" : false,
    "formScale" : null
  } ]
}
```

Reprovision a Provisioned Machine

You can reprovision a provisioned machine, or perform other entitled and enabled actions with the catalog service.

Prerequisites

- Log in to vRealize Automation as a consumer and current business group user.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the request ID (*$requestId*) of the request for which to view status. See "View All Your Requests," on page 47.
- "View Available Actions for a Provisioned Machine," on page 131.

Creating a JSON Input File

You can use the following procedure to create a JSON input file for use with this command action.

1 Copy the JSON Input File Template to a new JSON text file.

2 Substitute your specific values for the input variables in the template.

3 Save the file with a `.json` extension.

Input

You can use supported input parameters to control the command output.

The JSON Template Values table describes the information you can use in, in conjunction with the JSON template format, to create a JSON text file to include in the reprovision command request.

Table 3-13. JSON Template Values

Property	Description
resourceRef : id	Specifies the resource ID of the resource on which the operation is to be performed.
resourceActionRef : id	Specifies the resource action ID on which the operations is to be performed.
organization	Specifies the organization to which the resource belongs. Supply the tenant and subtenant information as necessary. The tenant corresponds to the organization and the subtenant corresponds to the business group.

Table 3-13. JSON Template Values (Continued)

Property	Description
State	Specifies the state of the request. At the time of requesting, the state is SUBMITTED. The other possible state values are UNSUBMITTED, SUBMITTED, DELETED, PENDING_PRE_APPROVAL, PRE_APPROVAL_SEND_ERROR, PRE_APPROVED, PRE_REJECTED, IN_PROGRESS, PROVIDER_SEND_ERROR, PROVIDER_COMPLETED, PROVIDER_FAILED, PENDING_POST_APPROVAL, POST_APPROVAL_SEND_ERROR, POST_APPROVED, POST_REJECTION_RECEIVED, ROLLBACK_ERROR, POST_REJECTED, SUCCESSFUL, and FAILED.
requestNumber	Contains Get information that is generated by the system.
requestData	Specifies other custom request data.

JSON Input File Template

You can use the following JSON template to create a JSON text file that contains the information you need to reprovision a machine.

This example reprov_action.json file contains a resource action request for the reprovision action (**`resourceActionRef`** and **`id`**) to be performed on the specified machine (**`resourceRef`** and **`id`**). The **`organization`** content include the names and labels of the tenant and business group.

```
{
    "@type": "ResourceActionRequest",
    "resourceRef": {
        "id": "b3adbe4f-274d-4a0c-8757-7843b8cb2ba4"
    },
    "resourceActionRef": {
        "id": "a1caee9b-d67f-41e8-a7b3-131616a0f6ac"
    },
    "organization": {
        "tenantRef": "MYCOMPANY",
        "tenantLabel": "QETenant",
        "subtenantRef": "eab762cb-6e75-4379-83ef-171a71c9f00e",
        "subtenantLabel": "MyTestAgentBusinessGroup"
    },
    "state": "SUBMITTED",
    "requestNumber": 0,
    "requestData": {
        "entries": []
    }
}
```

Example: `curl` Command

You can use the following command to call your JSON text file and reprovision a machine.

```
curl --insecure -H "Content-Type: application/json"
-H "Authorization: Bearer $token"
https://$host/catalog-service/api/consumer/requests --d @C:\reprov_action.json.txt
```

Example: API Explorer

You can use the following command to call your JSON text file and reprovision a machine.

```
rest post --headers --service catalog-service --u consumer/requests --d
@C:\reprov_action.json.txt
```

Working with Reservations

You can work with the reservation service to perform a variety of functions, such as creating and updating reservations.

The following reservation types are supported by the vRealize Automation REST API reservation service:

- Hyper-V
- KVM
- vSphere (except for FlexClone in vSphere)
- Xen
- Amazon
- vCloud

The following reservation types are not supported:

- Physical reservation
- OpenStack

The reservation service is extensible, which allows you to add new reservation types.

A reservation must belong to a business group, also referred to as a subtenant. A business group can have multiple reservations on the same resources or on different resources.

Creating a Reservation

You can create a reservation using the vRealize Automation REST API. This sample use case illustrates how to create a vSphere reservation.

The checklist provides the tasks required to create a reservation with the REST API. Perform the tasks in sequence to create a new reservation.

Table 3-14. Creating a Reservation Checklist

Task	Details	Permissions
❑ Request an HTTP bearer token	See "Request an HTTP Bearer Token," on page 10.	Logged in user
❑ Display a list of reservation types	See "Display a List of Supported Reservation Types," on page 136.	Fabric group administrator
❑ Display a schema definition for the reservation type	See "Display a Schema Definition for a Reservation Type," on page 138.	Fabric group administrator
❑ Get the business group ID, also referred to as the subtenant	See "Get the Business Group ID for a Reservation," on page 146.	Fabric group administrator
❑ Get compute resource information	See "Get a Compute Resource for the Reservation," on page 148.	Fabric group administrator
❑Get information about the permissible resource settings available for the reservation	See "Get the Resources Available For a Reservation," on page 151.	Fabric group administrator

Table 3-14. Creating a Reservation Checklist (Continued)

Task	Details	Permissions
❑Create a new reservation	See "Create a Reservation," on page 153.	Fabric group administrator
❑ Verify a new reservation	See "Verify a Reservation and Get Reservation Details," on page 158.	Fabric group administrator

Display a List of Supported Reservation Types

You can display a list of supported vRealize Automation reservation types with the reservation service.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations/types
Method	Get
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Species an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL that produces the result.
Content	Specifies an array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following information:
@type	Contains the `ReservationType` string.
createdDate	Specifies the create date.
lastUpdated	Specifies the last update date.
version	Specifies the version value.
Id	Specifies the unique reservation type identifier.
name	Specifies the reservation type name.
description	Specifies the reservation type description.

Property	Description
category	Specifies the reservation category of Virtual, Cloud or Physical.
serviceTypeId	Specifies the vRealize Automation service ID.
tenantId	This contains a null value.
FormReference	Specifies the user interface form reference. This field is valid for user interface elements only. ■ type -- user interface form type ■ formId -- user interface form ID
alertTypes	Contains the alert type list defined in the reservation type: ■ createdDate -- Alert type created date ■ lastUpdated -- Alert type last updated date ■ version -- Alert type version ■ id -- Unique identifier of alert type ■ name -- Name of alert type ■ description -- Long description of alert type ■ referenceResourceId -- Unique identifier of reference resource
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: `curl` Command

```
curl --insecure -H "Accept:application/json"
-H "Authorization: Bearer $token"
https://$host/reservation-service/api/reservations/types
```

The following command contains the example bearer token from "Request an HTTP Bearer Token," on page 10.

```
curl --insecure -H "Accept:application/json"
-H "Authorization: Bearer
MTQxMTY5OTkxODQyNTpkYmZmYjkzZTgzNjdmOGU0NThjZTp0ZW5hbnQ6cWV1c2VybmFtZTpmcml0ekBjb2tlLnZtd2
FyZS5jb206NDhmNGViNzQ3ZjYxY2YxMzdhNDAxOGY2MDAwOTFlZTJiZWI4MmJmZWU5ZTQ0MTI0YWI1M2U4NGNiOTk0
OTJjZjEwNjdhMzdmZTQ5YWMyMzA2NTA5M2UyNzlhMzI2ZGYxZDhlYTgxYmNkNjM5ZTNiNjIyYmEwYTRhOWJiMGE2ZTI="
https://myVRA.eng.mycompany.com/reservation-service/api/reservations/types
```

Example: API Explorer

Example: JSON Output

The following JSON output is returned based on your command input.

The following example output indicates that there are eight reservation types. The reservation type ID is `Infrastructure.Reservation.Virtual.vSphere`, and its schema class ID is `Infrastructure.Reservation.Virtual.vSphere`.

```
{
  "links": [],
  "content": [{
    "@type": "ReservationType",
    "createdDate": "2014-10-13T04:44:32.008Z",
```

```
    "lastUpdated": "2014-10-13T04:44:32.009Z",
    "version": 1,
    "id": "Infrastructure.Reservation.Virtual.vSphere",
    "name": "vSphere",
    "description": "vSphere Reservation",
    "category": "Virtual",
    "serviceTypeId": "com.mycompany.csp.iaas.blueprint.service",
    "tenantId": null,
    "formReference": {
      "type": "external",
      "formId": "Infrastructure.Reservation.Virtual.vSphere.form.new"
    },
    "schemaClassId": "Infrastructure.Reservation.Virtual.vSphere",
    "alertTypes": [{
      "createdDate": "2014-10-13T04:44:32.008Z",
      "lastUpdated": "2014-10-13T04:44:32.008Z",
      "version": 0,
      "id": "d248eeee-238c-4e87-9e95-f263b04d113f",
      "name": "storage",
      "description": null,
      "referenceResourceId": "storage"
  },//Omit 7 reservation types here
  ],
  "metadata": {
    "size": 20,
    "totalElements": 8,
    "totalPages": 1,
    "number": 1,
    "offset": 0
  }
}
```


Display a Schema Definition for a Reservation Type

You can display a schema definition for a specific vRealize Automation reservation type with the reservation service.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the schema class ID of the reservation type to create, in this example a vSphere reservation. See "Display a List of Supported Reservation Types," on page 136.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/data-service/schema/$schemaclassid/default
Method	Get

Input	Description
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$schemaclassid	Specifies the schema class of the reservation type. This is a common REST API. You can also call the REST server to display a schema class that is not a reservation type.

Output

The command output contains property names and values based on the API command input parameters.

Each field contains an array of data rows. Each data row represents one of the fields defined in the schema.

Property	Description
Id	Specifies the universally unique ID of the entity.
label	Specifies the field label.
dataType	Specifies the dataType field values.
type	Specifies the field value type: ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service determines the other names.
componentTypeid	Specifies the type ID of the component.
component	Specifies the unique identifier of the component
classId	Specifies the schema class of the field. This property is valid for complex and ref field types only.
label	Specifies the label of the field data type.
displayAdvice	Contains display advice for the field. This property is valid for a user interface element only.
permissibleValues	Optional field. If this field is a permissible value list field, define the meta info for the permissible value.
type	Specifies if the permissible value list is dynamic or static.
customAllowed	Specifies if a custom value is allowed during user input in this field.
dependencies	Specifies the list of fields that the current field depends on.
state	Provides a structure for defining the state of a content construct, for example {@link LayoutSection}. The element state identifies the field paths in the client data context upon which that element state depends. For example, the `callback` facet result indicates that facet evaluation must be delegated to the server of the object. This evaluation may be dependent on data collected in the client data context. For example, for a unique machine name, the evaluation requires the proposed name entered by the user.
dependencies	Contains the set of field paths on which the server-side evaluation of the facets depends.

Property	Description
facets	Provides a higher level view of an {@link Constraint} collection and its current values. All rendering code should use this class to provide a common place to get the current state of the field. If a field is considered in need of server-side evaluation, its `facets` setting is `callback`. IIf a field is considered mandatory, its `facets` setting is `mandatory`.
isMultiValued	Specifies if the field is a multi-value field, such as a list field. The state provides a higher level view of an {@link Constraint} collection and its current values. Rendering code should use this class to provide a common place to get the current state of the field.

Each reservation contains several fields. Some fields are common to all reservation types and some are type-specific. The list of type-specific fields is defined in a schema. You can call a data and schema service to get schema definition information. The data and schema services combines both fetch data and fetch schema REST calls.

Fields that are common to all reservation types are listed in the following table.

Table 3-15. Fields Common To All Reservation Types

Parameter (Field ID)	Description	Parameter Type
Id	Specifies the reservation ID.	GUID
name	Specifies the reservation name.	String
reservationTypeId	Specifies the reservation type, for example Infrastructure.Reservation.Virtual.vSphere.	String
tenantId	Specifies the tenant ID that contains the reservation.	String
subTenantId	Specifies the subtenant ID that contains the reservation.	GUID
enabled	Specifies whether the reservation is enabled.	Boolean
priority	Specifies the priority of the reservation during VM provisioning.	Integer
reservationPolicyId	Specifies the reservation policy ID to bind to this reservation.	GUID
alertPolicy	Specifies the alert policy of the reservation. The detail schema of this field refers to the alert policy.	JSON
extensionData	Contains type-specific fields. The detail schema of this field is retrieved by the data and schema service.	JSON

Example: `curl` Command

```
curl ––insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/reservation–service/api/data–service/schema/$schemaclassid/default
```

Example: API Explorer

Example: JSON Output

The schema definition in this example indicates that there are 9 extension fields supported for the vSphere type reservation. The following table lists the key attributes for those nine fields.

All the following field IDs except for `computeResource` and `machineQuota` depend on the `computeResource` field ID. The `computeResource` and `machineQuota` field IDs do not have a dependency on a field ID.

NOTE The information in the table is subject to change. You should call the data and schema service to retrieve the latest field information.

Table 3-16. Extension Fields Supported in vSphere Type Reservation

Field ID	Data Type	Type Class	Permissible Value
reservationNetworks	Complex Type	reservationNetwork	Y
reservationVCNSTransportZone	Entity Reference	NetworkScopes	Y
reservationVCNSSecurityGroups	Entity Reference	SecurityGroups	Y
reservationMemory	Complex Type	reservationMemory	Y
computeResource	Entity Reference	ComputeResource	Y
machineQuota	Integer	N/A	N
reservationStorages	Complex Type	reservationStorage	Y
resourcePool	Entity Reference	ResourcePools	Y
reservationVCNSRoutedGateways	Complex Type	reservationVCNSRoutedGateway	Y

The following JSON output is returned based on your command input.

```
{
  "fields": [{
    "id": "reservationNetworks",
    "label": "Network",
    "dataType": {
      "type": "complex",
      "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
      "componentId": null,
      "classId": "reservationNetwork",
      "typeFilter": null,
      "label": "Network"
    },
    "displayAdvice": "DATA_TABLE",
    "permissibleValues": {
      "type": "dynamic",
      "customAllowed": false,
      "dependencies": ["computeResource"]
    },
    "state": {
      "dependencies": [],
      "facets": [{
        "type": "mandatory",
        "value": {
          "type": "constantClause",
          "value": {
```

```
            "type": "boolean",
            "value": true
          }
        }
      }]
    },
    "isMultiValued": true
  },
  {
    "id": "reservationVCNSTransportZone",
    "label": "Transport Zone",
    "description": "Transport zone of the vCNS settings",
    "dataType": {
      "type": "ref",
      "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
      "componentId": null,
      "classId": "NetworkScopes",
      "typeFilter": null,
      "label": "Transport Zone"
    },
    "displayAdvice": null,
    "permissibleValues": {
      "type": "dynamic",
      "customAllowed": false,
      "dependencies": ["computeResource"]
    },
    "state": {
      "dependencies": [],
      "facets": []
    },
    "isMultiValued": false
  },
  {
    "id": "reservationVCNSSecurityGroups",
    "label": "Security Groups",
    "description": "Security groups of the vCNS settings",
    "dataType": {
      "type": "ref",
      "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
      "componentId": null,
      "classId": "SecurityGroups",
      "typeFilter": null,
      "label": "Security Group"
    },
    "displayAdvice": null,
    "permissibleValues": {
      "type": "dynamic",
      "customAllowed": false,
      "dependencies": ["computeResource"]
    },
    "state": {
      "dependencies": [],
      "facets": []
    },
    "isMultiValued": true
```

```
},
{
  "id": "reservationMemory",
  "label": "Memory",
  "dataType": {
    "type": "complex",
    "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
    "componentId": null,
    "classId": "reservationMemory",
    "typeFilter": null,
    "label": "Memory"
  },
  "displayAdvice": "DATA_TABLE",
  "permissibleValues": {
    "type": "dynamic",
    "customAllowed": false,
    "dependencies": ["computeResource"]
  },
  "state": {
    "dependencies": [],
    "facets": [{
      "type": "mandatory",
      "value": {
        "type": "constantClause",
        "value": {
          "type": "boolean",
          "value": true
        }
      }
    }]
  },
  "isMultiValued": false
},
{
  "id": "computeResource",
  "label": "Compute Resource",
  "description": "The compute resource for the reservation",
  "dataType": {
    "type": "ref",
    "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
    "componentId": null,
    "classId": "ComputeResource",
    "typeFilter": "InterfaceTypeId",
    "label": "Compute Resource"
  },
  "displayAdvice": null,
  "permissibleValues": {
    "type": "dynamic",
    "customAllowed": false,
    "dependencies": []
  },
  "state": {
    "dependencies": [],
    "facets": [{
      "type": "mandatory",
```

```
        "value": {
          "type": "constantClause",
          "value": {
            "type": "boolean",
            "value": true
          }
        }
      }]
    },
    "isMultiValued": false
  },
  {
    "id": "machineQuota",
    "label": "Machine Quota",
    "description": "The machine quota for the reservation",
    "dataType": {
      "type": "primitive",
      "typeId": "INTEGER"
    },
    "displayAdvice": null,
    "state": {
      "dependencies": [],
      "facets": []
    },
    "isMultiValued": false
  },
  {
    "id": "reservationStorages",
    "label": "Storage",
    "dataType": {
      "type": "complex",
      "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
      "componentId": null,
      "classId": "reservationStorage",
      "typeFilter": null,
      "label": "Storage"
    },
    "displayAdvice": "DATA_TABLE",
    "permissibleValues": {
      "type": "dynamic",
      "customAllowed": false,
      "dependencies": ["computeResource"]
    },
    "state": {
      "dependencies": [],
      "facets": [{
        "type": "mandatory",
        "value": {
          "type": "constantClause",
          "value": {
            "type": "boolean",
            "value": true
          }
        }
      }]
```

```
      },
      "isMultiValued": true
    },
    {
      "id": "resourcePool",
      "label": "Resource Pool",
      "description": "The resource pool for the reservation",
      "dataType": {
        "type": "ref",
        "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
        "componentId": null,
        "classId": "ResourcePools",
        "typeFilter": null,
        "label": "Resource Pool"
      },
      "displayAdvice": null,
      "permissibleValues": {
        "type": "dynamic",
        "customAllowed": false,
        "dependencies": ["computeResource"]
      },
      "state": {
        "dependencies": [],
        "facets": []
      },
      "isMultiValued": false
    },
    {
      "id": "reservationVCNSRoutedGateways",
      "label": "Routed Gateways",
      "dataType": {
        "type": "complex",
        "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
        "componentId": null,
        "classId": "reservationVCNSRoutedGateway",
        "typeFilter": null,
        "label": "Routed Gateways"
      },
      "displayAdvice": "DATA_TABLE",
      "permissibleValues": {
        "type": "dynamic",
        "customAllowed": false,
        "dependencies": ["computeResource"]
      },
      "state": {
        "dependencies": [],
        "facets": []
      },
      "isMultiValued": true
    }]
  }
```

Get the Business Group ID for a Reservation

You can get the business group ID for a vRealize Automation reservation with the reservation service. The business group is also referred to as the subtenant in the API. When you create a reservation, you must supply the business group ID, also referred to as the subtenant ID, in the REST command line. Use this procedure to obtain the `subTenantId` value.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/identity/api/tenants/$tenantId/subtenants
Method	Get
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$tenantId	Specifies the ID of the tenant. Use to indicate the tenant ID to be queried. Each subtenant, or business group, must belong to a tenant.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Species an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service determines the other names.
href	Specifies the URL which produces the result.
Content	Specifies an array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following information:
@type	Constants the `ReservationType` string.
Id	Specifies the unique reservation type identifier.
name	Specifies the reservation type name.
description	Specifies the reservation type description.
subtenantRoles	Specifies the business group roles.
extensionData	Specifies the extension data of the business group. For example, the email address of the vRealize Automation business group manager is `user1@mycompany.com`.

Property	Description
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: `curl` Command

```
insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/identity/api/tenants/qe/subtenants
```

Example: API Explorer

Example: JSON Output

In this example, all available subtenant IDs are displayed. The "Creating a Reservation," on page 135 use case, uses the **`ef58f604–528d–4441–a219–4725bead629b`** subtenant ID.

The following JSON output is returned based on your command input.


```
{
    "links": [],
    "content": [{
        "@type": "Subtenant",
        "id": "7d7dbb19–d2dc–44a3–9fc2–7435552c8a05",
        "name": "Development",
        "description": " Development ",
        "subtenantRoles": null,
        "extensionData": {
            "entries": [{
                "key": "iaas–manager–emails",
                "value": {
                    "type": "string",
                    "value": "user1@mycompany.com"
                }
            }]
        },
        "tenant": "qe"
    },
    {
        "@type": "Subtenant",
        "id": "ade5b8d3–decf–405e–bd0b–297f976ef721",
        "name": "Finance",
        "description": "Finance",
        "subtenantRoles": null,
        "extensionData": {
            "entries": [{
                "key": "iaas–manager–emails",
                "value": {
                    "type": "string",
                    "value": " user1@mycompany.com "
```

```
                }
            }]
        },
        "tenant": "qe"
    },
    {
        "@type": "Subtenant",
        "id": "ef58f604-528d-4441-a219-4725bead629b",
        "name": "Test Sub Tenant",
        "description": "VMPS",
        "subtenantRoles": null,
        "extensionData": {
            "entries": []
        },
        "tenant": "qe"
    },
    {
        "@type": "Subtenant",
        "id": "92926c91-37de-4647-9aee-70b8d557ce8d",
        "name": "Quality Engineering",
        "description": "created by demo content",
        "subtenantRoles": null,
        "extensionData": {
            "entries": [{
                "key": "iaas-manager-emails",
                "value": {
                    "type": "string",
                    "value": " user1@mycompany.com "
                }
            }]
        },
        "tenant": "qe"
    }],
    "metadata": {
        "size": 20,
        "totalElements": 4,
        "totalPages": 1,
        "number": 1,
        "offset": 0
    }
}
```


Get a Compute Resource for the Reservation

You can obtain a compute resource for the vRealize Automation reservation with the reservation service.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

When you create a reservation, you must provide compute resource information that corresponds to the `computeResource` parameter.

For a vSphere reservation type schema definition, the following `permissibleValues` field in the compute resource output indicates if the compute resource is available and if it has any dependencies.

```
"permissibleValues": {"type": "dynamic","customAllowed": false, "dependencies": []
```

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/data-service/schema/*$schemaclassid*/default/*$fieldid*/values
Method	Post
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$schemaclassid	From the schema definition, the `schemaclassid` of the compute resource field is `computeResource`. Provide the string **`computeResource`** for this placeholder.
HTTP body	Because the dependencies entry for this permissible value field is an empty string, provide an empty JASON string**`"{}"`** in the HTTP body.

Output

The command output contains property names and values based on the API command input parameters.

The `values` section contains an array of data rows, each of which represents one of the compute resource objects, returned in a pageable list. Each compute resource object contains the following information.

Property	Description
underlyingValue	Contains a JSON string representing one permissible value of field.
type	Specifies the data type of permissible value. ■ entityRef - Indicates that the object references a vRealize Automation entity. ■ complexRef - Indicates that the object is a user-defined complex structure, for example `struct` in C or `Pojo` in Java. ■ primary - Indicates the entity type such as string, integer, and so on.
componentId	Specifies the component ID.
classId	Specifies the schema class ID of the current data type.
Id	Specifies the unique compute resource identifier.
label	Specifies the compute resource label.
label	Contains the compute resource label. This value matches the `underlyingValue.label` .

Example: `curl` Command

```
curl ––insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/reservation–service/api/data–
service/schema/Infrastructure.Reservation.Virtual.vSphere
/default/ computeResource /values  –d “{}”
```

Example: API Explorer

Example: JSON Output

In this example output, there are 4 available compute resources for which you can create a vSphere type reservation, for example **cc254a84-95b8-434a-874d-bdfef8e8ad2c**. Save a copy of the **underlyingValue** section of the compute resource that you want to an XML editor file and use the section content later to create a reservation request.

The following JSON output is returned based on your command input.


```
{
  "values": [{
    "underlyingValue": {
      "type": "entityRef",
      "componentId": null,
      "classId": "ComputeResource",
      "id": "047e00f5-5424-4ed2-a751-4a334aeaff54",
      "label": "VC51-Cluster"
    },
    "label": "VC51-Cluster"
  },
  {
    "underlyingValue": {
      "type": "entityRef",
      "componentId": null,
      "classId": "ComputeResource",
      "id": "a4349488-9a56-4906-83a5-7d8b33c9d435",
      "label": "NSX61-RC-ManagementCluster"
    },
    "label": "NSX61-RC-ManagementCluster"
  },
  {
    "underlyingValue": {
      "type": "entityRef",
      "componentId": null,
      "classId": "ComputeResource",
      "id": "40b151ce-e409-4d2a-8dae-bb456139a660",
      "label": "NSX61-RC-ComputeClusterB"
    },
    "label": "NSX61-RC-ComputeClusterB"
  },
  {
    "underlyingValue": {
      "type": "entityRef",
      "componentId": null,
      "classId": "ComputeResource",
      "id": "cc254a84-95b8-434a-874d-bdfef8e8ad2c",
      "label": "NSX61-RC-ComputeClusterA"
    },
    "label": "NSX61-RC-ComputeClusterA"
  }]
}
```

Get the Resources Available For a Reservation

You can display information about available resources, such as storage and network information, for a vRealize Automation reservation with the reservation service.

Overview

Most of the available extension fields for a vSphere type reservation are permissible value fields.

All the following field IDs except for `computeResource` and `machineQuota` depend on the `computeResource` field ID. The `computeResource` and `machineQuota` field IDs do not have a dependency on a field ID.

NOTE The information in the table is subject to change. You should call the data and schema service to retrieve the latest field information.

Table 3-17. Extension Fields Supported in vSphere Type Reservation

Field ID	Data Type	Type Class	Permissible Value
reservationNetworks	Complex Type	reservationNetwork	Y
reservationVCNSTransportZone	Entity Reference	NetworkScopes	Y
reservationVCNSSecurityGroups	Entity Reference	SecurityGroups	Y
reservationMemory	Complex Type	reservationMemory	Y
computeResource	Entity Reference	ComputeResource	Y
machineQuota	Integer	N/A	N
reservationStorages	Complex Type	reservationStorage	Y
resourcePool	Entity Reference	ResourcePools	Y
reservationVCNSRoutedGateways	Complex Type	reservationVCNSRoutedGateway	Y

From a permissible value definition, the value list of the fields with permissible values depend on the compute resource field. This example illustrates how to use `resourcePool` to get a permissible value list for the resource pool setting.

For related information, see "Display a Schema Definition for a Reservation Type," on page 138.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- Get the required compute resource ID. See "Get a Compute Resource for the Reservation," on page 148.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/data-service/schema/$schemaclassid/default/$fieldid/values
Method	Post
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.

Input	Description
$token	Specifies a valid HTTP bearer token with necessary credentials.
$schemaclassid	Specifies the schema class ID. This example illustrates how to use the `resourcePool` field of a vSphere reservation type as an example. explain how to get permissible value list for resource pool. The schema class ID of a vSphere reservation is `Infrastructure.Reservation.Virtual.vSphere`. For this example, the input value for *$schemaclassid* is `Infrastructure.Reservation.Virtual.vSphere`.
$fieldId	Specifies the field ID of the resource. For example, the field ID for the resource pool is `resourcePool`. For this example, the input value for *$fieldId* is `resourcePool`.
HTTP body	Contains information about dependencies. Because the dependency of this permissible value field is `computeResource`, you must provide a dependency definition in the HTTP body.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
values	An array of data rows, each of which represents one of the resource pool objects returned in a pageable list. Each resource pool object contains an `underlyingValue` and `label` entry.
underlyingValue	JSON string representing one permissible value for a field: ■ type -- data type of `entityRef`, `complexRef`, or `primary` ■ component ID -- componentID ■ classId -- schema class ID of current data type ■ id -- unique resource pool ID ■ label -- resource pool label
label	Specifies the resource pool label. This value matches the `underlyingValue` value.

Example: `curl` Command

```
curl ––insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/reservation–service/api/data–service/schema/
Infrastructure.Reservation.Virtual.vSphere /default/ resourcePool /values  –d “{
  "text": "",
  "dependencyValues": {
    "entries": [{
      "key": "computeResource",
      "value": {
        "type": "entityRef",
        "componentId": null,
        "classId": "ComputeResource",
        "id": " cc254a84–95b8–434a–874d–bdfef8e8ad2c "
      }
    }]
  }
}”
```

Example: API Explorer

Example: JSON Output

The following JSON output is returned based on your command input.

In the following example output, the `CoreDev` resource pool is shown. Copy the output `underlyingValue` section into an XML editor and use it as input to create or update a reservation. Note that you can also use other REST calls such as `reservationNetworks` and `reservationStorages` to get other resources for the reservation.

```
{
  "values": [{
    "underlyingValue": {
      "type": "entityRef",
      "componentId": null,
      "classId": "ResourcePools",
      "id": " 4e51fabc-19e8-4e79-b413-d52309b3bb62",
      "label": " CoreDev"
    },
    "label": " CoreDev"
  },
  {
    "underlyingValue": {
      "type": "entityRef",
      "componentId": null,
      "classId": "ResourcePools",
      "id": "1186b5cc-cdef-4afb-8653-0ad41a36c194",
      "label": "Documentation"
    },
    "label": "Documentation"
  },
  //Omit other resource pool list
    ]
}
```

Create a Reservation

You can create a new vRealize Automation reservation with the reservation service.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

After you retrieve all permissible value field information, you have the input information required to create a reservation.

For the full list of tasks that you must perform before you create a reservation, see "Creating a Reservation," on page 135.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations
Method	Post
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
HTTP body	The HTTP body describes the reservation to create and calls the REST API used to create the reservation. Compose the HTTP body using one of the following methods: ■ Copy the HTTP body from the JSON output from this example and edit the applicable field values to compose the HTTP body input for the command line. ■ Use the API commands in "Verify a Reservation and Get Reservation Details," on page 158, remove the appropriate ID field from the HTTP response, and edit the field values to compose the HTTP body input for the command line.

Output

The command output contains property names and values based on the API command input parameters.

The output URL contains the new reservation ID.

Property	Description
status	When the reservation is successfully created, the HTTP response status is `201 created`.
Header.Location	The HTTP response contains a `Location` attribute that is formatted as https://*$host* /reservation-service/api/reservations/*$reservationId*.
$reservationId	Specifies the new reservation ID.

Copy the output response into an XML editor for use in a future procedure, such as updating or deleting the reservation.

Example: `curl` Command

The HTTP body is included as part of the command line input.

```
curl ––insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/reservation–service/api/reservations –d
“
{
  "name": "TestCreateReservation",
  "reservationTypeId": "Infrastructure.Reservation.Virtual.vSphere",
  "tenantId": "qe",
  "subTenantId": "ef58f604–528d–4441–a219–4725bead629b",
  "enabled": true,
  "priority": 3,
  "reservationPolicyId": "b71c3a5f–087a–4d9e–9a56–fab785a3d128",
  "alertPolicy": {
    "enabled": true,
    "frequencyReminder": 20,
    "emailBgMgr": false,
    "recipients": ["test1@mycompany.com",
    "test2@mycompany.com"],
```

```
  "alerts": [{
    "alertPercentLevel": 10,
    "referenceResourceId": "storage",
    "id": "storage"
  },
  {
    "alertPercentLevel": 20,
    "referenceResourceId": "memory",
    "id": "memory"
  },
  {
    "alertPercentLevel": 30,
    "referenceResourceId": "cpu",
    "id": "cpu"
  },
  {
    "alertPercentLevel": 40,
    "referenceResourceId": "machine",
    "id": "machine"
  }]
},
"extensionData": {
  "entries": [{
    "key": "reservationNetworks",
    "value": {
      "type": "multiple",
      "elementTypeId": "COMPLEX",
      "items": [{
        "type": "complex",
        "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
        "componentId": null,
        "classId": "reservationNetwork",
        "typeFilter": null,
        "values": {
          "entries": [{
            "key": "reservationNetworkPath",
            "value": {
              "type": "entityRef",
              "componentId": null,
              "classId": "Network",
              "id": "44cb65d5-b321-43dd-a2ab-8ecf387bff8f",
              "label": "VM Network SQA"
            }
          }]
        }
      }]
    }
  },
  {
    "key": "custom-Properties-key0",
    "value": {
      "type": "string",
      "value": "custom-property-value0"
    }
  },
```

```
{
  "key": "custom-Properties-key2",
  "value": {
    "type": "string",
    "value": "custom-property-value2"
  }
},
{
  "key": "reservationMemory",
  "value": {
    "type": "complex",
    "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
    "componentId": null,
    "classId": "reservationMemory",
    "typeFilter": null,
    "values": {
      "entries": [{
        "key": "hostMemoryTotalSizeMB",
        "value": {
          "type": "integer",
          "value": 57187
        }
      },
      {
        "key": "reservationMemoryReservedSizeMb",
        "value": {
          "type": "integer",
          "value": 15872
        }
      }]
    }
  }
},
{
  "key": "computeResource",
  "value": {
    "type": "entityRef",
    "componentId": null,
    "classId": "ComputeResource",
    "id": "cc254a84-95b8-434a-874d-bdfef8e8ad2c",
    "label": "NSX61-RC-ComputeClusterA"
  }
},
{
  "key": "machineQuota",
  "value": {
    "type": "integer",
    "value": 2
  }
},
{
  "key": "reservationStorages",
  "value": {
    "type": "multiple",
    "elementTypeId": "COMPLEX",
```

```
  "items": [{
    "type": "complex",
    "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
    "componentId": null,
    "classId": "reservationStorage",
    "typeFilter": null,
    "values": {
      "entries": [{
        "key": "storageTotalSizeGB",
        "value": {
          "type": "integer",
          "value": 394
        }
      },
      {
        "key": "reservationStorageReservedSizeGB",
        "value": {
          "type": "integer",
          "value": 32
        }
      },
      {
        "key": "reservationStorageEnabled",
        "value": {
          "type": "boolean",
          "value": true
        }
      },
      {
        "key": "reservationStoragePath",
        "value": {
          "type": "entityRef",
          "componentId": null,
          "classId": "StoragePath",
          "id": "f48a527b-30a6-4d54-8829-f549bc195b69",
          "label": "VNXe:qe-vnxe-nfs-1"
        }
      },
      {
        "key": "storageFreeSizeGB",
        "value": {
          "type": "integer",
          "value": 120
        }
      },
      {
        "key": "reservationStorageReservationPriority",
        "value": {
          "type": "integer",
          "value": 1
        }
      }]
    }
  }]
}
```

```
    },
    {
      "key": "resourcePool",
      "value": {
        "type": "entityRef",
        "componentId": null,
        "classId": "ResourcePools",
        "id": "4e51fabc-19e8-4e79-b413-d52309b3bb62",
        "label": "CoreDev"
      }
    }]
  }
}
"
```

Example: API Explorer

Example: JSON Output

The following JSON output is returned based on your command input.

The following example output contains the HTTP body and a location URL. The output URL contains the new reservation ID, for example `94d74105-831a-4598-8f42-efd590fea15c`.

```
Location:
https://$host/reservation-service/api/reservations/94d74105-831a-4598-8f42-efd590fea15c
```

Copy the location URL from this output and save it, for example for updating or deleting the reservation later.

Verify a Reservation and Get Reservation Details

After you create a vRealize Automation reservation, you can use the reservation ID to verify that the reservation exists. You can also use the ID to get information about the reservation in preparation for updating or deleting it.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Finish creating a new reservation. You can obtain the reservation ID from the output URL. See "Create a Reservation," on page 153.
- Get the reservation ID if you do not already know it. See "Display a List of Reservations," on page 163.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations/$reservationId This is the URL that is generated when you create a reservation using the REST API. See "Create a Reservation," on page 153.
Method	Get

Input	Description
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$reservationId	Specifies the unique identifier of the reservation to verify. See "Create a Reservation," on page 153.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
status	The HTTP response status is `201 created` to indicate that the reservation exists.
Header.Location	The HTTP response should contain a `location` attribute, format as https://*$host* /reservation-service/api/reservations/*$reservationId*.
$reservationId	The HTTP response should contain a location attribute, formatted as https://*$host* /reservation-service/api/reservations/*$reservationI*d.

Example: `curl` Command

In the following example, the reservation ID of 94d74105-831a-4598-8f42-efd590fea15c is the value you obtained when you created the reservation. See "Create a Reservation," on page 153.

```
curl ––insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/reservation–service/api/reservations/94d74105–831a–4598–8f42–efd590fea15c
```

Example: API Explorer

Example: JSON Output

The following JSON output is returned based on your command input.

Copy the output response into an XML editor for future step usage.


```
{
  "id": "94d74105–831a–4598–8f42–efd590fea15c ",
  "name": "TestReservation",
  "reservationTypeId": "Infrastructure.Reservation.Virtual.vSphere",
  "tenantId": "qe",
  "subTenantId": "ef58f604–528d–4441–a219–4725bead629b",
  "enabled": true,
  "priority": 3,
  "reservationPolicyId": "b71c3a5f–087a–4d9e–9a56–fab785a3d128",
  "alertPolicy": {
    "enabled": true,
    "frequencyReminder": 20,
    "emailBgMgr": false,
    "recipients": ["user1@mycompany.com",
    "user2@mycompany.com"],
    "alerts": [{
      "alertPercentLevel": 10,
      "referenceResourceId": "storage",
      "id": "storage"
    },
    {
```

```
      "alertPercentLevel": 20,
      "referenceResourceId": "memory",
      "id": "memory"
    },
    {
      "alertPercentLevel": 30,
      "referenceResourceId": "cpu",
      "id": "cpu"
    },
    {
      "alertPercentLevel": 40,
      "referenceResourceId": "machine",
      "id": "machine"
    }]
  },
  "extensionData": {
    "entries": [{
      "key": "key4",
      "value": {
        "type": "string",
        "value": "custom-property-value4"
      }
    },
    {
      "key": "key3",
      "value": {
        "type": "string",
        "value": "custom-property-value3"
      }
    },
    {
      "key": "reservationNetworks",
      "value": {
        "type": "multiple",
        "elementTypeId": "COMPLEX",
        "items": [{
          "type": "complex",
          "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
          "componentId": null,
          "classId": "reservationNetwork",
          "typeFilter": null,
          "values": {
            "entries": [{
              "key": "reservationNetworkProfile",
              "value": {
                "type": "entityRef",
                "componentId": null,
                "classId": "NetworkProfile",
                "id": "ed5d1503-08ac-42ca-804d-9167834a63a5",
                "label": "ETEDoNotDelete2014-10-13 13:10:56"
              }
            },
            {
              "key": "reservationNetworkPath",
              "value": {
```

```
            "type": "entityRef",
            "componentId": null,
            "classId": "Network",
            "id": "44cb65d5-b321-43dd-a2ab-8ecf387bff8f",
            "label": "VM Network SQA"
          }
        }]
      }
    }]
  }
},
{
  "key": "key0",
  "value": {
    "type": "string",
    "value": "custom-property-value0"
  }
},
{
  "key": "key2",
  "value": {
    "type": "string",
    "value": "custom-property-value2"
  }
},
{
  "key": "reservationMemory",
  "value": {
    "type": "complex",
    "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
    "componentId": null,
    "classId": "reservationMemory",
    "typeFilter": null,
    "values": {
      "entries": [{
        "key": "hostMemoryTotalSizeMB",
        "value": {
          "type": "integer",
          "value": 57187
        }
      },
      {
        "key": "reservationMemoryReservedSizeMb",
        "value": {
          "type": "integer",
          "value": 15888
        }
      }]
    }
  }
},
{
  "key": "key1",
  "value": {
    "type": "string",
```

```
      "value": "custom-property-value-Updated"
    }
  },
  {
    "key": "computeResource",
    "value": {
      "type": "entityRef",
      "componentId": null,
      "classId": "ComputeResource",
      "id": "047e00f5-5424-4ed2-a751-4a334aeaff54",
      "label": "VC51-Cluster"
    }
  },
  {
    "key": "machineQuota",
    "value": {
      "type": "integer",
      "value": 2
    }
  },
  {
    "key": "reservationStorages",
    "value": {
      "type": "multiple",
      "elementTypeId": "COMPLEX",
      "items": [{
        "type": "complex",
        "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
        "componentId": null,
        "classId": "reservationStorage",
        "typeFilter": null,
        "values": {
          "entries": [{
            "key": "storageTotalSizeGB",
            "value": {
              "type": "integer",
              "value": 394
            }
          },
          {
            "key": "reservationStorageReservedSizeGB",
            "value": {
              "type": "integer",
              "value": 31
            }
          },
          {
            "key": "reservationStorageEnabled",
            "value": {
              "type": "boolean",
              "value": true
            }
          },
          {
            "key": "reservationStoragePath",
```

```
              "value": {
                "type": "entityRef",
                "componentId": null,
                "classId": "StoragePath",
                "id": "f48a527b-30a6-4d54-8829-f549bc195b69",
                "label": "VNXe:qe-vnxe-nfs-1"
              }
            },
            {
              "key": "storageFreeSizeGB",
              "value": {
                "type": "integer",
                "value": 120
              }
            },
            {
              "key": "reservationStorageReservationPriority",
              "value": {
                "type": "integer",
                "value": 1
              }
            }]
          }
        }]
      }
    },
    {
      "key": "resourcePool",
      "value": {
        "type": "entityRef",
        "componentId": null,
        "classId": "ResourcePools",
        "id": "4e51fabc-19e8-4e79-b413-d52309b3bb62",
        "label": "CoreDev"
      }
    }]
  }
}
```


Display a List of Reservations

You can obtain and display a list of existing vRealize Automation reservations to obtain the required reservation ID value in preparation for updating or deleting a reservation.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations
Method	Get
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Species an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL that produces the result.
Content	Specifies an array of data rows, each of which represents one of the tenant objects returned in a pageable list.
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: `curl` Command

```
curl –-insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/reservation–service/api/reservations
```

Example: API Explorer

Example: JSON Output

The following JSON output is returned based on your command input.

The following example output lists two reservations, named `MyTestReservation1` and `MyTestReservation2`. For related information, see "Verify a Reservation and Get Reservation Details," on page 158.

You can use the `id` value for each reservation to update or delete them. For related information, see "Update a Reservation," on page 169 or "Delete a Reservation," on page 174.


```
{
  "links": [],
  "content": [{
  "id": "94d74105–831a–4598–8f42–efd590fea15c ",
  "name": "TestReservation",
  "reservationTypeId": "Infrastructure.Reservation.Virtual.vSphere",
```

```
"tenantId": "qe",
"subTenantId": "ef58f604-528d-4441-a219-4725bead629b",
"enabled": true,
"priority": 3,
"reservationPolicyId": "b71c3a5f-087a-4d9e-9a56-fab785a3d128",
"alertPolicy": {
  "enabled": true,
  "frequencyReminder": 20,
  "emailBgMgr": false,
  "recipients": ["user1@mycompany.com",
  "user2@mycompany.com"],
  "alerts": [{
    "alertPercentLevel": 10,
    "referenceResourceId": "storage",
    "id": "storage"
  },
  {
    "alertPercentLevel": 20,
    "referenceResourceId": "memory",
    "id": "memory"
  },
  {
    "alertPercentLevel": 30,
    "referenceResourceId": "cpu",
    "id": "cpu"
  },
  {
    "alertPercentLevel": 40,
    "referenceResourceId": "machine",
    "id": "machine"
  }]
},
"extensionData": {
  "entries": [{
    "key": "key4",
    "value": {
      "type": "string",
      "value": "custom-property-value4"
    }
  },
  {
    "key": "key3",
    "value": {
      "type": "string",
      "value": "custom-property-value3"
    }
  },
  {
    "key": "reservationNetworks",
    "value": {
      "type": "multiple",
      "elementTypeId": "COMPLEX",
      "items": [{
        "type": "complex",
        "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
```

```
        "componentId": null,
        "classId": "reservationNetwork",
        "typeFilter": null,
        "values": {
          "entries": [{
            "key": "reservationNetworkProfile",
            "value": {
              "type": "entityRef",
              "componentId": null,
              "classId": "NetworkProfile",
              "id": "ed5d1503-08ac-42ca-804d-9167834a63a5",
              "label": "ETEDoNotDelete2014-10-13 13:10:56"
            }
          },
          {
            "key": "reservationNetworkPath",
            "value": {
              "type": "entityRef",
              "componentId": null,
              "classId": "Network",
              "id": "44cb65d5-b321-43dd-a2ab-8ecf387bff8f",
              "label": "VM Network SQA"
            }
          }]
        }
      }]
    }
  },
  {
    "key": "key0",
    "value": {
      "type": "string",
      "value": "custom-property-value0"
    }
  },
  {
    "key": "key2",
    "value": {
      "type": "string",
      "value": "custom-property-value2"
    }
  },
  {
    "key": "reservationMemory",
    "value": {
      "type": "complex",
      "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
      "componentId": null,
      "classId": "reservationMemory",
      "typeFilter": null,
      "values": {
        "entries": [{
          "key": "hostMemoryTotalSizeMB",
          "value": {
            "type": "integer",
```

```
          "value": 57187
        }
      },
      {
        "key": "reservationMemoryReservedSizeMb",
        "value": {
          "type": "integer",
          "value": 15888
        }
      }]
    }
  }
},
{
  "key": "key1",
  "value": {
    "type": "string",
    "value": "custom-property-value-Updated"
  }
},
{
  "key": "computeResource",
  "value": {
    "type": "entityRef",
    "componentId": null,
    "classId": "ComputeResource",
    "id": "047e00f5-5424-4ed2-a751-4a334aeaff54",
    "label": "VC51-Cluster"
  }
},
{
  "key": "machineQuota",
  "value": {
    "type": "integer",
    "value": 2
  }
},
{
  "key": "reservationStorages",
  "value": {
    "type": "multiple",
    "elementTypeId": "COMPLEX",
    "items": [{
      "type": "complex",
      "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
      "componentId": null,
      "classId": "reservationStorage",
      "typeFilter": null,
      "values": {
        "entries": [{
          "key": "storageTotalSizeGB",
          "value": {
            "type": "integer",
            "value": 394
          }
```

```
            },
            {
              "key": "reservationStorageReservedSizeGB",
              "value": {
                "type": "integer",
                "value": 31
              }
            },
            {
              "key": "reservationStorageEnabled",
              "value": {
                "type": "boolean",
                "value": true
              }
            },
            {
              "key": "reservationStoragePath",
              "value": {
                "type": "entityRef",
                "componentId": null,
                "classId": "StoragePath",
                "id": "f48a527b-30a6-4d54-8829-f549bc195b69",
                "label": "VNXe:qe-vnxe-nfs-1"
              }
            },
            {
              "key": "storageFreeSizeGB",
              "value": {
                "type": "integer",
                "value": 120
              }
            },
            {
              "key": "reservationStorageReservationPriority",
              "value": {
                "type": "integer",
                "value": 1
              }
            }]
          }
        }]
      }
    },
    {
      "key": "resourcePool",
      "value": {
        "type": "entityRef",
        "componentId": null,
        "classId": "ResourcePools",
        "id": "4e51fabc-19e8-4e79-b413-d52309b3bb62",
        "label": "CoreDev"
      }
    }],
  "metadata": {
    "size": 0,
```

```
    "totalElements": 1,
    "totalPages": 1,
    "number": 1,
    "offset": 0
  }
}
```

Update a Reservation

You can update an existing vRealize Automation reservation with the reservation service.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the reservation ID of the reservation that you want to update. This information is required API command input. See "Display a List of Reservations," on page 163.
- Obtain the reservation field information for the reservation that you want to update. For example, if you want to change from one compute resource to another, you must obtain the new compute resource ID and its associated JSON section output. This information is required API command input. For example, to change the compute resource for a particular reservation, see "Get a Compute Resource for the Reservation," on page 148.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations/$reservationId
Method	Put
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$reservationId	Specifies the unique identifier of the reservation to update. For information about how to obtain the reservation ID, see "Display a List of Reservations," on page 163.
HTTP body	Contains the JSON information for the reservation, including the updated data for the parameters that you want to update. Most of this JSON string information is obtained by displaying the existing details of the *$reservationId*. See "Verify a Reservation and Get Reservation Details," on page 158. The rest of the JSON string information is obtained by using an API command to get the ID of the parameter you want to update. For example, to update the reservation to use a different compute resource than the one currently specified, replace the `computeResource` value of the exiting reservation with a new `computeResource` value in the command's HTTP input.

Output

If the command is successful, the HTTP response body is empty except for a `204 No Content` status statement.

Example: `curl` Command

You can use the following command to update the reservation ID **`94d74105-831a-4598-8f42-efd590fea15c`** to use compute resource ID 047e00f5-5424-4ed2-a751-4a334aeaff54.

```
curl -X PUT--insecure -H "Accept:application/json"
-H "Authorization: Bearer $token"
https://$host/reservation-service/api/reservations/94d74105-831a-4598-8f42-efd590fea15c -d
“
{
  "id": "94d74105-831a-4598-8f42-efd590fea15c",
  "name": "TestReservation",
  "reservationTypeId": "Infrastructure.Reservation.Virtual.vSphere",
  "tenantId": "qe",
  "subTenantId": "ef58f604-528d-4441-a219-4725bead629b",
  "enabled": true,
  "priority": 3,
  "reservationPolicyId": "b71c3a5f-087a-4d9e-9a56-fab785a3d128",
  "alertPolicy": {
    "enabled": true,
    "frequencyReminder": 20,
    "emailBgMgr": false,
    "recipients": ["user1@mycompany.com",
    "user2@mycompany.com"],
    "alerts": [{
      "alertPercentLevel": 10,
      "referenceResourceId": "storage",
      "id": "storage"
    },
    {
      "alertPercentLevel": 20,
      "referenceResourceId": "memory",
      "id": "memory"
    },
    {
      "alertPercentLevel": 30,
      "referenceResourceId": "cpu",
      "id": "cpu"
    },
    {
      "alertPercentLevel": 40,
      "referenceResourceId": "machine",
      "id": "machine"
    }]
  },
  "extensionData": {
    "entries": [{
      "key": "key4",
      "value": {
        "type": "string",
```

```
      "value": "custom-property-value4"
    }
  },
  {
    "key": "key3",
    "value": {
      "type": "string",
      "value": "custom-property-value3"
    }
  },
  {
    "key": "reservationNetworks",
    "value": {
      "type": "multiple",
      "elementTypeId": "COMPLEX",
      "items": [{
        "type": "complex",
        "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
        "componentId": null,
        "classId": "reservationNetwork",
        "typeFilter": null,
        "values": {
          "entries": [{
            "key": "reservationNetworkProfile",
            "value": {
              "type": "entityRef",
              "componentId": null,
              "classId": "NetworkProfile",
              "id": "ed5d1503-08ac-42ca-804d-9167834a63a5",
              "label": "TestNetworkProfile"
            }
          },
          {
            "key": "reservationNetworkPath",
            "value": {
              "type": "entityRef",
              "componentId": null,
              "classId": "Network",
              "id": "44cb65d5-b321-43dd-a2ab-8ecf387bff8f",
              "label": "VM Network SQA"
            }
          }]
        }
      }]
    }
  },
  {
    "key": "key0",
    "value": {
      "type": "string",
      "value": "custom-property-value0"
    }
  },
  {
    "key": "key2",
```

```
      "value": {
        "type": "string",
        "value": "custom-property-value2"
      }
    },
    {
      "key": "reservationMemory",
      "value": {
        "type": "complex",
        "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
        "componentId": null,
        "classId": "reservationMemory",

        "typeFilter": null,
        "values": {
          "entries": [{
            "key": "hostMemoryTotalSizeMB",
            "value": {
              "type": "integer",
              "value": 57187
            }
          },
          {
            "key": "reservationMemoryReservedSizeMb",
            "value": {
              "type": "integer",
              "value": 15888
            }
          }]
        }
      }
    },
    {
      "key": "key1",
      "value": {
        "type": "string",
        "value": "custom-property-value-Updated"
      }
    },
    {
      "key": "computeResource",
      "value": {
        "type": "entityRef",
        "componentId": null,
        "classId": "ComputeResource",
        "id": "047e00f5-5424-4ed2-a751-4a334aeaff54",
        "label": "VC51-Cluster"
      }
    },
    {
      "key": "machineQuota",
      "value": {
        "type": "integer",
        "value": 2
      }
```

```
},
{
  "key": "reservationStorages",
  "value": {
    "type": "multiple",
    "elementTypeId": "COMPLEX",
    "items": [{
      "type": "complex",
      "componentTypeId": "com.mycompany.csp.iaas.blueprint.service",
      "componentId": null,
      "classId": "reservationStorage",
      "typeFilter": null,
      "values": {
        "entries": [{
          "key": "storageTotalSizeGB",
          "value": {
            "type": "integer",
            "value": 394
          }
        },
        {
          "key": "reservationStorageReservedSizeGB",
          "value": {
            "type": "integer",
            "value": 31
          }
        },
        {
          "key": "reservationStorageEnabled",
          "value": {
            "type": "boolean",
            "value": true
          }
        },
        {
          "key": "reservationStoragePath",
          "value": {
            "type": "entityRef",
            "componentId": null,
            "classId": "StoragePath",
            "id": "f48a527b-30a6-4d54-8829-f549bc195b69",
            "label": "VNXe:qe-vnxe-nfs-1"
          }
        },
        {
          "key": "storageFreeSizeGB",
          "value": {
            "type": "integer",
            "value": 120
          }
        },
        {
          "key": "reservationStorageReservationPriority",
          "value": {
            "type": "integer",
```

```
                "value": 1
              }
            }]
          }
        }]
      }
    },
    {
      "key": "resourcePool",
      "value": {
        "type": "entityRef",
        "componentId": null,
        "classId": "ResourcePools",
        "id": "4e51fabc-19e8-4e79-b413-d52309b3bb62",
        "label": "CoreDev"
      }
    }]
  }
}
"
```

Example: API Explorer

Example: JSON Output

If the command is successful, the HTTP response body is empty except for a `204 No Content` status statement.

Delete a Reservation

You can delete an existing vRealize Automation reservation with the reservation service.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the reservation ID of the reservation that you want to delete. This information is required API command input. See "Display a List of Reservations," on page 163.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations/$reservationId
Method	Delete
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.

Input	Description
$token	Specifies a valid HTTP bearer token with necessary credentials.
$reservationId	Specifies the unique identifier of the reservation to delete. For information about how to obtain the reservation ID, see "Display a List of Reservations," on page 163.

Output

If the command is successful, the HTTP response body is empty except for a `204 No Content` status statement.

Example: `curl` Command

```
curl -X "Delete" --insecure -H "Accept:application/json"
-H "Authorization: Bearer $token"
https://$host/reservation-service/api/reservations/94d74105-831a-4598-8f42-efd590fea15c
```

Example: API Explorer

Example: JSON Output

If the command is successful, the HTTP response body is empty except for a `204 No Content` status statement.

Working with Reservation Policy

You can work with the reservation service to perform a variety of functions, such as creating and updating reservation policies.

Display a List of Reservation Policies

You can obtain and display a list of existing vRealize Automation reservation policies to obtain the required reservation policy ID in preparation for updating or deleting a reservation policy.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- "Display a List of Reservation Policies," on page 175

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations/policies
Method	Get

Input	Description
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Property	Description
Links	Species an array of link objects, each of which contains the following parts:
rel	Specifies the name of the link. ■ `Self` refers to the object which was returned or requested. ■ `First`, `Previous`, `Next`, and `Last` refer to corresponding pages of pageable lists. ■ Specifies the application or service that determines the other names.
href	Specifies the URL that produces the result.
Content	Specifies an array of data rows, each of which represents one of the tenant objects returned in a pageable list. Each tenant object contains the following information:
@type	Contains the `ReservationPolicy` string.
Id	Specifies the unique reservation policy ID.
name	Specifies the reservation policy name.
description	Specifies the reservation policy description.
reservationPolicyTypeId	Specifies the type of reservation policy. Supported vRealize Automation reservation policy types are Reservation.Policy.ComputeResource and Reservation.Policy.Storage.
Metadata	Specifies the paging-related data.
Size	Specifies the maximum number of rows per page.
totalElements	Specifies the number of rows returned.
totalPages	Specifies the total number of pages of data available.
Number	Specifies the current page number.
Offset	Specifies the number of rows skipped.

Example: `curl` Command

```
curl ––insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/reservation–service/api/reservations/policies
```

Example: API Explorer

Example: JSON Output

The following JSON output is returned based on your command input.

The following example output lists two reservation policies, named `reservationPolicyTest` and `reservationPolicyTest2`. You can use the `id` value for each reservation policy to update or delete them. For related information, see "Update a Reservation Policy," on page 180and "Delete a Reservation Policy," on page 181.


```
{
  "links": [],
  "content": [{
    "@type": "ReservationPolicy",
    "id": "8adafb54-4c85-4478-86f0-b6ae80ab5ca4",
    "name": "reservationPolicyTest",
    "description": "reservationPolicyDescTest",
    "reservationPolicyTypeId": "Reservation.Policy.ComputeResource"
  },
  {
    "@type": "reservationPolicy",
    "id": "fdd9854b-012e-41d7-ad17-fc73d4395714",
    "name": "reservationPolicyTest2",
    "description": "reservationPolicyDescTest2",
    "reservationPolicyTypeId": "Reservation.Policy.Storage"
  }],
  "metadata": {
    "size": 0,
    "totalElements": 2,
    "totalPages": 1,
    "number": 1,
    "offset": 0
  }
}
```


Create a Reservation Policy

You can create a vRealize Automation reservation policy using the REST API.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations/policies
Method	Post
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Input	Description
HTTP body	Describes the reservation policy to create. ■ *$name* - reservation policy name ■ *$description* - reservation policy description
$reservationPolicyTypeId	Specifies the reservation policy type ID. The supported reservation policy types are **`Reservation.Policy.ComputeResource`** and **`Reservation.Policy.Storage`**.

Output

The command output contains property names and values based on the API command input parameters.

The output URL contains the new reservation policy ID.

Property	Description
status	When the reservation policy is successfully created, the HTTP response status is `201 created`.
Header.Location	The HTTP response contains a `Location` attribute that is format as https://$host /reservation-service/api/reservations/policies/*$reservationPolicyId*.
$reservationPolicyId	Specifies the new reservation policy ID.

Copy the output response into an XML editor for use in a future procedure, such as updating or deleting the reservation policy.

Example: `curl` Command

You can use the following command to create a new reservation policy.

```
curl ––insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/reservation–service/api/reservations/policies –d “
{
  "name": "ABXReservationPolicyTest",
  "description": "ABXReservationPolicyDescTest",
  "reservationPolicyTypeId": "Reservation.Policy.ComputeResource"
}
“
```

Example: API Explorer

Example: JSON Output

The following JSON output is returned based on your command input.

The following example output contains the HTTP body and a location URL. The output URL contains the new reservation policy ID, for example `5fd2de36–659f–4beb–97af–77d683feb697`.

```
Location:
https://$host/reservation–service/api/reservations/policies/5fd2de36–659f–4beb–97af–77d683feb697
```

You can copy the location URL from this output to an XML editor for future use, for example for updating or deleting the reservation policy later.

Display a Reservation Policy by ID

You can get information about vRealize Automation reservation policy based on its ID using the REST API.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Create a reservation policy. See "Create a Reservation Policy," on page 177.
- Obtain the reservation policy ID of the reservation policy that you want to query. See "Display a List of Reservation Policies," on page 175.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations/policies/$id
Method	Get
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.

Output

The command output contains property names and values based on the API command input parameters.

Parameter	Description
$id	Specifies the reservation policy ID.
$name	Specifies the reservation policy name.
$description	Specifies the reservation policy description.
$reservationPolicyTypeId	Specifies the reservation policy type.

Example: `curl` Command

```
curl --insecure -H "Accept:application/json"
-H "Authorization: Bearer $token"
https://$host/reservation-service/api/reservations/policies/8adafb54-4c85-4478-86f0-b6ae80ab5ca4
```

Example: API Explorer

Example: JSON Output

The following JSON output is returned based on your command input.

The following sample output displays information for the specified reservation policy ID **8adafb54-4c85-4478-86f0-b6ae80ab5ca4**.

```
{
    "id": "8adafb54-4c85-4478-86f0-b6ae80ab5ca4",
    "name": "reservationPolicyTest",
    "description": "reservationPolicyDescTest",
    "reservationPolicyTypeId": "Reservation.Policy.ComputeResource"
}
```

Copy the response body into an XML editor if you want to update the reservation policy.

Update a Reservation Policy

You can update a vRealize Automation reservation policy using the REST API.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the reservation policy ID of the reservation policy that you want to update. See "Display a List of Reservation Policies," on page 175.
- Query the reservation policy and copy the response output to an XML editor for use as the basis of your command input for this task. See "Display a Reservation Policy by ID," on page 179.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations/policies/*$id*
Method	Put
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
HTTP body	Describes the reservation policy to update. ■ *$name* - reservation policy name ■ *$description* - reservation policy description
$reservationPolicyTypeId	Specifies the reservation policy type ID. The supported reservation policy types are **`Reservation.Policy.ComputeResource`** and **`Reservation.Policy.Storage`**.

Output

If the command is successful, the HTTP response body is empty except for a `204 No Content` status statement.

Example: `curl` Command

You can use the following example to update the name and description values for the reservation ID 94d74105-831a-4598-8f42-efd590fea15c.

```
curl -X PUT --insecure -H "Accept:application/json"
-H "Authorization: Bearer $token"
https://$host/reservation-service/api/reservations/policies/94d74105-831a-4598-8f42-efd590fea15c
-d '
{
  "name": "ReservationPolicyTestRename",
  "description": "ReservationPolicyDescTestRename",
  "reservationPolicyTypeId": "Reservation.Policy.ComputeResource"
}
'
```

Example: API Explorer

Example: JSON Output

If the command is successful, the HTTP response body is empty except for a `204 No Content` status statement.

Delete a Reservation Policy

You can delete a vRealize Automation reservation policy using the REST API.

Prerequisites

- Log in to vRealize Automation as a **fabric group administrator**.
- Verify that you have the host name and fully qualified domain name of the vRealize Automation instance.
- If you are not using the API Explorer, verify that you have a valid HTTP bearer token that matches your login credentials.
- Obtain the reservation policy ID of the reservation policy that you want to delete. See "Display a List of Reservation Policies," on page 175.

Input

You can use supported input parameters to control the command output.

Input	Description
URL	https://*$host*/reservation-service/api/reservations/policies/*$id*
Method	Delete
$host	Specifies the host name and fully qualified domain name or IP address of the vRealize Automation identity server.
$token	Specifies a valid HTTP bearer token with necessary credentials.
$id	Specifies the reservation policy ID.

Output

If the command is successful, the HTTP response body is empty except for a `204 No Content` status statement.

Example: `curl` Command

You can use the following example command to delete the publication ID 8adafb54-4c85-4478-86f0-b6ae80ab5ca4.

```
curl -X “Delete” ––insecure –H "Accept:application/json"
–H "Authorization: Bearer $token"
https://$host/reservation–service/api/reservations/policies/8adafb54–4c85–4478–86f0–b6ae80ab5ca4
```

Example: API Explorer

Example: JSON Output

If the command is successful, the HTTP response body is empty except for a `204 No Content` status statement.

Using the API Explorer 4

The API Explorer is a command line interface that you can use to explore the REST API services test methods

This chapter includes the following topics:

Install the API Explorer

You can download the REST API Explorer from the vRealize Appliance management console and install it on your machine.

You can run `java -version` in a UNIX shell or Windows Command Prompt window to verify the version.

You can request verbose help for a specific command with `help` *command_name*.

Prerequisites

- Verify that your machine has Java SE Development Kit (JDK) 7 installed and running.
- Verify your PATH environment variable includes the location of the correct version of Java.

Procedure

1 Open a Web browser.

2 Navigate to the vRealize Appliance management console by using its fully qualified domain name, `https://vra-va-hostname.domain.name:5480`.

3 Download the REST API Explorer (vcac-cli) distribution package.

4 Unzip the distribution package to a local folder.

The local folder now contains the `bin`, `repo`, and `etc` folders.

5 (UNIX only) Use `chmod` to grant execute privileges to the `vcac-cli` script.

```
%chmod +x bin/vcac-cli
```

6 (UNIX only) Determine which version of java the script uses.

```
%sh -x bin/vcac-cli
```

7 Update your PATH environment variable to include the location of the `bin` folder.

What to do next

Log in with the API Explorer

You can log in securely to a vRealize Automation server with the API Explorer.

When running a script in UNIX, you can prevent the plaintext password from appearing in the `spring-shell.log` file by storing the password in a file and redirecting standard input from that file.

Prerequisites

- "Install the API Explorer," on page 183, if necessary.
- Your PATH environment variable must contain the location of the `vcac-cli` (UNIX) or `vcac_cli.bat` (Windows) script.

Procedure

1 Enter the `login` command string in a Command Prompt window.

Option	Description
vcac_url	vRealize Automation URL
username@fqdn	vRealize Automation username with fully qualified domain name
tname	Tenant name

```
login --url vcac_url --user username@fqdn --tenant tname
```

2 Enter the password, when prompted.

3 If running in script mode, enter the password by redirecting standard input.

A successful login returns the `vcac-cli>` prompt without an error message.

If you omit the `--tenant` option, the command logs you in to the default tenant.

An error returns an explicit message. The following are possible error messages.

Error Message	Reason
Command failed java.lang.RuntimeException: java.net.UnknownHostException: vcac152-009-067a.eng.vmware.com	The hostname specified in the --url parameter is not a known host name. Check your spelling.
Command failed com.vmware.vcac.authentication.sts.AuthenticationFailedException: com.vmware.vim.sso.client.exception.AuthenticationFailedException: Provided credentials are not valid.	The specified -user or -password value(s) are incorrect. Check spelling, check to make sure you have specified the correct tenant.
Command failed org.springframework.web.client.HttpClientErrorException: 400 Bad Request	The specified -tenant value is unknown. Check your spelling.

Example: Logging in

Log in to API Explorer.

```
Welcome to vCAC CLI. For assistance press or type "hint" then hit ENTER.
vcac-cli>login --url https://vcac110-062-143.eng.vmware.com --user administrator@vsphere.local
Please enter your password: ******
```

The password is masked for security purposes and does not appear in `spring-shell.log`.

```
vcac-cli>login --url https://vcac148-084-173.eng.vmware.com
--user administrator@vsphere.local < /tmp/password.txt
```

Suppress Log Files

The API Explorer updates the spring-shell history file `spring-shell.log` and the message file `vcac-cli.log` in the `vcac-cli_install/bin` folder by default. You can suppress output to both of these files.

Prerequisites

- "Install the API Explorer," on page 183, if necessary.
- Your PATH environment variable must contain the location of the `vcac-cli` (UNIX) or `vcac_cli.bat` (Windows) script.

Procedure

- Run the following command.

  ```
  $ vcac-cli --profiles nologging
  ```

Choosing Your Mode of Operation

The REST API Explorer has three modes of operation to accommodate new and experienced users.

- Use the Interactive Mode on page 185

  The easiest way to use and learn the vRealize Automation API Explorer is with the interactive mode.
- Use the Command Line Mode on page 187

  The command line mode lets you incorporate vcac-cli commands in other scripts and programs.
- Use the Script Mode on page 188

  The script mode is similar to the command line mode, except that you can invoke multiple commands in sequence.

Use the Interactive Mode

The easiest way to use and learn the vRealize Automation API Explorer is with the interactive mode.

Every command you type in interactive mode is appended to the `spring-shell.log` file in your current folder. This file retains a history of commands you have issued. It also serves as an example of what a CLI Script File looks like. While using this mode, you can navigate your command history by pressing the up-arrow and down-arrow keys on your keyboard.

The vcac-cli supports tab auto-completion and context-sensitive help on both Windows and UNIX. For example, if you enter **`rest g`** and then press the Tab key, the command expands to `rest get`. If you press the Tab key again, vcac-cli displays all options for the command. See the following examples.

```
vcac-cli>rest
Rest delete    rest get    rest post    rest put

vcac-cli>rest get --
Rest get --service    rest get --u    rest get --uri

vcac-cli>rest get --f
rest get --f    rest get --format

vcac-cli>rest get --format
rest get --format
optional - format: format (JSON, table raw, …); default: 'JSON'

vcac-cli>rest get --format
JSON    compactTable    raw    solidBorderTable    table
```

Prerequisites

- "Install the API Explorer," on page 183, if necessary.
- Your PATH environment variable must contain the location of the `vcac-cli` (UNIX) or `vcac_cli.bat` (Windows) script.

Procedure

1 (UNIX) In a shell, enter**`vcac-cli`** or **`sh bin/vcac-cli`**.

2 (Windows) In a Command Prompt window, enter **`vcac-cli.bat`**.

The vcac-cli banner appears.

3 (Optional) Enter the `help` command for a list of supported commands.

```
vcac-cli>help
* ! - Allows execution of operating system (OS) commands.
* */ - End of block comment
* /* - Start of block comment
* // - Inline comment markers (start of line only)
* ; - Inline comment markers (start of line only)
* date - Displays the local date and time
* exit - Exits the shell
* help - list all commands usage
* login - Open a secure session to a VCAC server
* output - Set the command output parameters
* quit - Exits the shell
* rest delete - Invoke a DELETE http request
* rest get - Invoke a GET http request
* rest post - Invoke a POST http request
* rest put - Invoke a PUT http request
* script - Parses the specified resource file and executes its commands
* services - Displays a list of available services
* system properties - Shows the shell's properties
vcac-cli>
```

4 (Optional) Enter `help` *command_name* for verbose help on the command.

```
vcac-cli>help rest get
Keyword:                  rest get
Description:              Invoke a GET http request
 Keyword:                 s
 Keyword:                 sessionid
   Help:                  Session identifier
   Mandatory:             false
   Default if specified:  '__NULL__'
   Default if unspecified: '__NULL__'

 Keyword:                 ** default **
 Keyword:                 service
   Help:                  Name of the Service hosting the URI. e.g. catalog-service
   Mandatory:             true
   Default if specified:  '__NULL__'
   Default if unspecified: '__NULL__'

 Keyword:                 u
 Keyword:                 uri
   Help:                  URI of resource. e.g. consumer/catalogItems
   Mandatory:             true
   Default if specified:  '__NULL__'
   Default if unspecified: '__NULL__'

 Keyword:                 h
 Keyword:                 headers
   Help:                  Show request and response headers
   Mandatory:             false
   Default if specified:  'true'
   Default if unspecified: 'false'

* rest get - Invoke a GET http request
```

Use the Command Line Mode

The command line mode lets you incorporate vcac-cli commands in other scripts and programs.

You can invoke any supported vcac-cli command and option, including help.

Prerequisites

- "Install the API Explorer," on page 183, if necessary.
- Your PATH environment variable must contain the location of the `vcac-cli` (UNIX) or `vcac_cli.bat` (Windows) script.

Procedure

1 Enter the command string on the vcac-cli command line.

```
$ vcac-cli command_string
```

The output is displayed on the stderr stream.

2 (Optional) You can redirect the output to a file in Linux or Windows.

```
$ vcac-cli system properties 2> output.txt
```

Example: Run vcac-cli commands in the command line

```
$ vcac-cli system properties
app.home = /Users/myusername/vcac/cli/shell/target/appassembler
app.name = vcac-cli
app.pid = 12444
app.repo = /Users/myusername/vcac/cli/shell/target/appassembler/repo
. . .
```

Use the Script Mode

The script mode is similar to the command line mode, except that you can invoke multiple commands in sequence.

In script mode, you must first create a text file which contains a series of vcac-cli interactive-mode commands. Vcac-cli executes the commands in sequence.

Prerequisites

- "Install the API Explorer," on page 183, if necessary.
- Your PATH environment variable must contain the location of the `vcac-cli` (UNIX) or `vcac_cli.bat` (Windows) script.

Procedure

1 Create a text file containing a series of vcac-cli interactive-mode commands.

For example, enter the folllowing commands in a file named `script.txt`.

```
login --url https://vcac152-009-067.eng.vmware.com --user tanteater@example.com --password
password --tenant MYCOMPANY
rest get --service workitem-service --u workitems
```

2 Run the script and redirect the output.

```
$ vcac-cli script script.txt 2> script.out
```

Filtering and Formatting Your Output 5

You can control your JSON output display using command line and third-party filtering and formatting options.

To simplify your JSON output, consider using command line options to filter out unnecessary data and display only the information that you are interested in, such as the following information categories:

- Published catalog items
- Request status
- Provisioned machine identifiers

You can also use command line options or JSON formatting tools, such as Open Data Protocol (OData), to control the JSON results of your API command line inputs.

This chapter includes the following topics:

Using a JSON Command Line Format and Validation Tool

Working with JSON directly can introduce errors. An extra semicolon or bracket can cause a call to return an exception.

You can reduce command line errors significantly if you use a JSON formatter to validate the JSON data and present it in an easy-to-read format.

A JSON formatter that keeps a history of your submissions is an especially useful feature for you debugging JSON errors.

Filtering JSON Output with Command Line Options

You can use filters in your command line to limit JSON output to specific conditions.

For example, you can use a filter in a catalog item request to display only catalog items that contain a specific catalog ID. For details, see "Find a Catalog Item by Name," on page 36.

Formatting Your JSON Output

You can format XML and JSON output with several formatting tools, including JSON command line parameters and ODATA formatting calls.

You can specify formatting options to be performed on that JSON output file, either in the command console or in a JSON file that you create using a command line redirect.

Common formatting tools include ODATA and JSON command line options. For example, you can use the requestID resource call to format the output of a command that displays request status. You can also use an ODATA equivalent to format that same information

Helpers for Pretty-Printing XML/JSON

- alias ppxml='python -c "import sys, xml.dom.minidom; print xml.dom.minidom.parseString(sys.stdin.read()).toprettyxml()"'
- alias ppjson='python -mjson.tool'

Related Tools 6

In addition to the API Explorer and API Services documentation, and the API explorer and curl commands described for the various use cases, you can expand your options using related tools.

You can use third party tools and the vRealize Automation REST API services reference to further expand your programming options.

For more information, see *Using Application Services REST APIs*.

This chapter includes the following topics:

Using the REST API Services Reference Documentation

The *REST API Reference* documentation describes the available API services and their resources, data elements, and data types that you can use to manage items in the service catalog programmatically and remotely.

You can perform various vRealize Automation functions programmatically using REST API service calls.

To use the API service reference documentation effectively, you must know which service and resource to use. See Chapter 1, "Overview of the vRealize Automation REST API," on page 7 for a complete list of services and their descriptions. If you need more information, click a link for a detailed description of a service and a list of the tasks you can perform with it.

For a more terse description of all the available API services, see the *REST API Reference*, which is a collection of zipped resource files located on the VMware vRealize Automation Documentation page at https://www.vmware.com/support/pubs/vcac-pubs.html.

After you download the `.zip` file from the VMware vRealize Automation Documentation page, unzip it and use the `index.html` file to view the API service topics.

Using the Chrome Developer Tools or Firebug

You can use the Chrome Developer Tools or Firebug to reveal the data you need to construct a request for some of the supported service calls by using the vRealize Automation REST API.

You can adapt these steps to perform a different action, such as adding a tenant.

Prerequisites

- Open a Chrome browser session and log in to the vRealize Automation console as a business group user with access to catalog items.
- Open a command prompt or a shell and log in to the vRealize Automation command line interface.

Procedure

1 Click the **Catalog** tab in the vRealize Automation console.

2 Click the catalog Item you want to request.

3 Enter the request information for the catalog item, but do not submit your changes.

4 Press the Ctrl-Shift-I keys simultaneously to open the Chrome Developer Tools.

5 Click the **Network** tab.

6 Click **Record Network Log**.

7 Click **Submit** in the console.

8 Verify that the network logs contain the relevant data.

   a Locate a `makeRequest POST` in the network recordings.

   b Click **makeRequest POST** to view its details.

   c Scroll to view the `Form Data url` and `postData` sections.

   The `url` section shows the vRealize Automation service and URI for you to use. This example uses the catalog-service, under the uri consumer/requests.

   The `postData` section shows the JSON data passed in the HTTP POST call. You can insert the JSON data in a JSON text file, for example, `request.json`, and submit it with the POST method in the command line.

   NOTE Click **Clear** to purge the network logs if they become too large to navigate easily.

9 Enter the following call in the vRealize Automation CLI window, where the `request.json` text file contains the JSON data from the postData section.

```
rest post --headers --service catalog-service --uri consumer/requests --data request.json
```

   This call makes the same request that was submitted with the console.

What to do next

See "Using a JSON Command Line Format and Validation Tool," on page 189 if you encounter errors with the JSON data you submitted.

Index

W