

Scenario Based Workflows

Created by Shruthi Subramanian, last modified by Suruchi Suruchi on Jan 03, 2023

This page will give you an insight into the Automation Toolkit Workflows that must be followed for flawless execution and usage of the toolkit.



NOTE:

- Automation Tool Kit **DOES NOT** support the **creation/export** of duplicate resources.
- DO NOT modify/remove** any commented rows or column names. You may **re-arrange** the columns if needed (**except NSGs**).
- A double colon (::) or Semi-Colon (;) has a special meaning in the Tool Kit. Do **not** use them in the OCI data / values.
- Do **not** include any **unwanted space** in **cells** you fill in; do not place any **empty rows** in between.
- To learn about how to add attributes, refer [Support for additional attributes \(Flat TF Files\)](#)
- Any entry made/moved post **<END>** in any of the Tabs, of CD3 will not be processed. Any resources created by the automation & then moved after the **<END>** will cause the resources to be removed.
- The components that get created as part of VCNs Tab (Example: IGW, SGW, LPG, NGW, DRG) will have the same set of Tags attached to them.
- Automation Tool Kit does not support sharing of Block Volumes.
- Detail on the know behaviour of the toolkit can be found at [Known Behaviour Of Automation Toolkit](#)
- Option to Modify Network -
Some points to consider while modifying networking components are -
 - Converting the exported VCN to Hub/Spoke/Peer VCN is not allowed. Route Table rules based on the peering for new LPGs to existing VCNs will not be auto populated. Users are requested to add an entry to the RouteRulesInOCI sheet to support the peering rules.
 - Adding a new VCN as Hub and other new VCNs as Spoke/Peer is allowed. Gateways will be created as specified in VCNs sheet.
 - Adding new VCNs as None is allowed. Gateways will be created as specified in VCNs sheet.
 - The addition of new Subnets to exported VCNs and new VCNs is allowed.

Scenarios	Execution Steps
Validate CD3	<p>Steps:</p> <ol style="list-style-type: none"> Modify the setUpOCI.properties file to set non_gf_tenancy to "false". Execute setUpOCI.py and choose the service from the menu options. Choose "Validate CD3" Choose the options from sub-menu. Make sure to have created Identity (terraform apply) before you validate Tags, Network, C LoadBalancers and Management Services. <p>Expected ERROR: Compartment Network does not exist in OCI.</p> <p>→ These error mean that the component is not found in OCI. So, please make sure to create the Compartment "Network" before</p>
Greenfield - Create new components	<p>Steps:</p> <ol style="list-style-type: none"> Modify the setUpOCI.properties file to set non_gf_tenancy to "false". Execute setUpOCI.py and choose the service from the menu options.
NonGreenfield/Export existing components from tenancy	<p>Steps:</p> <ol style="list-style-type: none"> Modify the setUpOCI.properties file to set non_gf_tenancy to "true". Execute setUpOCI.py and choose the service from the menu options.
Greenfield - Create Identity components	<p>Steps:</p> <ol style="list-style-type: none"> Modify the setUpOCI.properties file to set non_gf_tenancy to "false". Execute setUpOCI.py and choose the service from the menu options → Choose "Identity" → Choose "Create Compartment" Change directory to that of outdir and execute terraform init/plan/apply. Change your directory to /cd3user/tenancies/<prefix>, cat cmds.log. Copy and execute the command for fetch compartments Execute setUpOCI.py and choose the service from the menu options → Choose "Identity" → Choose "Create Policies" Execute Terraform Plan and Terraform Apply in outdir.
Use an existing DRG in OCI while creating the network	<p>Steps:</p> <ol style="list-style-type: none"> Modify the setUpOCI.properties file to set non_gf_tenancy to "false". Execute setUpOCI.py and choose the service from the menu options → Choose "Network" → Choose "Create Network" Change directory to that of outdir and execute terraform init Import the DRG into your state file. Execute <pre>terraform import "module.drags[\\"<<drags terraform variable name>>\"] .oci_core_drg.drg" <<drg ocid>></pre> <ol style="list-style-type: none"> Execute Terraform Plan and Terraform Apply in outdir. Terraform Plan will indicate to add all the other
Add a new Network Component - VCN/DHCP/DRG/Subnet	<p>Steps:</p> <ol style="list-style-type: none"> Modify the setUpOCI.properties file to set non_gf_tenancy to "false". Add the data to the appropriate Excel sheets. Execute setUpOCI.py and choose "Network". Choose sub-menu → "Modify Network". <p>Note: Make sure to export Sec Rules, Route Rules, DRG Route Rules (Option 3,4 of sub-menu) to CD3 Excel Sheet before executing</p>
Add a new Compute VM to an existing Infrastructure (such as Identity, Network) in OCI	<p>Steps:</p> <ol style="list-style-type: none"> Modify the setUpOCI.properties file to set non_gf_tenancy to "true". Execute setUpOCI.py and choose "Export Network" from the menu options. Execute the generated shell script to sync the Terraform with existing network components of OCI. Add the new VM details to the Excel sheet. Modify the setUpOCI.properties file to set non_gf_tenancy to "false". Execute setUpOCI.py and choose "Network" and then "Add/Modify/Delete Instances/Boot Backup Policy" from the menu Execute Terraform Plan and Terraform Apply in outdir.
To add a new Route Table/Update existing Route Rules/Add new Route Rules/Delete existing Route Rules/Delete existing Route Table/ Add new Security List/ Update existing Security Rule/ Add new Security Rules/Delete existing Security Rules	<p>Steps:</p> <ol style="list-style-type: none"> Modify the setUpOCI.properties file to set non_gf_tenancy to "false". Add/Update the data to the appropriate Excel sheets. (RouteRulesInOCI, SecRulesInOCI, DRGRouteRulesInOCI). Execute setUpOCI.py and choose "Network". Choose sub-menu options "Modify SecRules"/ "Modify RouteRules"/ "Modify DRG RouteRules" or a combination of the requirement.

Scenarios	Execution Steps
Note: This will create TF for only those Security Lists and Route Tables in VCNs which are part of cd3 and skip any VCNs that have been created outside of cd3 execution.	
Add new components after export of non-greenfield tenancy - For Networking - VCN/DHCP/DRG/IGW/NGW/SGW/LPG/Subnet	Steps: 1. Modify the setUpOCI.properties file to set non_gf_tenancy to "false" . 2. Add the data to the appropriate Excel sheets. 3. Execute setUpOCI.py and choose "Network" . 4. Choose sub-menu option "Modify Network" .
To Export Instances in batches using different filters	Steps: 1. Modify the setUpOCI.properties file to set non_gf_tenancy to "true" . 2. Choose "Export Compute" . 3. Specify the prefix of the instances to export or specify the AD to export. 4. Once the execution completes, take a backup of the files generated for instances in out directory(* _instances.tfvars and tf_import_cmds_instances_nonGF.sh) and a backup of the 'Instances' tab of the Input CD3 Excel Sheet. 5. Repeat steps from 1 to 4 to export next set of Instances using another filter. 6. Once you export all the required instances using multiple filters, move the files from backup to the out directory and then exec generated for Instances. Consolidate the data of exported instances from the Excel sheet backups.

Sample of CD3 after export:

(DO NOT Modify the highlighted columns)

VCNs tab:

Region	Compartment Name	VCN Name	CIDR Block	DRG Required	IGW Required	NGW Required	SGW Required	LPG Required	Hub/Spoke/P
Phoenix	OMCDev-VM	ocs-vcn	10.100.0.0/16	n	igw	n	n	n	exported
Phoenix	OMCDev-VM	oke-vcn-quick-DevOps-Default-158338dec	10.0.0.0/16	oke-drg	oke-igw-quick-DevOps-Default-158338dec	oke-ngw-quick-DevOps-Default-158338dec	n	n	exported
Phoenix	OMCDev-VM	dcs-slave-vcn	10.0.0.0/16	n	Internet Gateway for dcs-slave-vcn	n	n	dcs-slave-vcn_lpg0	exported

Subnets tab:

Region	Compartment Name	VCN Name	Subnet Name	CIDR Block	Availability Domain(AD1 AD2 AD3 Regional)	Type(private public)	DHCP Option Name	Route Table Name	Seclist Names	Add Default Seclist	Configure SGW Route(n object_storage all_services)	Configure NGW Route(y n)	Configure IGW Route(y n)
Phoenix	OMCDev-VM	ocs-vcn	subnet2	10.100.3.0/24	AD3	private		n	n	y	-	-	-
Phoenix	OMCDev-VM	ocs-vcn	subnet	10.100.2.0/24	AD1	public		n	n	y	-	-	-
Phoenix	OMCDev-VM	ocs-vcn	ocs_subnet	10.100.10.0/24	Regional	public		n	n	y	-	-	-
Phoenix	OMCDev-VM	oke-vcn-quick-DevOps-Default-158338dec	oke-subnet-quick-DevOps-Default-158338dec-regional	10.0.10.0/24	Regional	private		oke-wkr-routetable-DevOps-Default-158338dec	oke-wkr-secist-quick-DevOps-Default-158338dec	n	-	-	-
Phoenix	OMCDev-VM	dcs-slave-vcn	shrsubr-oke-svcbsubnet-quick-DevOps-Default-158338dec-regional	10.0.20.0/24	Regional	public		oke-lb-routetable-DevOps-Default-158338dec	oke-lb-secist-quick-DevOps-Default-158338dec	n	-	-	-
Phoenix	OMCDev-VM	dcs-slave-vcn	dcsl-slave-subnet-wdWU:PHX-AD-3	10.0.2.0/24	AD3	public		n	n	y	-	-	-
Phoenix	OMCDev-VM	dcs-slave-vcn	dcsl-slave-subnet-wdWU:PHX-AD-2	10.0.1.0/24	AD2	public		n	n	y	-	-	-

No labels