Create the new SQL Database

Overview

Platform:	Azure
Owner of this SOP:	Fully Managed POD A
Cloud Services:	Domain Joined Virtual Machines

Problem

When requesters are asking for create a new SQL database.

tester	Reviewer
@ Subhankar Ghosh	

Solution

Cloud Services operators will perform basic health checks and update the associated record as required. In most cases, the record will be an Incident Task (which may be assigned or manually created).

Pre-requisites

- The operator needs to have access to the **Cloud Factory** ServiceNow Dashboard. If this access is missing raise a request for Access to ServiceNow and request the group "Cloud Factory - Dashboard Access".
- Must be a member of the "Cloud Svcs- CloudOne" assignment group to work on ServiceNow tickets. As well access to the required report in this SOP. Use the same Access to ServiceNow request to gain this group if needed.
- The operator needs to be part of **OperationsCloudOneAccess** role in AWS & **CloudOne Operator** role in Azure . If this is missing raise a Manage Cloud Access request to get this custom role.

Procedure

Steps to create & manage Incident Task

Create Incident Task

Follow the below steps to create incident task. This step is required when the operator has received only an Incident.

On *Incident Tasks* Tab of the Incident ticket, click *New* and assign it to "CloudOne" team. Populate the Assignment group and Configuration item (CI). (Hint the CI is the VM on which the health check is being performed)
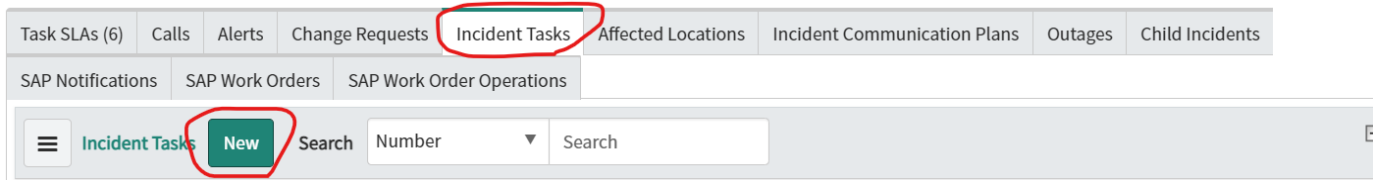
Steps

- Azure SQL server need to be present or be created before creation of SQL database.
- Clone the Blue Print Azure-PaaS-SQL https://gitlab.com/bhp-cloudfactory/azure-blueprints/azr-paas-sql-app-insight into your repository under new project.
- change all the variables in the .tfvars file as per your requirement. Please go through the blueprint Readme.md file for more details.
- change the variables in the env.sh file to match your environment and select proper `TFC_WORKSPACE_NAME`as shown in the below snapshot.

```
export ENVIRONMENT="npe" # prd, pov, pit


# ADD VM(S) SUBSCRIPTION ID HERE
export SUBSCRIPTION_ID="01818878-c294-4617-a15a-8b7934a433c4"


# NPE
export TFB_ORG="BHPEG-Dev" # or BHPEG
```

```
export REGION="aue1"

export ENV=${ENVIRONMENT}
export TFC_ADDR="app.terraform.io"

export TFC_WORKSPACE_NAME="a_azr_toolingdev_npe_mssql_01818878-c294-4617-a15a-8b7934a433c4"
```

- After changing all the values in required files as per request, commit the code and push to Gitlab repo.

```
                                         git add .

                                        git commit -m "new DB creation"
       700361a] new DB creation
 1 file changed, 22 insertions(+)

C:\Users\2075199\Downloads\clone\sql-pass>git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 374 bytes | 124.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
To gitlab.com:bhp-cloudfactory/tooling-foundations/fm-sops/sql-pass.git
```

- Verify terraform plan in TFC-Workspace after successfully completed Gitlab pipeline

ᛘ **main** ⟋ 700361a5

⬡ new DB creation   ✓ ✓ ✓

✓ **Plan finished** 3 hours ago   Resources: **13** to add, **0** to change, **0** to destroy ⌃

Started 3 hours ago > **Finished** 3 hours ago

+ 13 to create

> + 🔺 module.app_insights.azurerm_application_insights.this

> + 🔺 module.app_insights. ⋯ .azurerm_monitor_diagnostic_setting.this[0]

> + 🔺 module.resource_group[0].azurerm_resource_group.this

> + 🔺 module.sql_admin_secrets.azurerm_key_vault_secret.this[0]

> + 🔺 module.sql_admin_secrets.azurerm_key_vault_secret.this[1]

> + 🔺 module.sql_db["test1"].azurerm_mssql_database.this

> + 🔺 module.sql_elastic_pool[0].azurerm_mssql_elasticpool.this

> + 🔺 module.sql_pvt_endpoint.azurerm_private_endpoint.this

> + 🔺 module.sql_server.azurerm_mssql_server_security_alert_policy.this

> + 🔺 module.sql_server.azurerm_mssql_server_vulnerability_assessment.this

> + 🔺 module.sql_server.azurerm_mssql_server.this

- Execute Apply manually.

ᛘ **main** ⟋ 700361a5

⬡ new DB creation   ✓ ✓ ✓ ✓ ⚙

    ✓ apply   ▶

> + ▲ module.app_insights.azurerm_application_insights.this	✓ Created
> + ▲ module.app_insights. ⋯ .azurerm_monitor_diagnostic_setting.this[0]	✓ Created
> + ▲ module.resource_group[0].azurerm_resource_group.this	✓ Created
> + ▲ module.sql_admin_secrets.azurerm_key_vault_secret.this[0]	✓ Created
> + ▲ module.sql_admin_secrets.azurerm_key_vault_secret.this[1]	✓ Created
> + ▲ module.sql_db["test1"].azurerm_mssql_database.this	✓ Created
> + ▲ module.sql_elastic_pool[0].azurerm_mssql_elasticpool.this	✓ Created
> + ▲ module.sql_pvt_endpoint.azurerm_private_endpoint.this	✓ Created
> + ▲ module.sql_server.azurerm_mssql_server_security_alert_policy.this	✓ Created
> + ▲ module.sql_server.azurerm_mssql_server_vulnerability_assessment.this	✓ Created
> + ▲ module.sql_server.azurerm_mssql_server.this	✓ Created

- Also if any request come for another DB creation with existing environment, need to add another SQL DB code in .tfvars file and run pipeline. It will create another SQL DB.

```
"test2" = {
  name                         = "aue1-npe-fmpoda-sql-db-002" #
  create_mode                  = "Default"
  collation                    = ""
  sku_name                     = "ElasticPool" # Use dedicated S
  license_type                 = ""
  auto_pause_delay_in_minutes  = 0
  max_size_gb                  = 5
  min_capacity                 = 0
  read_replica_count           = 0
  read_scale                   = false
  sample_name                  = ""
  restore_point_in_time        = ""
  zone_redundant               = false
  retention_days               = 91
  threat_detection_policy = {
    use_server_default    = false
    email_account_admins  = false
    email_addresses       = ["subhankar.ghosh@bhp.com"]
    retention_days        = 91
  }
}
}
```

✅ **Plan finished**   2 hours ago   Resources: **1** to add, **0** to change, **0** to destroy   ⌃

Started 2 hours ago   >   **Finished** 2 hours ago

```
+  1 to create
```

```
∨  +  ▲  module.sql_db["test2"].azurerm_mssql_database.this  ⬢

   +   auto_pause_delay_in_minutes :       0
   +   collation :                         Known after apply
   +   create_mode :                       "Default"
   +   creation_source_database_id :       Known after apply
   +   elastic_pool_id :                   "/subscriptions/01818878-c294-4617-a15a-8b7934a433c4/resourceGroups/it-aue1-npe-arg-sql/providers/Microsof
                                           t.Sql/servers/it-aue1-npe-fmpoda-sql-dbs-ndtywa/elasticPools/it-aue1-npe-fmpoda-sql-ep"
```

- Check in Azure portal for all the resource status.

☐	🗄 it-aue1-npe-fmpoda-sql-db (it-aue1-npe-fmpoda-sql-dbs-ndtywa/it⋯	SQL database	Australia East
☐	🗄 it-aue1-npe-fmpoda-sql-db-002 (it-aue1-npe-fmpoda-sql-dbs-ndty⋯	SQL database	Australia East
☐	🗄 it-aue1-npe-fmpoda-sql-dbs-ndtywa	SQL server	Australia East
☐	🖧 it-aue1-npe-fmpoda-sql-dbs-ndtywa-pvt-endpo.nic.035e3fde-e8f2-⋯	Network interface	Australia East
☐	⟨⌁⟩ it-aue1-npe-fmpoda-sql-dbs-ndtywa-pvt-endpoint	Private endpoint	Australia East

Related articles

- AWS VPC Endpoints configuration change
- Renewal / Upload SSL certificate for existing Application
- Monitoring GitLab Access Audit Report
- Azure Storage Account Blueprint SOP - BPAZR022
- Azure App Service FTP Credential Sharing and Reset Procedure