



BOSCH

Invented for life

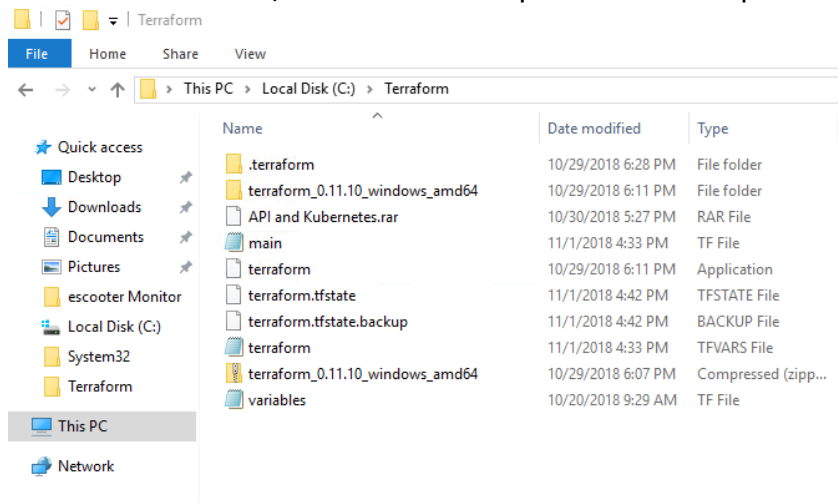
Terraform Deployment Steps

inside.Docupedia Export

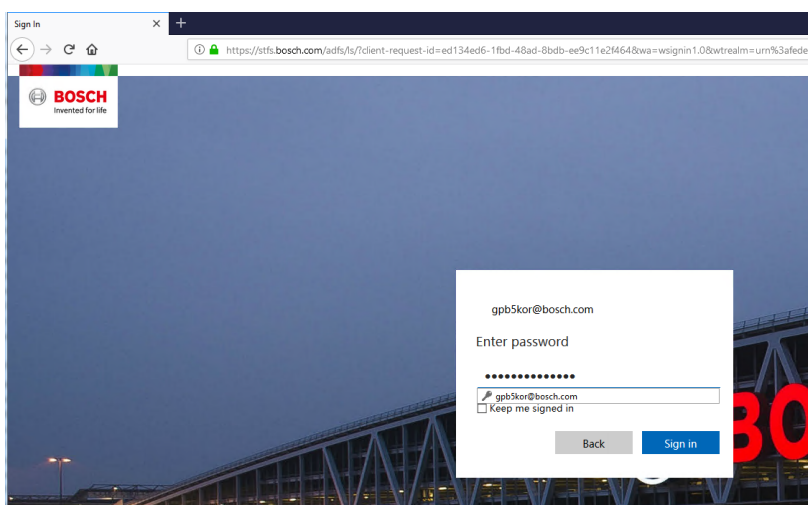
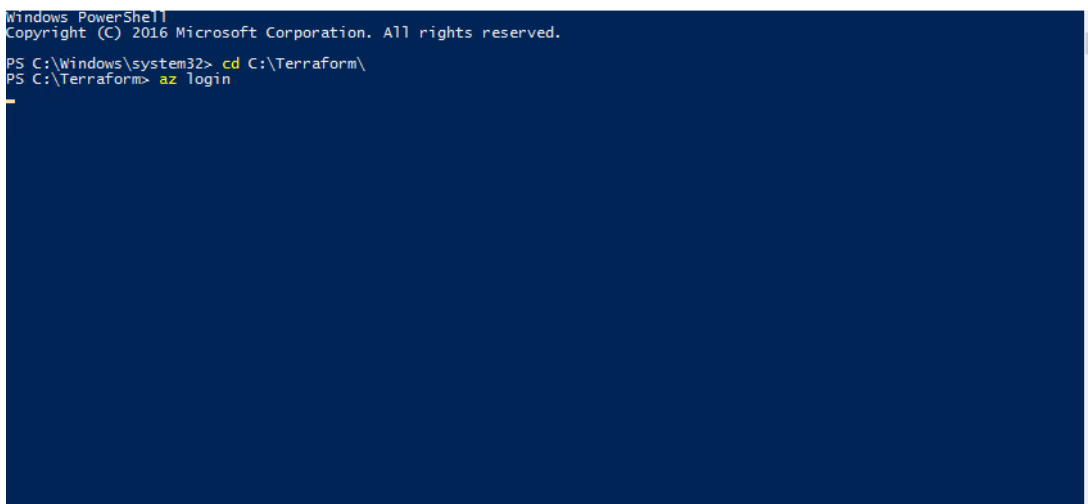
Author: Shanmugapriya Manickaraj (RBEI/BSF6)
Date: 17-Jul-2019 12:27

Table of Contents

1. Download the Terraform exe file from the below link
https://releases.hashicorp.com/terraform/0.11.11/terraform_0.11.11_windows_amd64.zip
2. Create a folder in C:\Terraform and unzip the terraform zip folder.



3. Copy the Terraform files "main.tf", "terraform.tfvars" and "variables.tf" into C:\Terraform folder
4. Open the Powershell and navigate to the folder **C:\Terraform**
5. Login into azure portal using username and password using the powershell command **"az login"**



6. After login you will get list of subscription

```

PS C:\Terraform> az login
Note, we have launched a browser for you to login. For old experience with device code, use "az login --use-device-code"
You have logged in. Now let us find all the subscriptions to which you have access...
[
  {
    "cloudName": "AzureCloud",
    "id": "175b8af5-87d3-4414-a238-afd138e952d6",
    "isDefault": false,
    "name": "AED-ES-ENG-PROD",
    "state": "Enabled",
    "tenantId": "0ae51e19-07c8-4e4b-bb6d-648ee58410f4",
    "user": {
      "name": "gpb5kor@bosch.com",
      "type": "user"
    }
  },
  {
    "cloudName": "AzureCloud",
    "id": "bddd05d-2806-4a0d-9c3f-ccfeecf8d9c9",
    "isDefault": false,
    "name": "AA-ICD-IL-TrainingPortal-Prod",
    "state": "Enabled",
    "tenantId": "0ae51e19-07c8-4e4b-bb6d-648ee58410f4",
    "user": {
      "name": "gpb5kor@bosch.com",
      "type": "user"
    }
  },
  {
    "cloudName": "AzureCloud",
    "id": "bdfa39c4-0fe2-40de-82a7-554e3e6f3cda",
    "isDefault": false,
    "name": "AED-ES-ENG-DEV",
    "state": "Enabled",
    "tenantId": "0ae51e19-07c8-4e4b-bb6d-648ee58410f4",
    "user": {
      "name": "gpb5kor@bosch.com",
      "type": "user"
    }
  },
  {
    "cloudName": "AzureCloud",
    "id": "0d3a3a2b-2635-4743-84b7-aaa5e69fe5ba",
    "isDefault": false,
    "name": "Visual Studio Enterprise",
    "state": "Enabled",
    "tenantId": "0ae51e19-07c8-4e4b-bb6d-648ee58410f4",
    "user": {
      "name": "gpb5kor@bosch.com",
      "type": "user"
    }
  }
]

```

7. Choose the subscription which you want to use

```

PS C:\Terraform> az account set --subscription "0d3a3a2b-2635-4743-84b7-aaa5e69fe5ba"
PS C:\Terraform>

```

8 Type the command **"/Terraform refresh"** The command is used to reconcile the state Terraform knows about (via its state file) with the real-world infrastructure. This can be used to detect any drift from the last-known state, and to update the state file.

```

PS C:\Terraform> .\terraform.exe refresh
Empty or non-existent state file.

Refresh will do nothing. Refresh does not error or return an erroneous
exit status because many automation scripts use refresh, plan, then apply
and may not have a state file yet for the first run.

```

9. Then Type the command **"/Terraform plan"** The command is used to create an execution plan. Terraform performs a refresh, unless explicitly disabled, and then determines what actions are necessary to achieve the desired state specified in the configuration files.

```

PS C:\Terraform> .\terraform.exe plan
Refreshing Terraform state in-memory prior to plan...
The refreshed state will be used to calculate this plan, but will not be
persisted to local or remote state storage.

-----
An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
  + Create

Terraform will perform the following actions:

+ azure_monitor_metric_alert.Kubernetes_Network_Unavailable
  id: <computed>
  action.#: "1"
  action.435204557.action_group_id: "/subscriptions/bdfa29c4-0fe2-40de-82a7-554e3e6f3cda/resourceGroups/dev-escooter-
-alert/providers/Microsoft.Insights/actionGroups/dev-escooter-emailalert"
  auto_mitigate: "false"
  criteria.#: "1"
  criteria.0.aggregation: "Average"
  criteria.0.dimension.#: "1"
  criteria.0.dimension.0.name: "condition"
  criteria.0.dimension.0.operator: "Include"
  criteria.0.dimension.0.values.#: "1"
  criteria.0.dimension.0.values.0: "NetworkUnavailable"
  criteria.0.metric_name: "kube_node_status_condition"
  criteria.0.metric_namespace: "Microsoft.ContainerService/managedClusters"
  criteria.0.operator: "GreaterThan"
  criteria.0.threshold: "0"
  description: "Threshold - Network Unavailable for one or more node"
  enabled: "true"
  frequency: "PT1M"
  name: "qa-Alert-Kubernetes_Service-Network_Unavailable"
  resource_group_name: "qa-escooter-solution"
  scopes.#: "1"
  scopes.1584454502: "/subscriptions/bcb0bb98-5e20-45cd-a907-c679150b2175/resourcegroups/qa-escooter-
solution/providers/Microsoft.ContainerService/managedClusters/qa-escooter-cluster"
  severity: "1"
  tags.%: <computed>
  window_size: "PT5M"

Plan: 1 to add, 0 to change, 0 to destroy.

```

10. Then type the command **"/Terraform apply".** The command is used to apply the changes required to reach the desired state of the configuration, or the pre-determined set of actions generated by a Terraform plan execution plan

11. The source code is found below SharePoint link.

10.