Azure Service Fabric



Vision Scope

Prepared for

Microsoft

22-Jan-16

Version 1.0 Draft

Prepared by

**Kristian Nese**

CTO

kristian.nese@lumagate.com

Contributors

**Add Contributors to Doc Properties**

Table of Contents

[1 Before you begin 3](#_Toc440991313)

[2 Lab 1: Creating Azure Site Recovery Vault and enabling VM Protection 3](#_Toc440991314)

[3 Lab 2: Performing a Planned Failover and assigning RBAC to tenant 15](#_Toc440991315)

1. Before you begin

Preview Documentation: Note that Azure Service Fabric is currently in a public preview, and some of content is subject to change

The objective of this Lab is to learn how to implement Service Fabric and explore some of its capabilities from an operation standpoint, as well as a developer using the Programming Model for stateful and stateless actors and services

* Admin access to an Azure Subscription
* Visual Studio 2015 with the Azure and Service Fabric SDK installed
* Microsoft Azure PowerShell (1.0.2 or higher)

1. Lab 1: Deploying Service Fabric

In this lab you will create and deploy a Service Fabric Cluster in Azure, and then (optionally) locally on your computer.

**Part 1 – Deploy Azure Service Fabric Cluster**

1. From your computer, open PowerShell ISE and login to your Azure subscription with the ‘Login-AzureRM’ cmdlet, and select the subscription using ‘Select-AzureRMSubscription –SubscriptionName’ cmdlet.
2. Run the following cmdlet to provision a new Service Fabric Cluster in a new Azure Resource Group

Get-AzureRmResourceProvider -ProviderNamespace “Microsoft.RecoveryServices”

Get-AzureRmResourceProvider -ProviderNamespace “Microsoft.SiteRecovery”

1. If not, register the ASR Resource Provider with the following cmdlets:

Register-AzureRmResourceProvider -ProviderNamespace “Microsoft.SiteRecovery”

Register-AzureRmProviderFeature -FeatureName betaAccess -ProviderNamespace “Microsoft.RecoveryServices”

Register-AzureRmResourceProvider -ProviderNamespace “Microsoft.RecoveryServices”

1. The registration can take up to an hour before completed. You can check the state of the process by executing two PowerShell cmdlets

Get-AzureRmResourceProvider -ProviderNamespace “Microsoft.RecoveryServices”

Get-AzureRmResourceProvider -ProviderNamespace “Microsoft.SiteRecovery”

Once this has completed, you should have successfully registered the ASR RP to your Azure subscription

**Part 2 – Creating the ASR environment in Azure**

Open PowerShell ISE and follow the guidelines.

1. Define some variables you will be using through this recipe:

mkdir -Path "c:\ASR"

$RG = "BCDR"

$Location = "west europe"

$vault = "HyperV"

$path = "c:\ASR"

1. Run the following cmdlet to create a new dedicated Resource Group

New-AzureRmResourceGroup -Name $RG -Location $Location

ResourceGroupName : BCDR

Location : westeurope

ProvisioningState : Succeeded

Tags :

ResourceId : /subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR

1. Run the following cmdlet to create a new ASR vault to your Resource Group, generate vault settings and import the file

$vault = New-AzureRmRecoveryServicesVault -Name $vault -ResourceGroupName $RG -Location $Location

$FilePath = Get-AzureRmRecoveryServicesVaultSettingsFile -Vault $vault -Path $path | % FilePath # changed this to make sure we get the right filepath

FilePath

--------

c:\ASR\HyperV\_2016-01-18T09-40-26.VaultCredentials

Import-AzureRmSiteRecoveryVaultSettingsFile -Path $FilePath

ResourceName ResouceGroupName ResourceNamespace ResouceType

------------ ---------------- ----------------- -----------

HyperV BCDR Microsoft.RecoveryServices vaults

1. Create a new Hyper-V Site in the ASR vault, and verify the progress

$sitename = "KNHVSite"

New-AzureRmSiteRecoverySite -Name $sitename

Name : 2791e93c-7826-489b-8f76-56581daba037

ID : /Subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR/providers/Microsoft.RecoveryServices/vaults/HyperV/replicationJobs/2791e93c-7826-489b-8f76-56581daba037

Type : Microsoft.RecoveryServices/vaults/replicationJobs

JobType : CreateSite

DisplayName : Create a site

ClientRequestId : f638fdc2-af1a-4461-93a4-a96d67c7a487-2016-01-18 09:41:00Z-P ActivityId: e94f192e-b12a-4f77-87a8-c1ef6d04700f

State : Succeeded

StateDescription : Completed

StartTime : 18-Jan-16 10:40:58 AM

EndTime : 18-Jan-16 10:40:59 AM

TargetObjectId : 646de81b-2f5c-4b73-ad37-85800b9b35f1

TargetObjectType : Server

TargetObjectName : KNHVSite

AllowedActions :

Tasks : {RegisterVmmTask}

Errors : {}

# Query the job

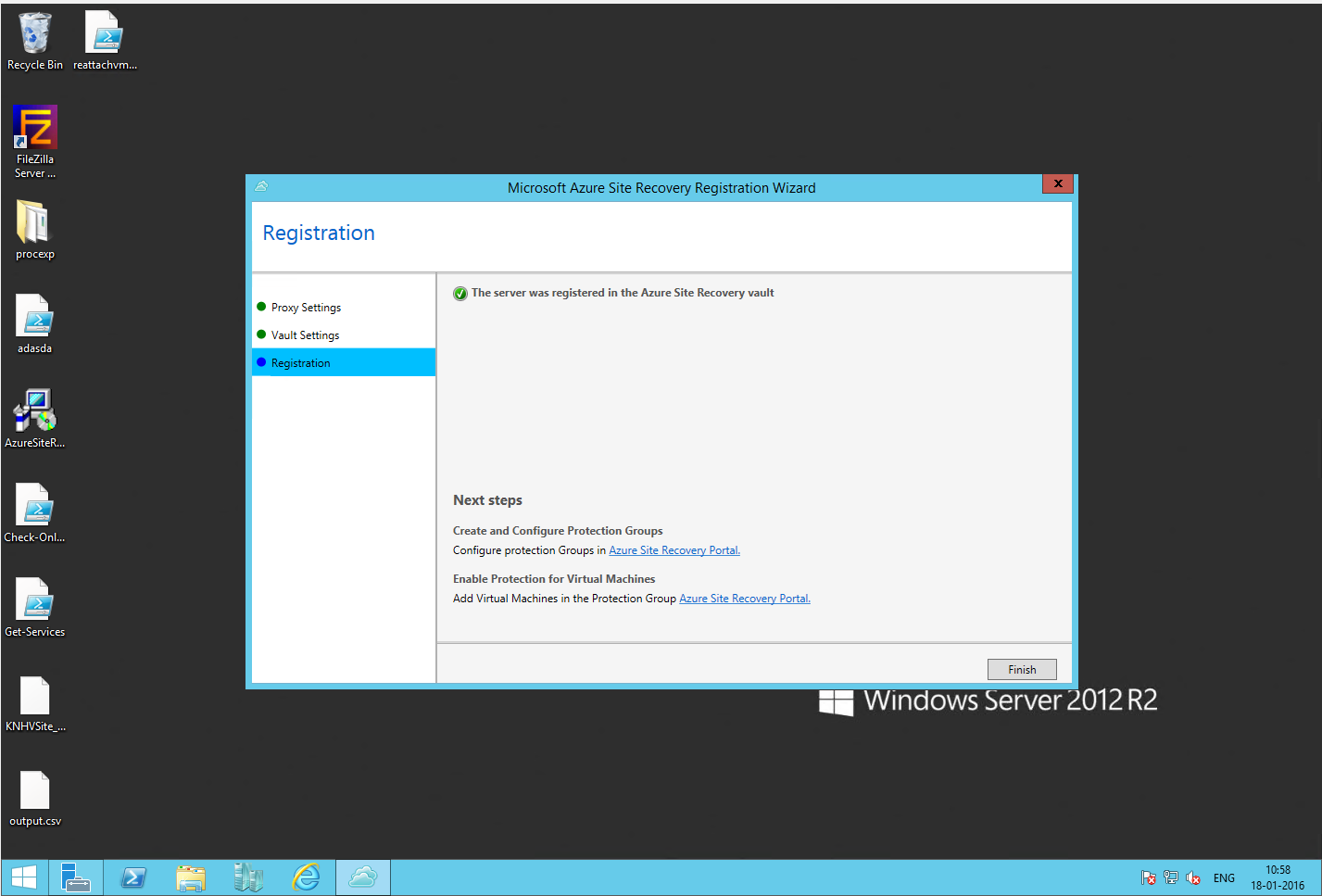
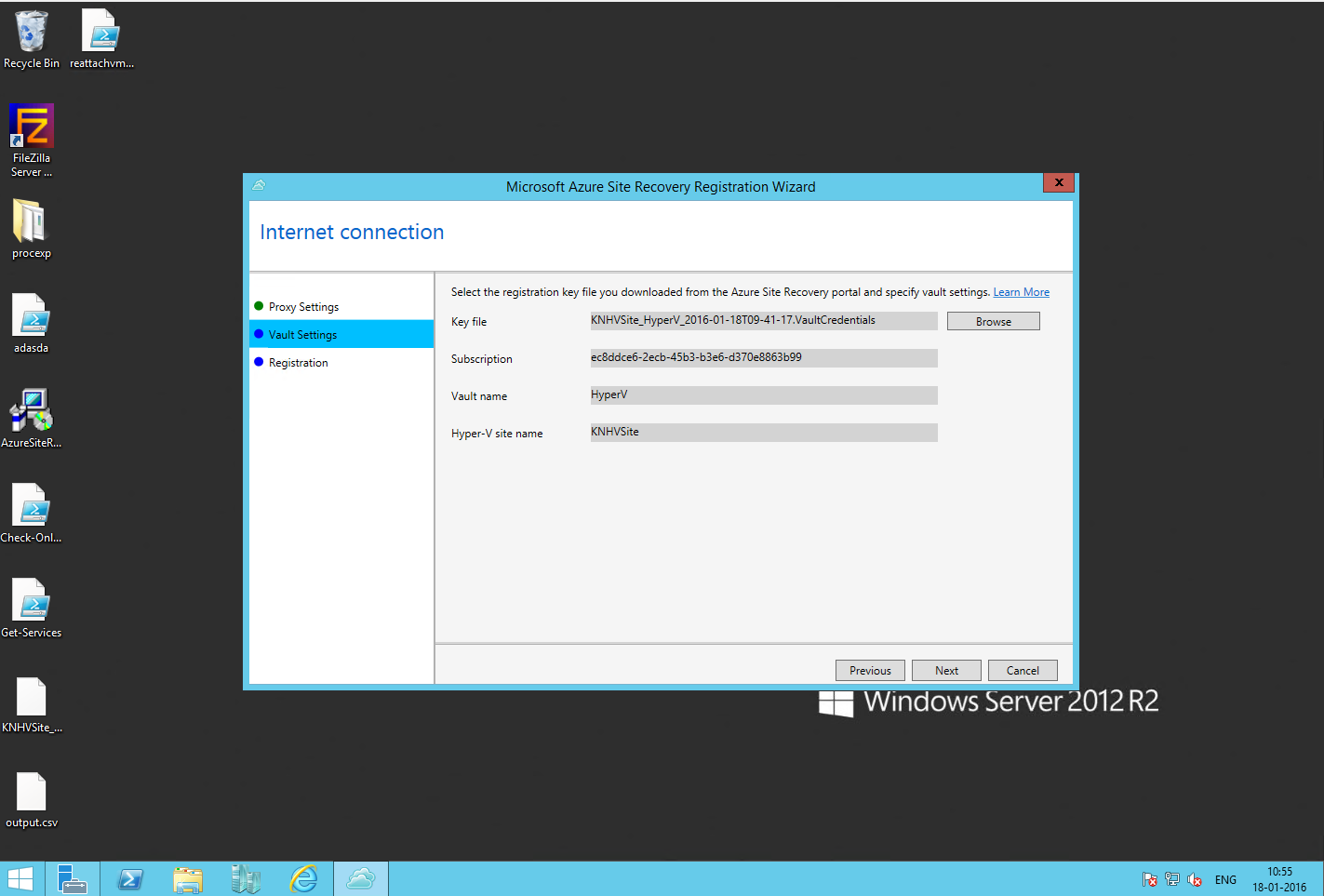
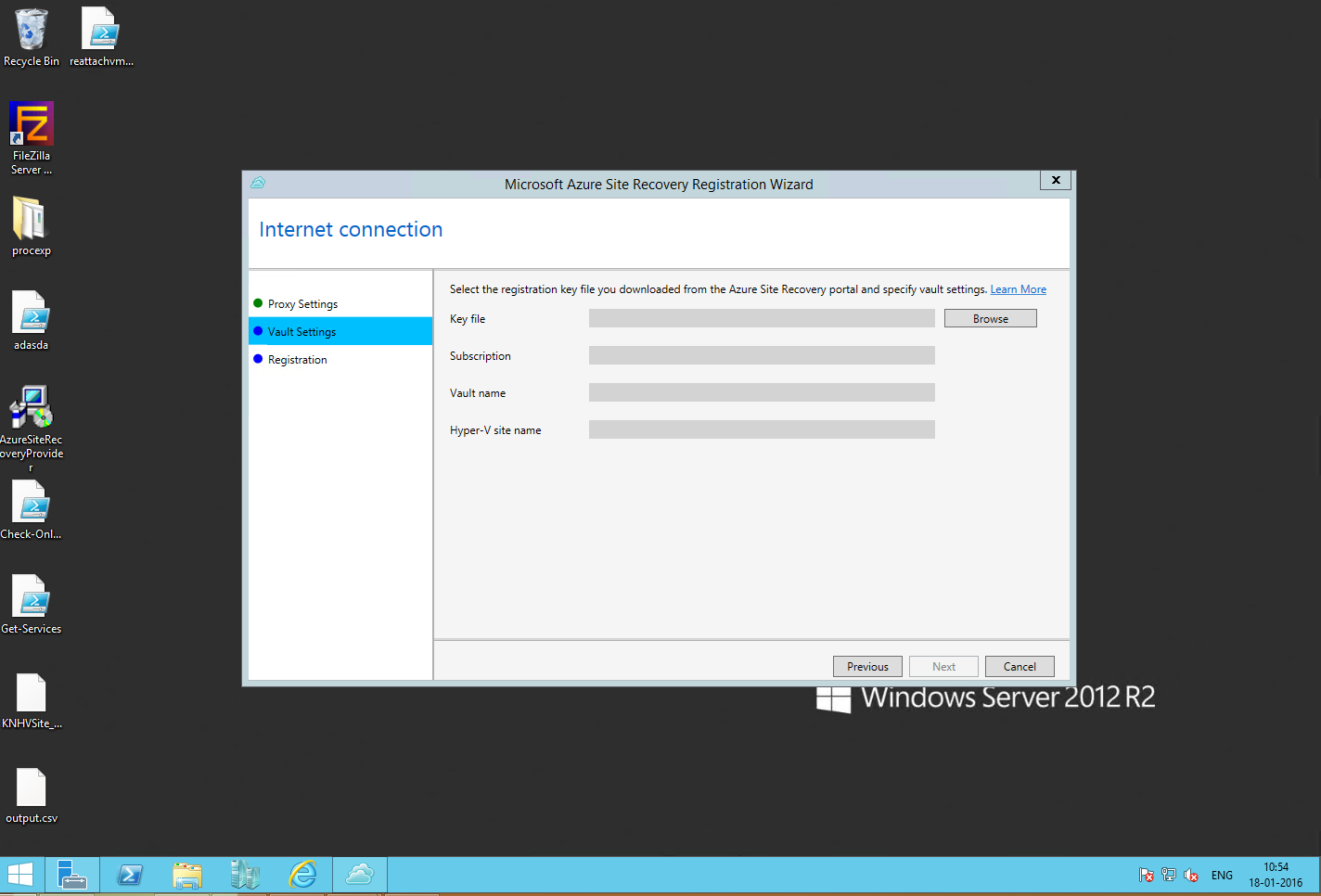
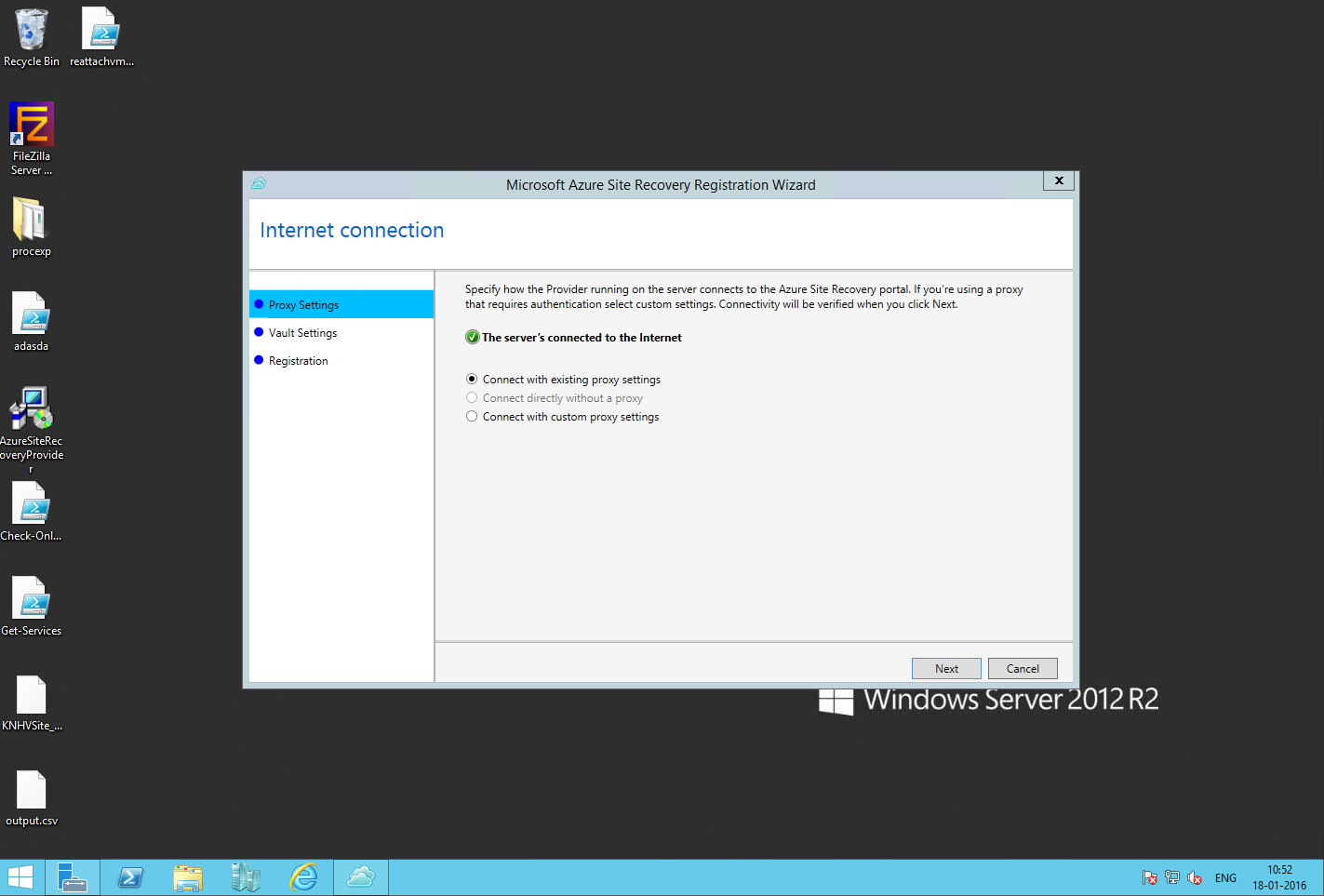
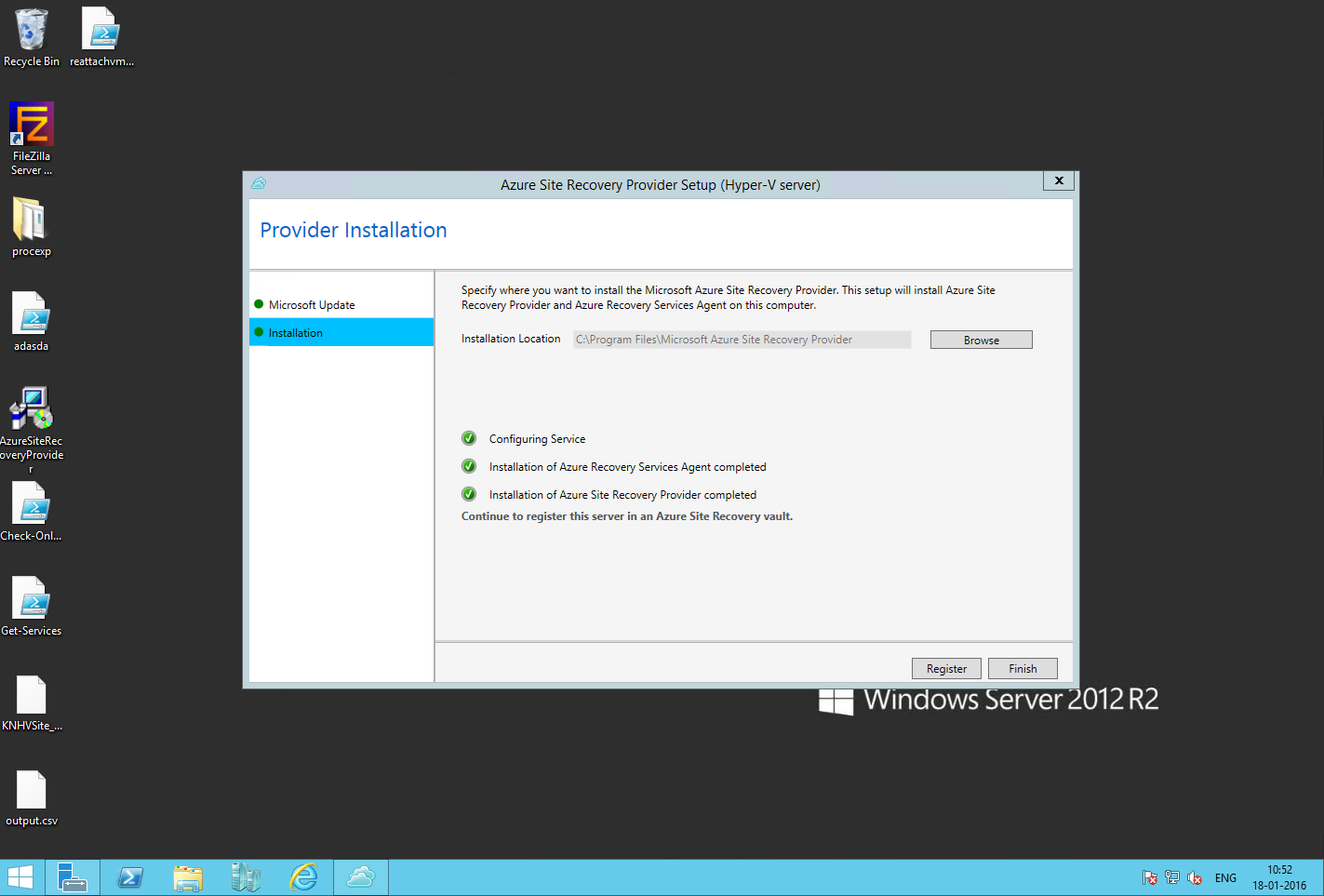
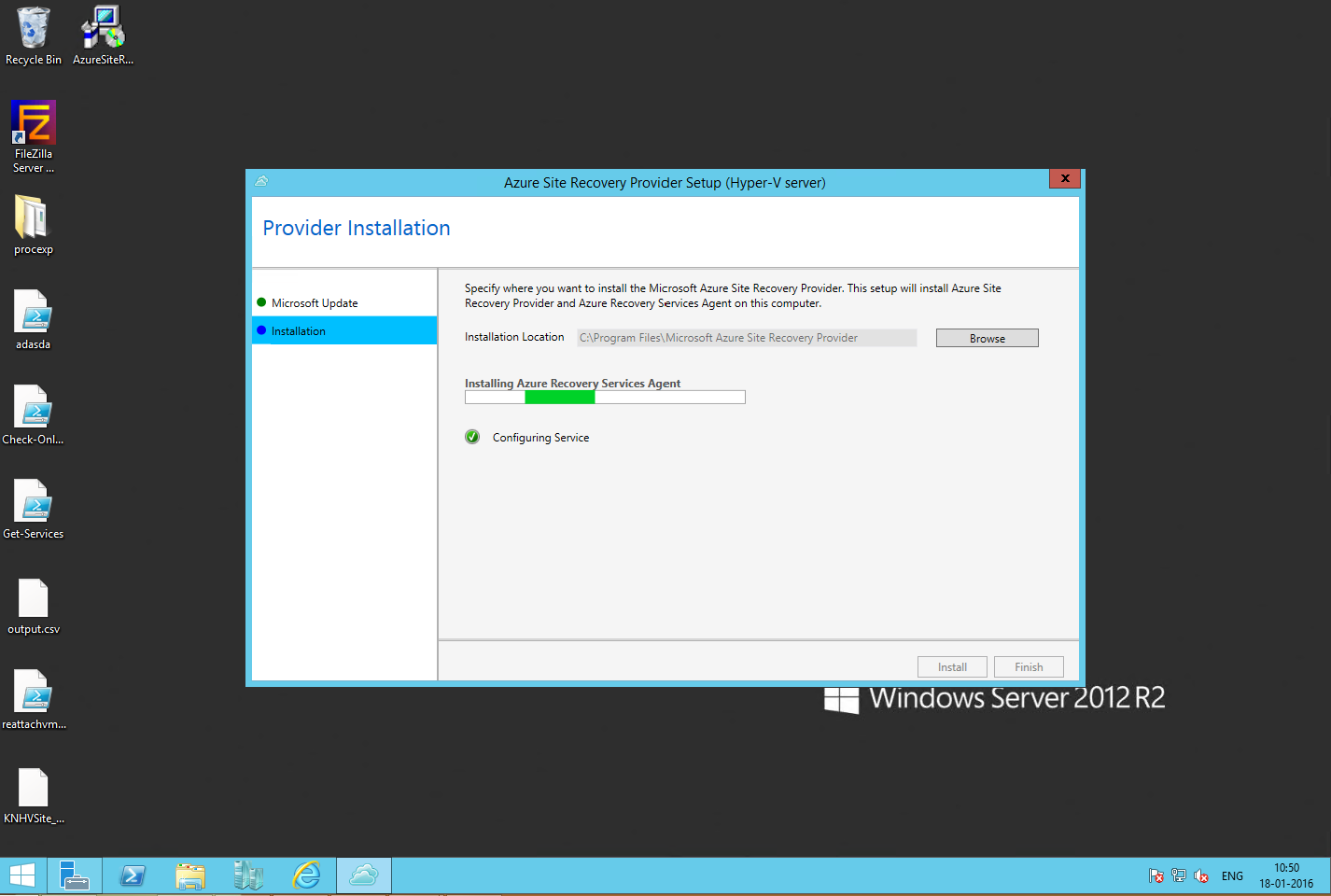
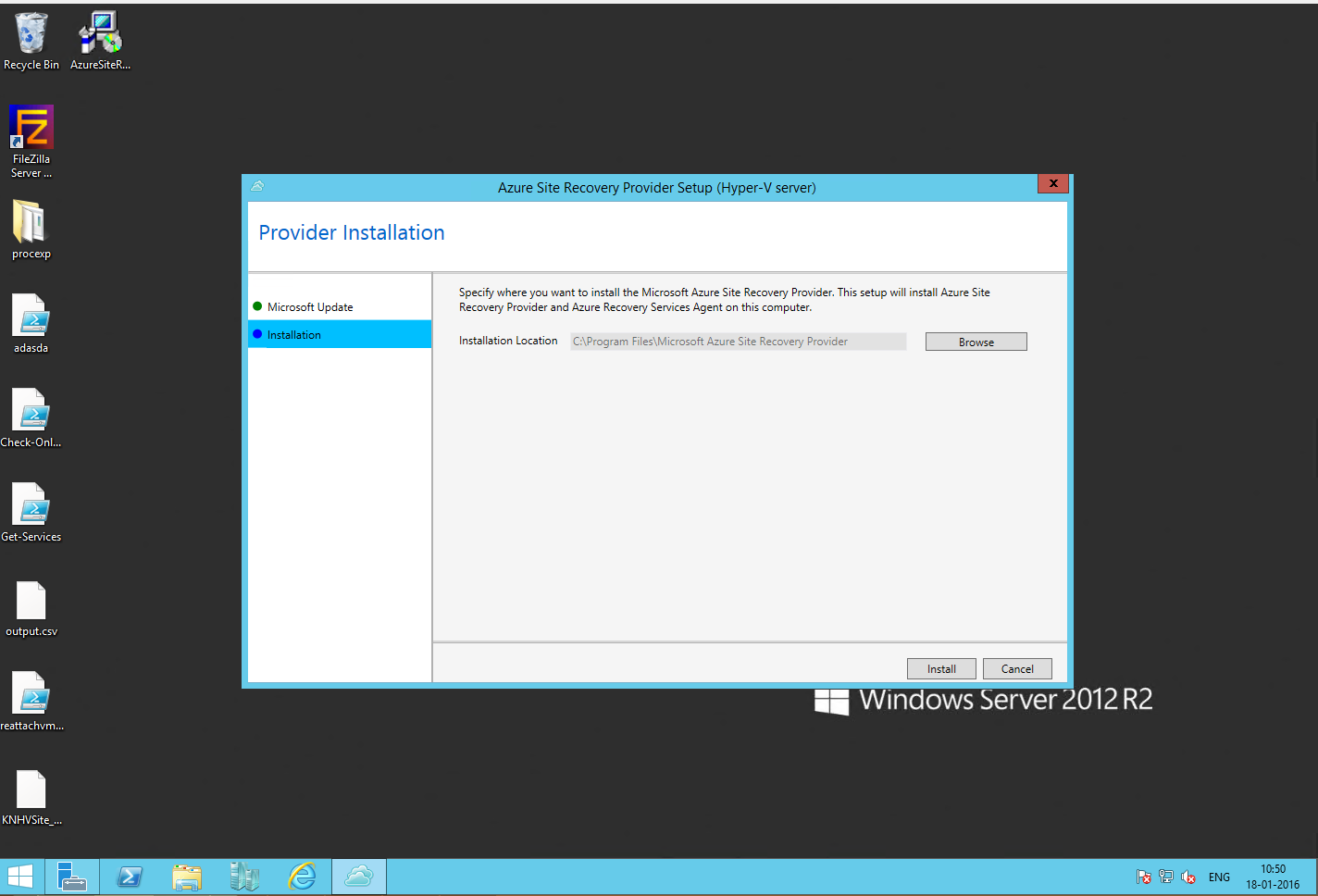
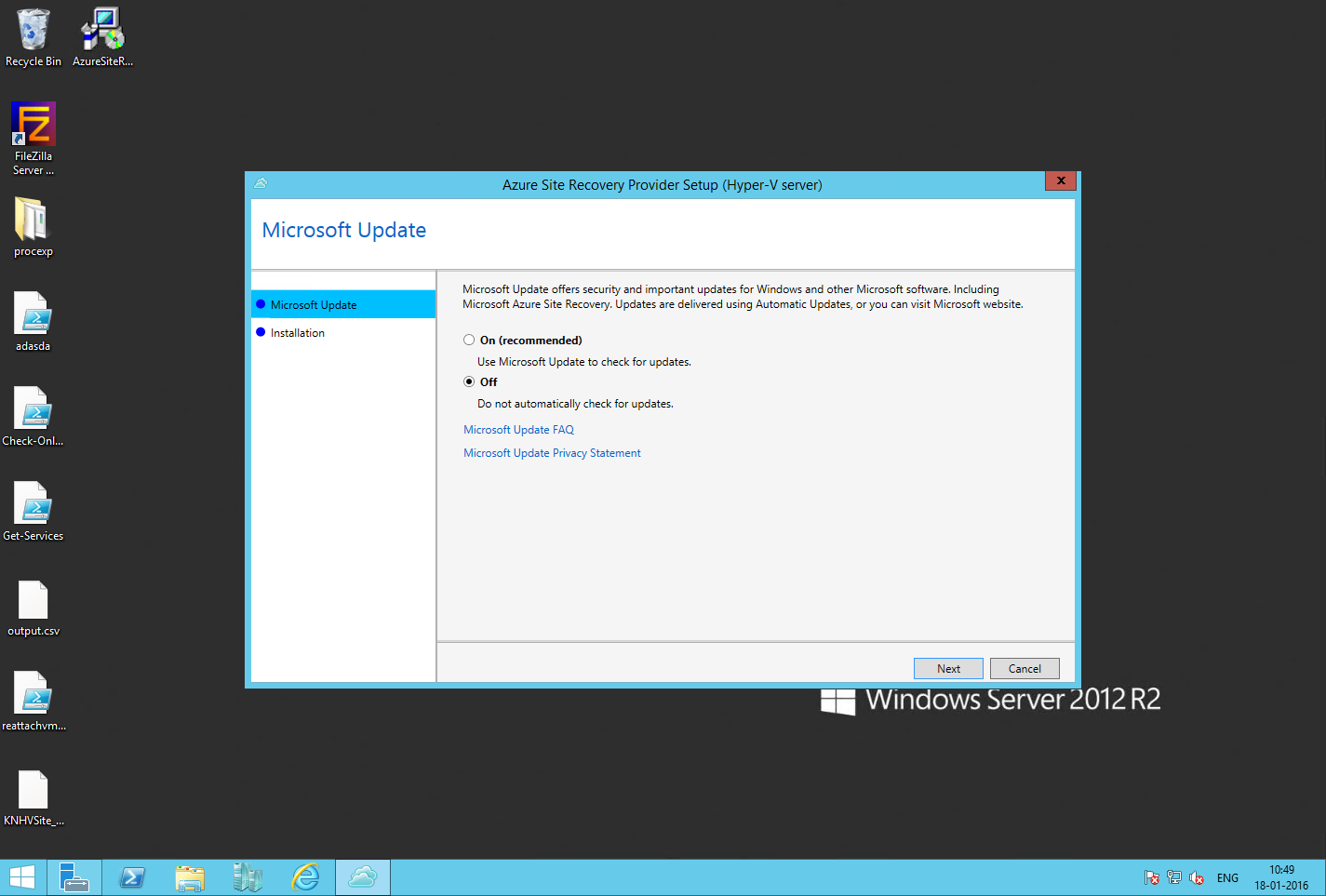
Get-AzureRmSiteRecoveryJob

1. Generate and download registration key for the site, and then copy the downloaded key to your Hyper-V host afterwards as you need it during the agent registration

$SiteIdentifier = Get-AzureRmSiteRecoverySite -Name $sitename | Select -ExpandProperty SiteIdentifier

Get-AzureRmRecoveryServicesVaultSettingsFile -Vault $vault -SiteIdentifier $SiteIdentifier -SiteFriendlyName $sitename -Path $Path

1. On your Hyper-V host, download the installer for the latest version of the Provider from <https://aka.ms/downloaddra>
2. Run the installer on your host and complete the registration step by following the example screenshots:



1. When the above steps completed successfully, you should query the ASR RP in Azure to verify that your server is registered. This can be achieved using the following cmdlet:

$server = Get-AzureRmSiteRecoveryServer -FriendlyName $server.FriendlyName

PS C:\> $server

FriendlyName : HyperLuma.LumaLab.dk

Name : ead02c10-3b63-42c9-8f08-51e63e7330a2

ID : /Subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR/providers/Microsoft.RecoveryServices/vaults/HyperV/replicationFabrics/

KNHVSite/replicationRecoveryServicesProviders/ead02c10-3b63-42c9-8f08-51e63e7330a2

Type : Microsoft.RecoveryServices/vaults/replicationFabrics/replicationRecoveryServicesProviders

FabricType : HyperVSite

Connected : True

LastHeartbeat : 18-Jan-16 10:46:05 AM

ProviderVersion : 5.1.1400.0

ServerVersion : 3.2.7510.0

1. For being able to replicate virtual machines to Azure, we must also create a storage account with geo-replication enabled if it’s not already present within the same Azure Region on your subscription. It is important that this storage account is created with ARM for the VMs to appear in the new portal during failover. If it’s created in the classic model, the VMs will not appear as expected in the new portal. In this example, we will create a new storage account in the same Resource Group containing the ASR vault, using the following cmdlet:

$storageaccountID = New-AzureRmStorageAccount -ResourceGroupName $RG -Name "knasrstorage01" -Type Standard\_GRS -Location $Location

ResourceGroupName : bcdr

StorageAccountName : knasrstorage01

Id : /subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/bcdr/providers/Microsoft.Storage/storageAccounts/knasrstorage01

Location : westeurope

AccountType : StandardGRS

CreationTime : 18-Jan-16 10:51:04 AM

CustomDomain :

LastGeoFailoverTime :

PrimaryEndpoints : Microsoft.Azure.Management.Storage.Models.Endpoints

PrimaryLocation : westeurope

ProvisioningState : Succeeded

SecondaryEndpoints :

SecondaryLocation : northeurope

StatusOfPrimary : Available

StatusOfSecondary : Available

Tags : {}

Context : Microsoft.WindowsAzure.Commands.Common.Storage.AzureStorageContext

1. Create a replication policy and associate it with the protection container using the following cmdlets

$ReplicationFrequencyInSeconds = "300"

$PolicyName = “replicapolicy”

$Recoverypoints = 1

$storageaccountID = Get-AzureRmStorageAccount -Name "knasrstorage01" -ResourceGroupName $RG | Select -ExpandProperty Id

$PolicyResult = New-AzureRmSiteRecoveryPolicy -Name $PolicyName -ReplicationProvider “HyperVReplicaAzure” -ReplicationFrequencyInSeconds $ReplicationFrequencyInSeconds -RecoveryPoints $Recoverypoints -ApplicationConsistentSnapshotFrequencyInHours 1 -RecoveryAzureStorageAccountId $storageaccountID

Name : 5a14f3b4-7324-43b8-a0c5-5e1f07a08e54

ID : /Subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR/providers/Microsoft.RecoveryServices/vaults/HyperV/replicationJobs/5a14f3b4-7324-43b8-a0c5-5e1f07a08e54

Type : Microsoft.RecoveryServices/vaults/replicationJobs

JobType : AddProtectionProfile

DisplayName : Configuring protection group

ClientRequestId : 1b60fcb9-2520-40cc-bac3-8f94236e1402-2016-01-18 11:10:34Z-P ActivityId: c252e7fc-b540-4fab-8fd4-39de0e91e637

State : Succeeded

StateDescription : Completed

StartTime : 18-Jan-16 12:10:32 PM

EndTime : 18-Jan-16 12:10:32 PM

TargetObjectId : be2ea5f3-d82f-4c7d-81bb-57f0c8a0b797

TargetObjectType : ProtectionProfile

TargetObjectName : replicapolicy

AllowedActions : {Cancel, Restart}

Tasks : {AddProtectionProfilePreflightsCheckTask, AddProtectionProfileTask}

Errors : {}

Get the protection container and start the association using the following cmdlets:

$protectionContainer = Get-AzureRmSiteRecoveryProtectionContainer

$Policy = Get-AzureRmSiteRecoveryPolicy -FriendlyName $PolicyName

$associationJob = Start-AzureRmSiteRecoveryPolicyAssociationJob -Policy $Policy -PrimaryProtectionContainer $protectionContainer

**Part 3 – Enabling protection for Virtual Machines**

Once the environment and pre-reqs are configured in Azure and on the Hyper-V host, you can start protect your running VMs in Hyper-V

1. Verify that the VM(s) on your Hyper-V host are visible in ASR by running the following cmdlets:

$VMFriendlyName = "KN02"

$protectionEntity = Get-AzureRmSiteRecoveryProtectionEntity -ProtectionContainer $protectionContainer -FriendlyName $VMFriendlyName

FriendlyName : KN02

Name : 703fd16c-8bce-4b28-b32e-d58724cd261c

ID : /Subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR/providers/Microsoft.RecoveryServices/vaults/HyperV/replicationFabrics/KNHVSite/replicationProtectionContainers/cloud\_646de81b-2f5c-4b73-ad37-8

5800b9b35f1/replicationProtectableItems/703fd16c-8bce-4b28-b32e-d58724cd261c

Type :

FabricObjectId : 703fd16c-8bce-4b28-b32e-d58724cd261c

ProtectionContainerId : cloud\_646de81b-2f5c-4b73-ad37-85800b9b35f1

ProtectionStatus : Unprotected

AllowedOperations :

ActiveLocation :

ProtectionStateDescription :

ReplicationHealth :

TestFailoverStateDescription :

Policy :

OSDiskId :

OSDiskName :

OS :

Disks : {KN02}

ReplicationProvider :

1. Next, we enable protection for the selected virtual machine using the following cmdlets:

$Ostype = "Windows"

$DRjob = Set-AzureRmSiteRecoveryProtectionEntity -ProtectionEntity $protectionEntity -Policy $Policy -Protection Enable -RecoveryAzureStorageAccountId $storageaccountID -OS $OStype -OSDiskName $protectionEntity.Disks[0].Name

Name : 5fd745b8-5d98-4b70-a01c-56fd4d709a9b

ID : /Subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR/providers/Microsoft.RecoveryServices/vaults/HyperV/replicationJobs/5fd745b8-5d98-4b70-a01c-56fd4d709a9b

Type : Microsoft.RecoveryServices/vaults/replicationJobs

JobType : EnableDr

DisplayName : Enable protection

ClientRequestId : f5437822-fc2e-4bd1-933f-1ffb07829403-2016-01-19 12:06:06Z-P ActivityId: 4f657456-302b-4aed-8ea5-427c4fe4402a

State : InProgress

StateDescription : InProgress

StartTime : 19-Jan-16 01:06:02 PM

EndTime :

TargetObjectId : 703fd16c-8bce-4b28-b32e-d58724cd261c

TargetObjectType : ProtectionEntity

TargetObjectName : KN02

AllowedActions : {Cancel, Restart}

Tasks : {EnableProtectionPrerequisitesCheck, CreateProtectionTarget, EnableProtection, VmStartInitialReplication...}

Errors : {}

Errors : {}

1. Monitor the process to see when the VM has completed the initial failover, using one of the examples below:

$DRjob = Get-AzureRmSiteRecoveryJob -Job $DRjob

$DRjob | Select-Object -ExpandProperty State Succeeded

$DRjob | Select-Object -ExpandProperty StateDescription Completed

1. Once complete, complete the remaining steps to create a virtual network in Azure that the VM will connect to during failover, and select the VM size to use. You will update the VM configuration during this process when following these cmdlets:

$Subnet1Name = "Subnet-1"

$vNetName = "KNASRVnet01"

$vNetAddressPrefix = "192.168.0.0/16"

$vNetSubnetAddressPrefix = "192.168.0.0/24"

$SubnetConfig = New-AzureRmVirtualNetworkSubnetConfig -Name $Subnet1Name -AddressPrefix $vNetSubnetAddressPrefix

$vNET = New-AzureRmVirtualNetwork -Name $vNetName -ResourceGroupName $RG -Location $Location -AddressPrefix $vNetAddressPrefix -Subnet $SubnetConfig

$VM = Get-AzureRmSiteRecoveryVM -ProtectionContainer $protectionContainer -FriendlyName $VMFriendlyName

$UpdateJob = Set-AzureRmSiteRecoveryVM -VirtualMachine $VM –Size “Standard\_A5” -PrimaryNic $VM.NicDetailsList[0].NicId -RecoveryNetworkId $vNET.Id -RecoveryNicSubnetName $vNET.Subnets[0].Name

$UpdateJob = Get-AzureRmSiteRecoveryJob -Job $UpdateJob

$UpdateJob

Name : 0344000c-718f-45e5-9c3a-83f39a65e43e

ID : /Subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR/providers/Microsoft.RecoveryServices/vaults/HyperV/replicationJobs/0344000c-718f-45e5-9c3a-83f39a65e43e

Type : Microsoft.RecoveryServices/vaults/replicationJobs

JobType : UpdateVmProperties

DisplayName : Update the virtual machine

ClientRequestId : d989406d-2179-4bb8-a2e7-e2c19770b1e0-2016-01-19 14:06:02Z-P ActivityId: fd9e56e2-00d9-4724-ad42-6e8d33d60c5e

State : Succeeded

StateDescription : Completed

StartTime : 19-Jan-16 03:05:58 PM

EndTime : 19-Jan-16 03:05:59 PM

TargetObjectId : 703fd16c-8bce-4b28-b32e-d58724cd261c

TargetObjectType : ProtectionEntity

TargetObjectName : KN02

AllowedActions : {Restart}

Tasks : {UpdateVmPropertiesTask}

Errors : {}

1. Our next task is to perform a DR drill using the test failover scenario. This will connect the VM to the newly created virtual network in Azure. Use the following cmdlet

$TFjob = Start-AzureRmSiteRecoveryTestFailoverJob -ProtectionEntity $protectionEntity -Direction PrimaryToRecovery -AzureVMNetworkId $vNET.Id

Get-AzureRmSiteRecoveryJob -Job $TFjob

Name : 287f0153-a64d-4eb7-8342-7f2d9315b0cd

ID : /Subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR/providers/Microsoft.RecoveryServices/vaults/HyperV/replicationJobs/287f0153-a64d-4eb7-8342-7f2d9315b0cd

Type : Microsoft.RecoveryServices/vaults/replicationJobs

JobType : TestFailover

DisplayName : Test failover

ClientRequestId : 66b61280-23a5-4685-8e85-b63f3c99d71e-2016-01-19 14:07:03Z-P ActivityId: 915ee206-723a-452e-82a6-8e299f5348ad

State : Suspended

StateDescription : WaitingForStopTestFailover

StartTime : 19-Jan-16 03:06:59 PM

EndTime :

TargetObjectId : 703fd16c-8bce-4b28-b32e-d58724cd261c

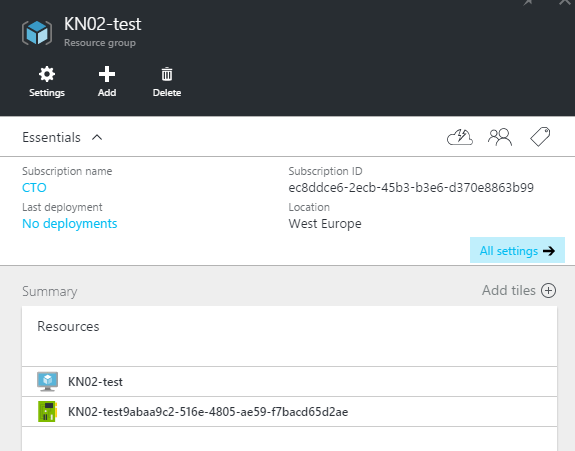
TargetObjectType : ProtectionEntity

TargetObjectName : KN02

AllowedActions : {Cancel, RestartTestFailoverCleanup}

Tasks : {VmTfoPrerequisitesCheck, VmTfoNetworkCreation, VmTfoInitiateFailover, VmTfoAttachNetwork...}

Errors : {}

1. This will now create a new Resource Group in Azure (VMname-test) that contains the Virtual Machine and the vNic created. Storage and Network is located in the Resource Group you created earlier. 

For this drill, we didn’t provide any public IP to the vNic, so a jump host in the same network would be required in order to get access.

Once the process has completed, you can confirm the test failover and delete the DR drill by executing the following cmdlet:

$TFjob = Resume-AzureRmSiteRecoveryJob -Job $TFjob

Name : 287f0153-a64d-4eb7-8342-7f2d9315b0cd

ID : /Subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR/providers/Microsoft.RecoveryServices/vaults/HyperV/replicationJobs/287f0153-a64d-4eb7-8342-7f2d9315b0cd

Type : Microsoft.RecoveryServices/vaults/replicationJobs

JobType : TestFailover

DisplayName : Test failover

ClientRequestId : 66b61280-23a5-4685-8e85-b63f3c99d71e-2016-01-19 14:07:03Z-P ActivityId: 915ee206-723a-452e-82a6-8e299f5348ad

State : InProgress

StateDescription : InProgress

StartTime : 19-Jan-16 03:06:59 PM

EndTime :

TargetObjectId : 703fd16c-8bce-4b28-b32e-d58724cd261c

TargetObjectType : ProtectionEntity

TargetObjectName : KN02

AllowedActions : {Cancel, RestartTestFailoverCleanup}

Tasks : {VmTfoPrerequisitesCheck, VmTfoNetworkCreation, VmTfoInitiateFailover, VmTfoAttachNetwork...}

Errors : {}

If you look in the portal, you can see that Azure Resource Manager is doing a clean-up of the created resources as part of the DR Drill

You have now successfully completed Lab 1

1. Lab 2: Performing a Planned Failover and assigning RBAC to tenant

In this lab you will perform a planned failover of your protected virtual machine and assign RBAC so that the tenant can access it post failover

**Part 1 – Performing a planned failover**

1. Open up PowerShell and execute the following cmdlets to initiate a planned failover, and track the progress:

Start-AzureRmSiteRecoveryPlannedFailoverJob -ProtectionEntity $protectionEntity -Direction PrimaryToRecovery -Optimize ForDowntime

Get-AzureRmSiteRecoveryJob | Where-Object {$\_.JobType -eq "PlannedFailover"}

Name : 487dbc35-e636-41e7-ad12-f7f95bbee0a8

ID : /Subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/BCDR/providers/Microsoft.RecoveryServices/vaults/HyperV/replicationJobs/487dbc35-e636-41e7-ad12-f7f95bbee0a8

Type : Microsoft.RecoveryServices/vaults/replicationJobs

JobType : PlannedFailover

DisplayName : Planned failover

ClientRequestId : fefb8218-98f7-4e1f-8faf-90032b98bae8-2016-01-19 14:47:57Z-P ActivityId: 8a97ec96-220e-4c56-aa4c-4cd92f0628e3

State : Succeeded

StateDescription : Completed

StartTime : 19-Jan-16 03:47:53 PM

EndTime : 19-Jan-16 03:53:51 PM

TargetObjectId : 703fd16c-8bce-4b28-b32e-d58724cd261c

TargetObjectType : ProtectionEntity

TargetObjectName : KN02

AllowedActions : {Cancel}

Tasks : {VmPfoPrerequisitesCheck, VmShutdownVmOnPrimary, VmPrepareForFailover, VmFailover...}

Errors : {}

1. Once the failover has completed, we will perform several tasks in order to get RDP access to our VM in Azure. Follow these examples but remember to modify the variables and names according to your own configuration. We will create a Network Security Group, security rules, public IP address and associate all of that with the virtual network interface attached to that virtual machine

# Create a NSG for the newly created Resource Group and attach a public IP address to the VM network interface for RDP access

$VM = Get-AzureRmVM -Name "kn02" -ResourceGroupName "KN02"

$VMResourceGroup = Get-AzureRmResourceGroup -Name "KN02"

$VMNetworkInterfaceName = $VM.NetworkInterfaceIDs[0].Split('/')[-1]

$VMNetworkInterfaceObject = Get-AzureRmNetworkInterface -ResourceGroupName $VMResourceGroup.ResourceGroupName -Name $VMNetworkInterfaceName

$NSGname = "DRNSG"

$NSG = New-AzureRmNetworkSecurityGroup -Name $NSGname -ResourceGroupName $VMResourceGroup.ResourceGroupName -Location "west europe"

$NSG | Add-AzureRmNetworkSecurityRuleConfig -Name "EnableRDP" -Direction Inbound -Protocol Tcp -Priority 1000 -Access Allow -SourcePortRange '\*' -SourceAddressPrefix '\*' -DestinationAddressPrefix '\*' -DestinationPortRange '3389' | Set-AzureRmNetworkSecurityGroup

# Allocating a public IP address and assign it to the network interface of the VM

$PIP = New-AzureRmPublicIpAddress -Name "drpublic" -ResourceGroupName $VMResourceGroup.ResourceGroupName -Location "west europe" -AllocationMethod Dynamic

$VMNetworkInterfaceObject.IpConfigurations[0].PublicIpAddress = $pip

Set-AzureRmNetworkInterface -NetworkInterface $VMNetworkInterfaceObject

# Associate NSG with the vNic

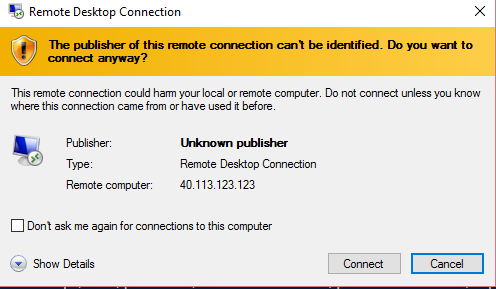
$VMNetworkInterfaceObject.NetworkSecurityGroup = $NSG

Set-AzureRmNetworkInterface -NetworkInterface $VMNetworkInterfaceObject

1. Our last step is to verify access to the virtual machine using RDP. If the VM wasn’t enabled with RDP on Hyper-V prior to the failover, this will block access.

Run the following cmdlet to launch the RDP file:

Get-AzureRmRemoteDesktopFile -ResourceGroupName $VMResourceGroup.ResourceGroupName -Name KN02 -Launch



1. Log in and verify that the operating system work as expected and that any workload/files you had on your machine is fully functional and present

You have now successfully completed part 1.

**Part 2 – Assign RBAC to tenant**

1. In PowerShell, run the following cmdlet to assign access to the Resource Group containing the virtual machine your tenant should be able to manage

New-AzureRmRoleAssignment -SignInName "ryan.irujo@lumagate.com" -ResourceGroupName $VMResourceGroup.ResourceGroupName -RoleDefinitionName "Virtual Machine Contributor"

RoleAssignmentId : /subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/KN02/providers/Microsoft.Authorization/roleAssignments/7c258d20-8ed2-4dd1-85c8-47b75c928cb0

Scope : /subscriptions/ec8ddce6-2ecb-45b3-b3e6-d370e8863b99/resourceGroups/KN02

DisplayName : Ryan Irujo

SignInName : ryan.irujo@lumagate.com

RoleDefinitionName : Virtual Machine Contributor

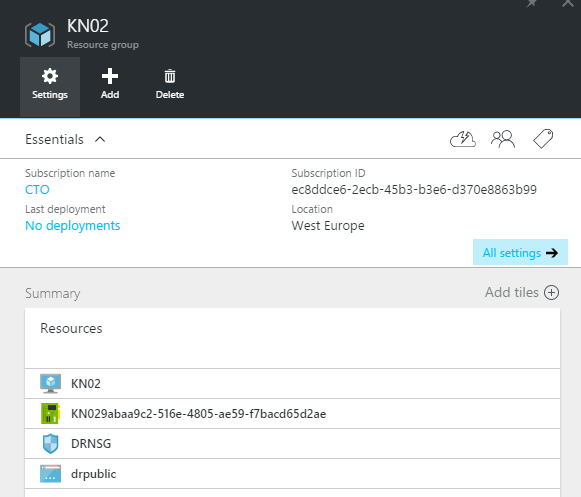
RoleDefinitionId : 9980e02c-c2be-4d73-94e8-173b1dc7cf3c

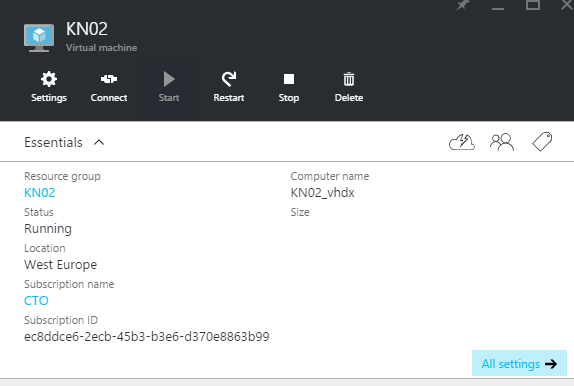
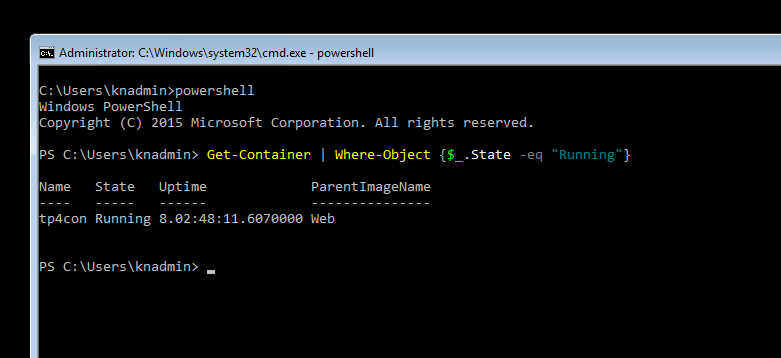
ObjectId : 708f6158-6781-4e62-9414-6a11505d3f73

ObjectType : User

**Part 3 – Accessing VM post failover**

1. In the Azure portal, log in as your tenant and verify that you have access to the virtual machine using RDP



1. Click on the VM object, and connect to it directly from the portal. 
2. Verify that the machine is operating as expected and that you have the content you would expect 

You have now successfully completed lab 2, assigning RBAC to a Resource Group and accessing the resources as a tenant.