

Prerequisites

- Basic knowledge of Oracle Cloud Infrastructure and Terraform
- Access to an Oracle Cloud Infrastructure tenancy with required IAM policies to create network resources
- Appropriate Policies for an OKE cluster
- A virtual cloud network (VCN) to launch a Linux instance in OCI
- Sufficient Quota on the below resources :
 - Compute instance
 - Block volume (If you intend to create Kubernetes persistent volumes)
 - Load balancer (If you intend to create a load balancer to distribute traffic between the nodes)

11

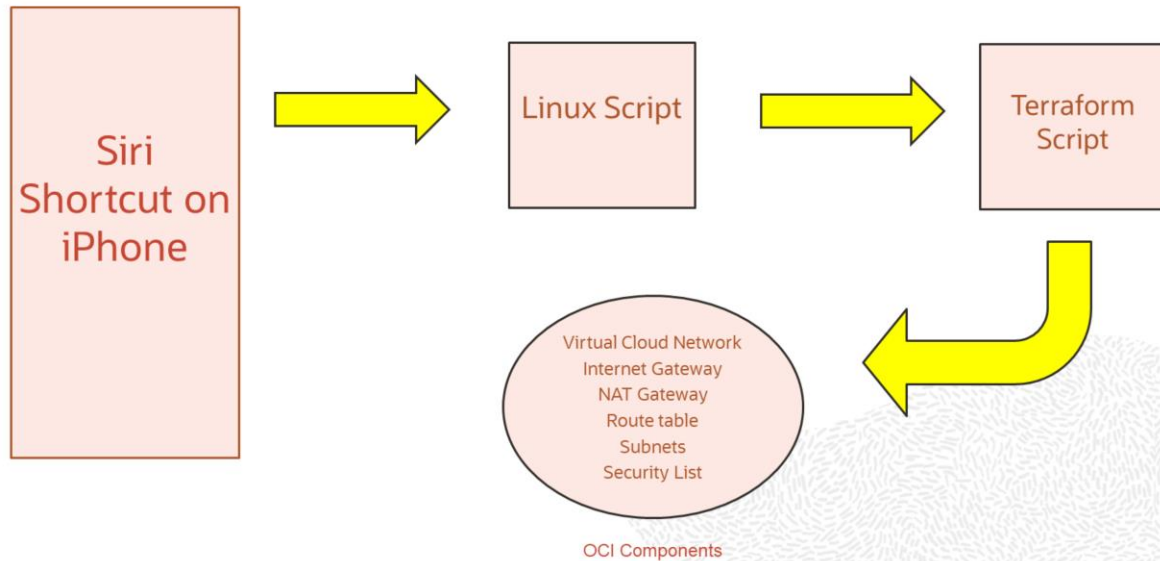


Before, creating a cluster using Siri, consider the above Prerequisites.

IAM Policies are required to create below network resources :

- VCN, subnets
- Internet gateway
- Route table
- Security lists

High level Flow



12

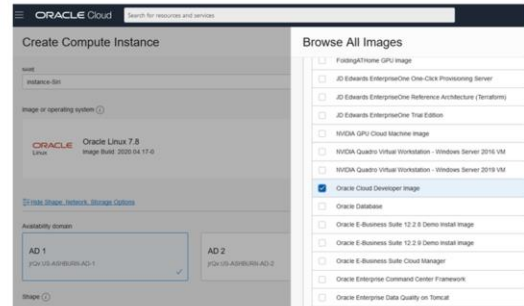


Below is the flow to create a cluster using Siri :

- Ask Siri to create a cluster and provide a name for the cluster.
- Siri shortcut calls the Linux script that in turn calls the Terraform script.
- The Terraform script creates the below Network Components and then creates the cluster :
 - Virtual Cloud Network
 - Internet Gateway
 - NAT Gateway
 - Route table
 - Subnets
 - Security List

High level steps : Step 1

- Provision a Linux Instance (VM) with public IP on OCI to host Terraform Scripts
- You may select Oracle Developer image which comes pre-installed with many useful tools.
 - You will be required to specify a SSH key to ssh into the Linux Instance.
 - The same key will be used later in the iOS Shortcut.



13

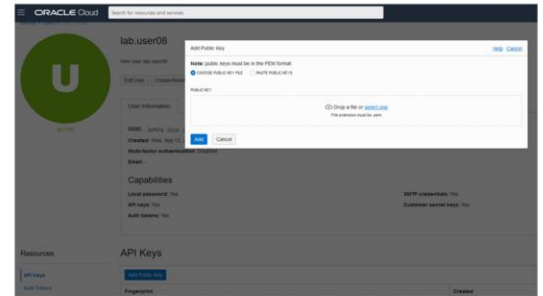
OCI Compute lets you provision and manage compute hosts, known as instances . You can launch instances as needed to meet your compute and application requirements. After you launch an instance, you can access it securely from your computer, restart it, attach and detach volumes, and terminate it when you're done with it.

Oracle Cloud Infrastructure offers both bare metal and virtual machine instances.

Follow instruction here - [Launching Your First Linux Instance](#) for launching Linux instance.

High level steps : Step 2

- Prepare Linux Instance
 - Generate and upload required OCI API Signing key
 - Public key should be uploaded for the OCI user via Console and private key should be available on the Linux Instance
 - Download or update (for Oracle Developer's Image) Terraform on OCI VM



14

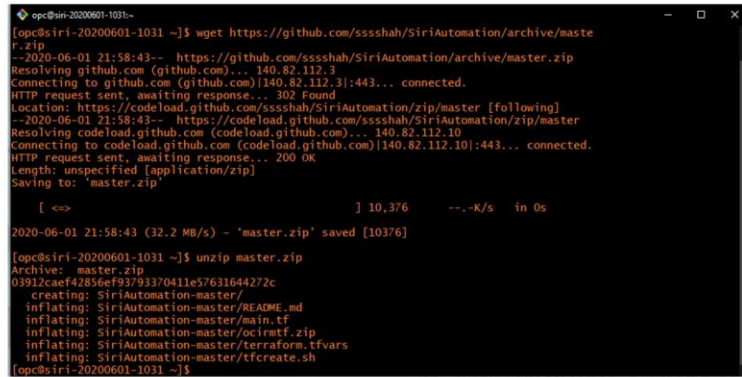
Your API requests will be signed with your private key, and Oracle will use the public key to verify the authenticity of the request. You must upload the public key for the OCI User via Console.

Refer here to generate API key : [How to generate an API Signing Key](#)

Download Terraform : [Download Terraform by HashiCorp](#)

High level steps : Steps 3

- Download scripts from GitHub to your Linux Instance.
- GitHub Link : <https://github.com/sssshah/SiriAutomation>



```
opc@siri-20200601-1031 ~]$ wget https://github.com/sssshah/SiriAutomation/archive/master.zip
--2020-06-01 21:58:43-- https://github.com/sssshah/SiriAutomation/archive/master.zip
Resolving github.com (github.com)... 140.82.112.3
Connecting to github.com (github.com):140.82.112.3:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/sssshah/SiriAutomation/zip/master [following]
--2020-06-01 21:58:43-- https://codeload.github.com/sssshah/SiriAutomation/zip/master
Resolving codeload.github.com (codeload.github.com)... 140.82.112.10
Connecting to codeload.github.com (codeload.github.com):140.82.112.10:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: 'master.zip'

[ <> ] 10.376 --.-K/s in 0s

2020-06-01 21:58:43 (32.2 MB/s) - 'master.zip' saved [10376]

opc@siri-20200601-1031 ~]$ unzip master.zip
Archive: master.zip
03912caef42856ef93793370411e57631644272c
  creating: SiriAutomation-master/
  inflating: SiriAutomation-master/README.md
  inflating: SiriAutomation-master/main.tf
  inflating: SiriAutomation-master/ocirmtf.zip
  inflating: SiriAutomation-master/terraform.tfvars
  inflating: SiriAutomation-master/tfcreate.sh
opc@siri-20200601-1031 ~]$
```

15



Run the below command to download the scripts from Github :

- `wget https://github.com/sssshah/SiriAutomation/archive/master.zip`
- `unzip master.zip`

The master.zip folder contains the following files :

- `tfcreate.sh` called from iPhone shortcut
- `main.tf` Terraform script that creates Oracle Cloud Infrastructure Objects and Kubernetes cluster
- `terraform.tfvars` Terraform variables file
- `ocirmtf.zip` used with the Oracle Resource Manager (Not meant to be used with Siri)

High level steps : Step 4

- Update *terraform.tfvars* downloaded from GitHub with your tenancy information and private API Signing key
 - Update Tenancy, Compartment and User OCID, fingerprint, private key path and region
 - The private API signing key should already be uploaded to the Linux Instance
 - Update main.tf for different subnet names, CIDR block etc. Alternatively, use terraform.tfvars to customize.

[illegible]

The file terraform.tfvars is the Terraform Variables file and contains the below variables: tenancy_ocid, compartment_ocid, user_ocid, fingerprint, private_key_path and region.

Updating the file `main.tf` is an optional step.

To retrieve the Tenancy and User OCID, refer here : [Tenancy's OCID and User's OCID](#)

High level steps : Step 5

- Test the terraform script from Linux Instance
 - Run tfcreate.sh script
 - Delete the cluster once the test is successful

```
opc@siri-20200601-1031:~/SiriAutomation-master  
[opc@siri-20200601-1031 SiriAutomation-master]$ ./tfcreate.sh my-cluster
```

```
opc@siri-20200601-1031:~/SiriAutomation-master  
[opc@siri-20200601-1031 SiriAutomation-master]$ terraform destroy
```

This is an optional step, just to confirm the terraform script works fine without any errors. In order to run tfcreate.sh, you may need to change the file permissions.

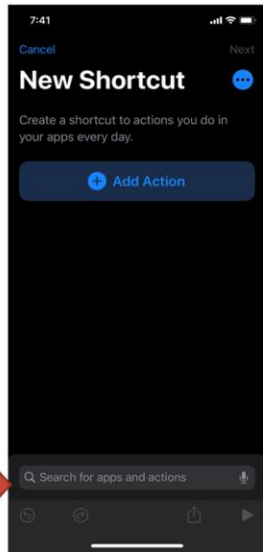
High level steps : Step 6

- Configure iPhone Shortcut to run the Terraform script on the Linux Instance
 1. Create Siri Shortcut
 2. Add Actions to Shortcut
 3. Configure script to run when called from Siri
 4. Add SSH key on Linux instance
 5. Specify script location and input parameters

Step 6 is to create and configure the Siri Shortcut. Let us go through each of these steps in detail.

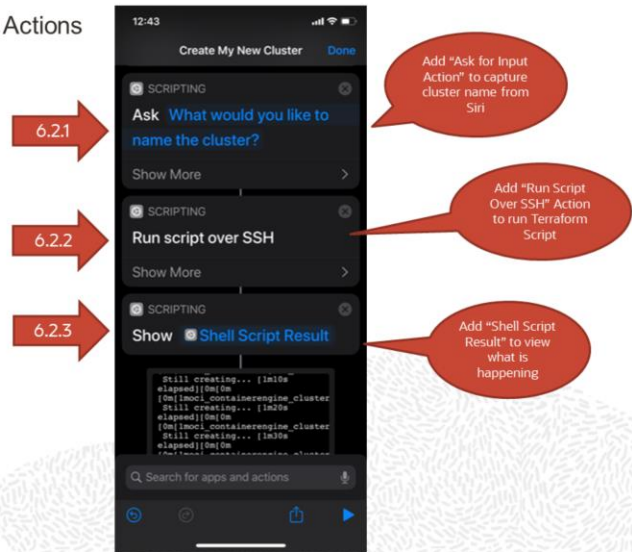
High level steps : Step 6

Step 6.1 – Create Shortcut



You can search
for Actions here

Step 6.2- Add Actions

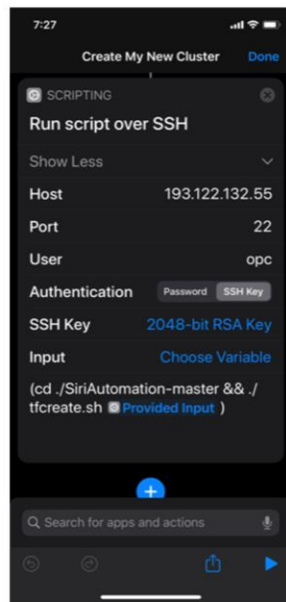


Step 6.1 : Create a new Siri Shortcut

Step 6.2 : Click on Add Action

- Step 6.2.1 : Add "Ask for Input Action"
- Step 6.2.2 : Add "Run Script over SSH"
- Step 6.2.3 : Add "Shell Script Result"

High level steps : Step 6



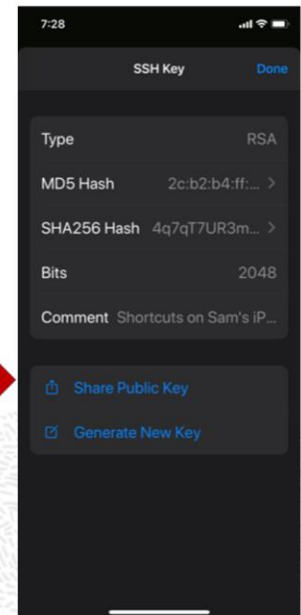
Step 6.3 : Configure script to run when called from Siri

Location of Linux Instance
Or host (public IP)

6.3.1

6.3.2

Select SSH key Option and
Touch the SSH Key



Touch Share
Public Key to
copy the Public
Key via text,
email or other
ways

6.3.3

20

Step 6.3 : Configure script to run when called from Siri

- Step 6.3.1 : Provide the Host address or Public IP of Linux instance
- Step 6.3.2 : Select the SSH Key for Authentication
- Step 6.3.3 : Select Share Public Key

High level steps : Step 6



Add SSH key copied from iOS to `authorized_keys` file on your Linux Instance

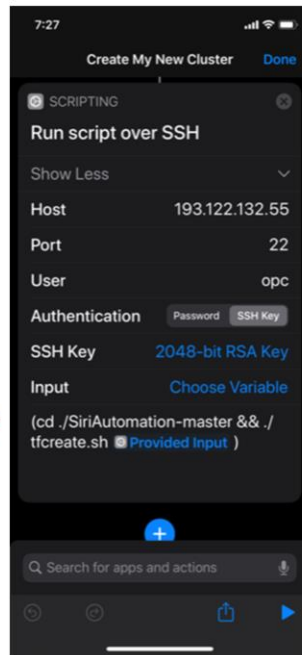
```
opc@siri-20200601-1031:~/ssh
[opc@siri-20200601-1031 ~]$ cd .ssh
[opc@siri-20200601-1031 .ssh]$ ls -al
total 12
drwx-----, 2 opc opc 55 Jun 1 22:44 .
drwx-----, 12 opc opc 4096 Jun 2 13:40 ..
-rw-----, 1 opc opc 400 Jun 1 14:34 authorized_keys
-rw-r--r--, 1 opc opc 1679 Jun 1 22:44 rn_oci_api_key.pem
[opc@siri-20200601-1031 .ssh]$
```



Step 6.4 : From your iOS device copy the key to your SSH terminal window. Copy the SSH key to `authorized_keys` file on your Linux Instance.

High level steps : Step 6

Specify location of Script
& Input parameters to be
passed from Siri



* You may get an iOS error when you run it with Siri voice activation: "the authenticity of host can't be established because it has not been seen before by this device". Run it as by clicking the blue array first before trying the Siri Voice command so the host is now trusted.



22



Step 6.5.1 : Provide the path to the *tfcreate.sh* script and the input parameters.

Before trying out with Siri, try with the blue array icon first. Now, the host is trusted and you can continue testing with Siri.

The *tfcreate.sh* script calls *main.tf*, which creates all the required OCI Resources.

High level steps : Step 7

- Run Shortcut from the iPhone – “Hey Siri!, *create my new cluster*”
 - where <create my new cluster> is the name of the shortcut on iOS you created earlier
 - Siri will ask you for the cluster name. Reply Siri with a name you want to assign to your cluster.
 - Within a few minutes, an OKE cluster is created and is Active !!!

23

The final step is to talk to Siri and ask for a cluster to be created. Siri will respond back and ask you for the cluster name. Provide a name for the OKE cluster and relax! The cluster will be provisioned within a couple of minutes.