

Final Project

Azure Site Recovery (ASR) Services for Virtual Machine Migrations

Problem Statement:

There was a real use case for McKesson within the last year. The problem was we needed in a short period of time migrate Virtual Machines from our VMWare infrastructure, that was coming to capacity to Azure. The other problem is the group tasked needed to learn ASR, along with the challenges that come with ASR. In this use case it goes through the process of the Azure Site Recovery setup along with processing a failover.

Overview of the Technology:

To solve the problem, I built an Azure Site Recovery system which includes; windows 2012 R2 vm in our VMWare virtual infrastructure to act as the ASR gateway, added the proper permissions to the VCenter to allow read access for ASR, configured ASR to connect to the VCenter and finally build the failover replication processes for the VM's in question.

High Level Steps:

- 1) Build ASR in Azure, download the Gateway Server software.
- 2) Install and configure software on Gateway server VM.
- 3) Connect VCenter(s) and establish inventory.
- 4) Build replication processes.
- 5) Execute replication and failover.

Data Source:

McKesson VCenter VM inventory

Hardware Used:

Windows 2012 R2 VM for Gateway server, Windows 7 laptop to access Azure Portal.

Software Used:

Azure Site Recovery Services

YouTube Links:

2 Min: <https://youtu.be/Aax2aDeOCS8>

15 Min: <https://youtu.be/lmrq8edwxv4>

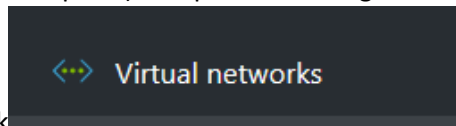
Git Repo: https://github.com/dacoudriet-mck/final_project.git

Steps to Setup Azure Recovery Services Vault

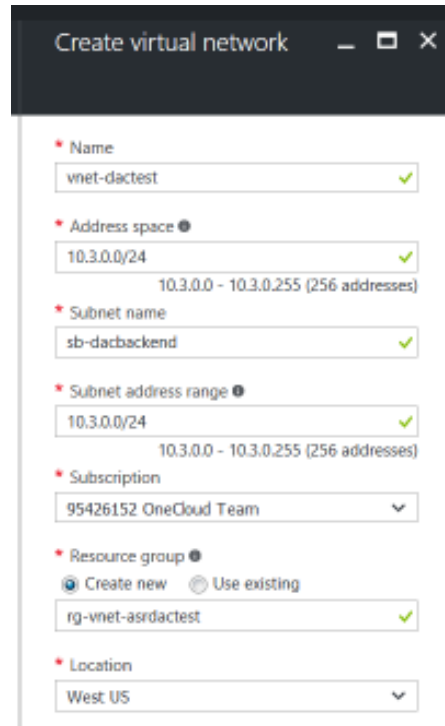
1. Assumptions:
 - a. Know how to access to Azure portal (portal.azure.com)
 - b. Access to VCenter for on premise Management Server setup.
 - c. Will use azure portal for all deploys. If familiar with Powershell/Azure CLI and Azure Templates you are welcome to use those methods.
 - d. Azure naming conventions we are developing are going to be followed.
 - e. **Management/Configuration and Process server is the same windows 2012 R2 vm deployed in VCenter.**
2. In VCenter deploy a self managed Windows 2012 R2 server in either NDH or DDC.
 - a. 16 GB Ram
 - b. 4 CPU
 - c. At least a 600 GB secondary drive.
 - d. NOTE: These are very low end config for testing only.
 - e. Turn off Windows Firewall and UAC.
 - f. Add the following service acct as a local administrator: namck\dd8x1m0
 - g. **Going forward for all ASR setup on the server log in as that service account.**
 - h. NOTE: Password will be supplied.
 - i. **Download from [\\mtsis-212.oc.mckesson.com\c\\$\installs](https://mtsis-212.oc.mckesson.com/c$/installs) the VMWare CLI 6.0 (VMware-PowerCLI-6.0.0-2548067.exe) and install on your server logged in as the service account.**

NOTE: Just follow the default install instructions and reboot when completed.

3. In Azure (in your test subscription) Setup the following items:



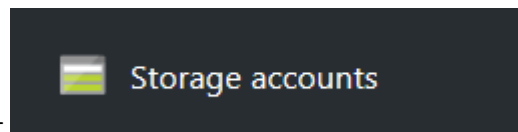
- a. Virtual Network
 - i. Select virtual Networks and click add.



- ii.
- iii. Name – “vnet-<initials>test”
- iv. Address space leave as is
- v. Subnet name – “sb-<initials>backend”
- vi. Subnet range leave as is
- vii. Subscription – Will be your test subscription
- viii. Resource group -create new
 - 1. Name: “rg-vnet-asr<initials>test”
- ix. Location - West US or East US 2.

NOTE: It is important to remember this location since your other azure resources will be deployed in the same location. Also we chose West US or East US 2 since that is the location of each of our Express Routes.

- x. Click Create.
- xi. This will then create a virtual network in the resource group of the name you supplied.



b. Storage Account

- i. Click on Storage accounts and click add.

The screenshot shows the 'Create storage account' form in the Azure portal. The form is titled 'Create storage account' and includes a close button. Below the title, there is a note: 'The cost of your storage account depends on the usage and the options you choose below. [Learn more](#)'. The form contains several fields and options:

- Name:** saasrdactst01 (with a green checkmark icon)
- Deployment model:** Resource manager (selected), Classic
- Account kind:** General purpose (selected)
- Performance:** Standard (selected), Premium
- Replication:** Zone-redundant storage (ZRS) (selected)
- Storage service encryption:** Disabled (selected), Enabled
- Subscription:** 95426152 OneCloud Team (selected)

ii.

iii. Name – “saasr<initials>tst01”

1. This MUST be unique across azure.
2. And can only contain lowercase and only numbers and letters and no hypens.

iv. Model – Resource Manager

v. Kind – General Purpose

vi. Performance – Standard - This is test only if production we would probably use premium for OS disks.

vii. Replication – just kept it zone redundant for test only.

viii. Encryption – Disabled – once again test only

ix. Subscription – Your Test subscription as used above.

The screenshot shows the 'Resource group' section of the Azure portal. It includes the following fields and options:

- Resource group:** Create new (selected), Use existing
- Resource group name:** rg-sa-asrdactest (with a green checkmark icon)
- Location:** West US (selected)

x.

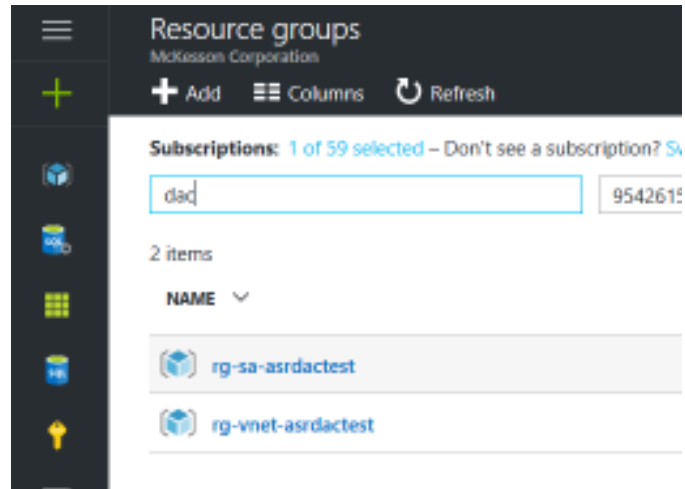
cont....

xi. Resource Group

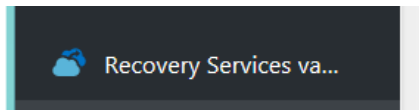
1. Create New
2. Name – “rg-sa-asr<initials>test”

xii. Click Create

c. When completed the following resource groups should exist:



- i.
- ii. The rg-sa.... Containing the storage account
- iii. The rg-vnet.... Containing the virtual Network.
- iv. We used separate resource groups to get in the habit of deploying items that are part of the same life cycle. This is model will not always be the case since storage and compute might be in the same resource group depending on the use case.

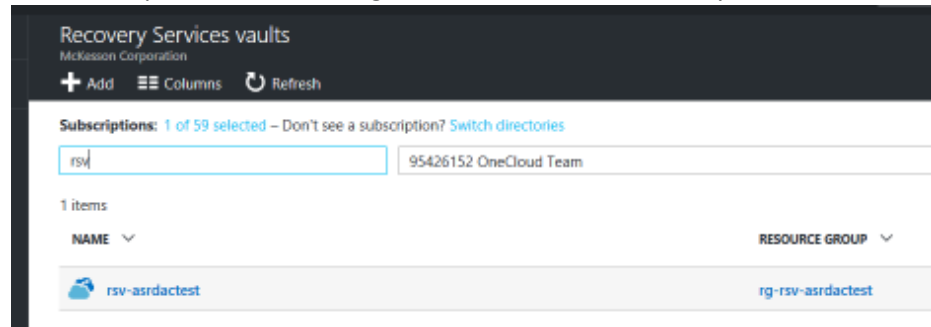


d. Recovery Services Vault.

- i. Click add on the Recovery Services Vault.

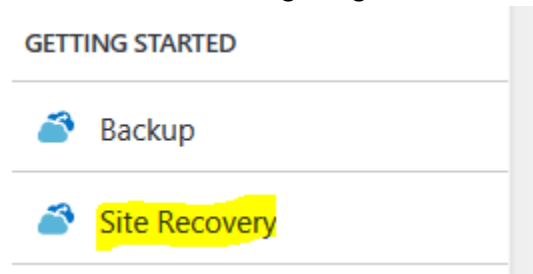
- ii.
- iii. Name – “rsv-asr<initials>test”
- iv. Subscription – Once again use your test subscription.
- v. Resource Group
 - 1. Create New
 - 2. Name: “rg-rsv-asr<intials>test>”
- vi. Click Create

- vii. When Completed the following will exist under the recovery services vault.



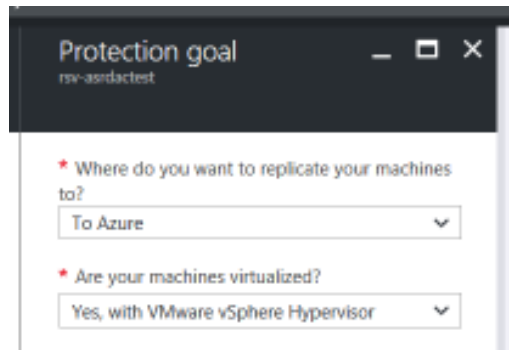
- e. Configure the Recovery Services Vault.

- i. Select the vault and under getting started select site recovery

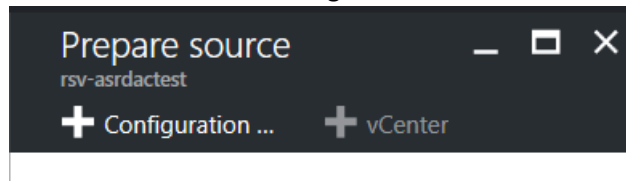


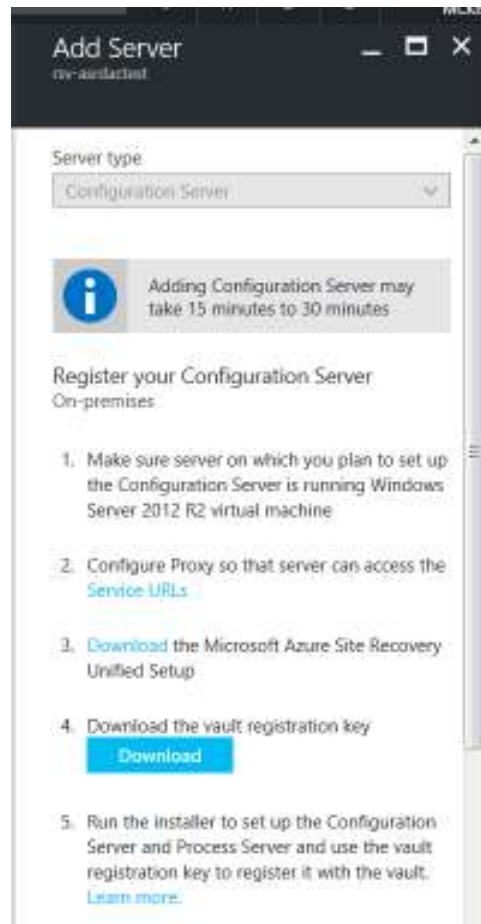
- ii. This will take you through the wizard of setting up the Management server deployed in VCenter.
iii. Select Site Recovery and Select Step 1.

1. Protection Goal – Azure from VMware



2. Source -> select Add Configuration



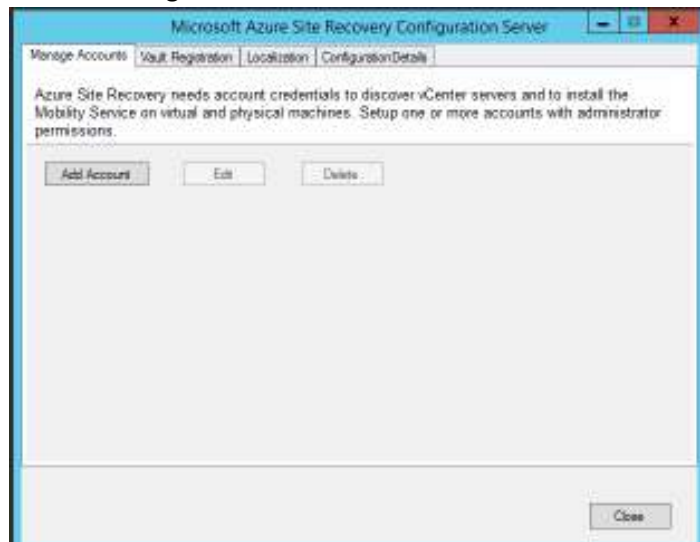


- 3.
4. Step 1 and 2 were done above with setup of the VM.
5. Step 3 Download the ASR install and copy it to your VCenter server setup in Section 2. – This will take some time to download.
6. Step 4 Download the registration key and save it to your VCenter Server
7. Step 5 run the unified setup on your VCenter Server.
 - a. Right click on the install and select run as administrator.
 - b. Select install configuration and process server -> Click Next.
 - c. Third Party MySQL install -check accept -> Click Next
 - d. Site Recovery Registration Key -> browse to the key you downloaded in step 6 above - >Click Next.
 - e. Connect directly WITHOUT Proxy. -> Click Next
 - f. It will go through the Pre-Requisite Checks. NOTE: CPU/MEM and/or DISK will give warning. Just ignore. -> Click Next.
 - g. Supply a MYSQL root and db password. This is for the local machine. Note somewhere what you used **but it should not be referenced during any additional setup work**. -> Click Next
 - h. Protect VMware machines – Select Yes -> it will verify that the VMWare CLI is installed (**section 2 step “i”**) -> Click Next
 - i. Software install location. I left it as the default and ignore the warning -> Click Next.

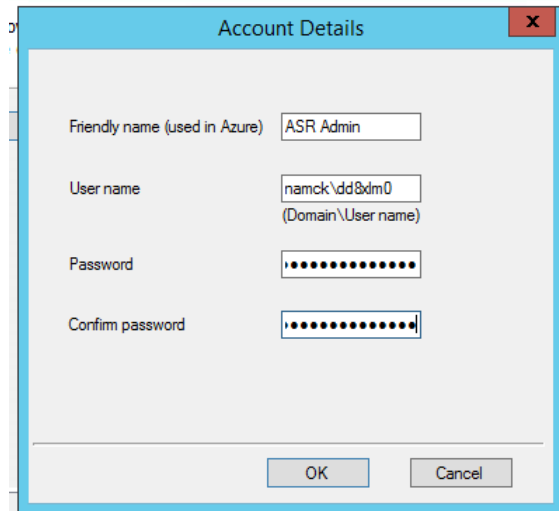
- j. Network Select -> Use vm NIC assigned to the vm and leave the port as 9443 -> Click Next
- k. Summary -> Click Install **(This process will take 30 – 40 minutes)**
- l. When completed it should look similar to:

	Steps	Status
✓	Download MySQL	Already Installed
✓	Configuring Internet Information Services features	Done
✓	Install third-party components	Done
✓	Install configuration server and process server	Done
✓	Install Microsoft azure site recovery provider	Done
✓	Install master target server	Done
✓	Install Microsoft azure recovery services agent	Done
✓	Registering with Site Recovery	Done
✓	Validating server configuration	Done

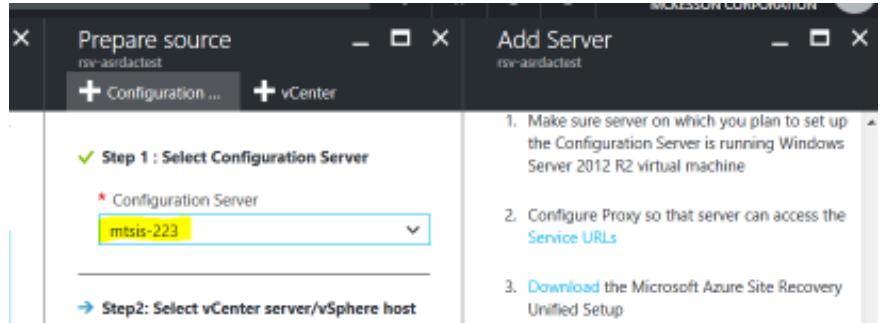
- m. Click Finish.
- n. It will first prompt to reboot -> Click OK, this is just a notice not a force reboot.
- o. You will then be prompted to save the passphrase to your clipboard. Paste that into a notepad file and save to a drive on the server.
- p. Step 6 the cpsconfigtool.exe will automatically load.
- q. On the Manage Accounts tab



- r. Click Add Account



- s.
- t. Step 7: Add the Service account.
 - Friendly Name: Name visible in Azure
 - User name: Domain\UserID
 - Password: password supplied
- u. Click OK
- v. It will state successfully added and then appear in the drop down list.
- w. Reboot the Server
- x. The in Azure the following will appear:



- y. Click Add vCenter -> wait for server to reboot and services started.

- z.
- aa. Configuration Server should be automatic
 - bb. VCenter -> select NDH or DDC pod1 or pod2, this example was NDH Pod1. **NOTE: For the test Pick a DC where you can or have vm's deployed that you have local admin login permissions to.**
 - cc. Port leave as 443
 - dd. Host account is the ASR admin created above in step "s".
 - ee. Click OK -> This process takes time while it downloads the inventory.
 - ff. When Completed it will show success:

NAME	STATUS	START TIME	DURATION	
Adding vCenter server	Successful	1/24/2017, 12:29:09 AM	00:47:00	..

- gg. And appear in the prepare source screen:

- hh. Click OK.
- ii. Target this is just for setup (and default) and not the permanent end point for all replication.
 - Select your subscription
 - Failover Model: Resource Manager

- Click Add Storage account
- Select Create New
- On the Create Storage account screen

Create storage account

★ Name

saasrdapprimary01 ✓
.core.windows.net

Performance ⓘ

Standard Premium

Replication ⓘ

Locally-redundant storage (LRS) ▼

- Name: saasr<initials>primary01
- Performance: Standard, Replication: LRS
- NOTE: The initial storage account must be created in the same resource group as the Recovery Vault. The other storage account can be used for other processes we will train on later.

- Click virtual network and select the one created above. This doesn't need to be in the same resource

Target
rsv-asrdactest

+ Storage account + Network

✓ Step 1 : Select Azure subscription

* Subscription ⓘ
ETS Azure Services

* Select the deployment model used after failover ⓘ
Resource Manager

✓ Step 2 : Ensure that at least one compatible Azure storage account exist

Storage account(s) ⓘ
saasrdactste201

✓ Step 3 : Ensure that at least one compatible Azure virtual network exist

Network(s) ⓘ
vnet-dactest

group.

- When completed it will look like this above -> Click OK.
- jj. Create a default replication policy:

Replication policy
rsv-asrdactest

+ Create and Ass... Associate Dissociate

- kk. Click the Add create and associate button.

Create and associate po...
rty-azsantest

* Name policy-default

Source type VMware / Physical machines

Target type Azure

* RPO threshold in mins 15

* Recovery point retention in hours 24

* App-consistent snapshot frequency in mins 60

Failback replication policy name policy-default-failback

- ll.
- mm. Give it a standard name and it will auto fill in all the default values (not shown associated configuration server), leave as it. (scroll down from image above) Click OK
- nn. It will then create a standard failback policy when done click OK.
- oo. Final step on capacity planning select yes I did complete and click ok.
- pp. Click OK again and the initial setup is complete.

8. To view the configuration/modify under the Manage of the Recovery Services vault (menu item below the getting started) select Site Recovery Infrastructure. In the middle for VMWare select configuration servers -> Click on your server -> Expand out the process,

Associated servers

NAME	STATUS	SERVER
▼ Process Ser...		
mtsis-223	Connected	Process
▼ vCenter Ser...		
ndh1-vb...	Connected	vCenter
▼ Master Targ...		
mtsis-223	Connected	Master

vcenter and master target.
show as connected.

All should

9. Additional items to try:

- a. Add another vcenter (from the screen in step 8 select add vcenter and the steps are the same as earlier).
- b. Spin up a vm that you will be the local administrator of in VCenter (self managed). Note: It must be in a vcenter you have configured above.
- c. Add additional logins on the Configuration Server (follow steps o, p and q from the configuration install section).
 - NOTE: for Linux the login must be root and be formatted as “.\root” as the user name.

ARM Templates to Build the virtual Network with a subnet:

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "BU_name": {
      "defaultValue": "ETS",
      "metadata": {
        "description": "BU name per our scaffold naming conventions"
      },
      "type": "String"
    },
    "vnet_context": {
      "defaultValue": "DevTest",
      "metadata": {
        "description": "Azure vnet context per our scaffold naming conventions"
      },
      "type": "String"
    },
    "vnet_number": {
      "defaultValue": "01",
      "metadata": {
        "description": "Azure vnet number per our scaffold naming conventions and what exists currently in region and subscription"
      },
      "type": "String"
    },
    "CIDRBlock": {
      "defaultValue": "0.0.0.0/22",
      "metadata": {
        "description": "CIDR supplied by Network for next available IP space"
      },
      "type": "String"
    },
    "subnetFE1": {
      "defaultValue": "0.0.0.0/24",
      "metadata": {
        "description": "CIDR for Front END subnet"
      },
      "type": "String"
    },
    "subnetBE1": {
      "defaultValue": "0.0.0.0/24",
      "metadata": {
```

```
        "description": "CIDR for Backend subnet 1"
      },
      "type": "String"
    },
    "subnetBE2": {
      "defaultValue": "0.0.0.0/24",
      "metadata": {
        "description": "CIDR for Backend subnet 2"
      },
      "type": "String"
    },
    "subnetDMZ": {
      "defaultValue": "0.0.0.0/25",
      "metadata": {
        "description": "CIDR for DMZ subnet"
      },
      "type": "String"
    },
    "subnetMGMT": {
      "defaultValue": "0.0.0.0/25",
      "metadata": {
        "description": "CIDR for MGMT"
      },
      "type": "String"
    }
  },
  "variables": {
    "location": "[resourceGroup().location]",
    "vnet_name": "[concat('vnet-', parameters('BU_name'), parameters('vnet_context'),
parameters('vnet_number'), '-', variables('location'))]",
    "routeTables_udr": "udr-DEFAULT_TRANSIT",
    "subnetFE1_name": "[concat('sb-', parameters('BU_name'), 'frontend1')]",
    "subnetBE1_name": "[concat('sb-', parameters('BU_name'), 'backend1')]",
    "subnetBE2_name": "[concat('sb-', parameters('BU_name'), 'backend2')]",
    "subnetmgmt_name": "[concat('sb-', parameters('BU_name'), 'mgmt')]",
    "subnetDMZ_name": "[concat('sb-', parameters('BU_name'), 'DMZ')]"
  },
  "resources": [
    {
      "comments": "Build RouteTable",
      "type": "Microsoft.Network/routeTables",
      "name": "[variables('routeTables_udr')]",
      "apiVersion": "2016-03-30",
      "location": "[resourceGroup().location]",
      "properties": {
```



```
        "routes": [
            {
                "name": "default",
                "properties": {
                    "addressPrefix": "0.0.0.0/0",
                    "nextHopType": "VirtualAppliance",
                    "nextHopIpAddress": "10.15.249.4"
                }
            }
        ],
    },
    "resources": [],
    "dependsOn": []
},
{
    "comments": "vnet setup",
    "type": "Microsoft.Network/virtualNetworks",
    "name": "[variables('vnet_name')]",
    "apiVersion": "2016-03-30",
    "location": "[resourceGroup().location]",
    "properties": {
        "addressSpace": {
            "addressPrefixes": [
                "[parameters('CIDRBlock')]"
            ]
        },
        "subnets": [
            {
                "name": "[variables('subnetFE1_name')]",
                "properties": {
                    "addressPrefix": "[parameters('subnetFE1')]",
                    "routeTable": {
                        "id": "[resourceId('Microsoft.Network/routeTables',
variables('routeTables_uvr'))]"
                    }
                }
            },
            {
                "name": "[variables('subnetBE1_name')]",
                "properties": {
                    "addressPrefix": "[parameters('subnetBE1')]",
                    "routeTable": {
                        "id": "[resourceId('Microsoft.Network/routeTables',
variables('routeTables_uvr'))]"
                    }
                }
            }
        ]
    }
}
```

```
    }
  },
  {
    "name": "[variables('subnetBE2_name')]",
    "properties": {
      "addressPrefix": "[parameters('subnetBE2')]",
      "routeTable": {
        "id": "[resourceId('Microsoft.Network/routeTables',
variables('routeTables_uds'))]"
      }
    }
  },
  {
    "name": "[variables('subnetDMZ_name')]",
    "properties": {
      "addressPrefix": "[parameters('subnetDMZ')]",
      "routeTable": {
        "id": "[resourceId('Microsoft.Network/routeTables',
variables('routeTables_uds'))]"
      }
    }
  },
  {
    "name": "[variables('subnetmgmt_name')]",
    "properties": {
      "addressPrefix": "[parameters('subnetmgmt')]",
      "routeTable": {
        "id": "[resourceId('Microsoft.Network/routeTables',
variables('routeTables_uds'))]"
      }
    }
  }
],
"resources": [],
"dependsOn": [
  "[resourceId('Microsoft.Network/routeTables', variables('routeTables_uds'))]"
]
}
```

JSON Template to Build Storage account:

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "storageAccountName": {
      "type": "string",
      "defaultValue": "<deploymentprefix><index>sa",
      "metadata": {
        "description": "Storage Account Name"
      }
    },
    "storageAccountType": {
      "type": "string",
      "defaultValue": "Standard_LRS",
      "allowedValues": [
        "Standard_LRS",
        "Standard_GRS",
        "Standard_ZRS",
        "Premium_LRS"
      ],
      "metadata": {
        "description": "Storage Account type"
      }
    },
    "location": {
      "defaultValue": "WestUS",
      "allowedValues": [
        "WestUS",
        "EastUS",
        "CentralUS",
        "EastUS2",
        "WestUS2"
      ],
      "type": "string",
      "metadata": {
        "description": "Location"
      }
    }
  },
  "variables": {
```

```
    "storageAccountName": "[concat(uniquestring(resourceGroup().id),  
'standardsa')]"  
  },  
  "resources": [  
    {  
      "type": "Microsoft.Storage/storageAccounts",  
      "name": "[variables('storageAccountName')]",  
      "apiVersion": "2016-01-01",  
      "location": "[parameters('location')]",  
      "sku": {  
        "name": "[parameters('storageAccountType')]"  
      },  
      "kind": "Storage",  
      "properties": {  
      }  
    }  
  ],  
  "outputs": {  
    "storageAccountName": {  
      "type": "string",  
      "value": "[variables('storageAccountName')]"  
    }  
  }  
}
```