Move a virtual machine in an availability zone using Azure PowerShell and CLI

Article • 09/27/2023

This article details using Azure PowerShell and CLI cmdlets to move Azure single instance VMs from regional to zonal availability zones. An availability zone is a physically separate zone in an Azure region. Use availability zones to protect your apps and data from an unlikely failure or loss of an entire data center.

To use an availability zone, create your virtual machine in a supported Azure region.

> ⓘ **Important**
>
> Regional to zonal move of single instance VM(s) configuration is currently in *Public Preview*.

Prerequisites

Verify the following requirements:

Requirement	Description
Subscription permissions	Ensure you have *Owner* access on the subscription containing the resources that you want to move. Managed identity needs these permissions: - Permission to write or create resources in user subscription, available with the *Contributor role*. - Permission to create role assignments. Typically available with the *Owner* or *User Access Administrator* roles, or with a custom role that has the `Microsoft.Authorization` role assignments or write permission assigned. This permission isn't needed if the data share resource's managed identity is already granted access to the Azure data store. Learn more about Azure roles.
VM support	Review the supported regions. - Check supported compute, storage, and networking settings.
VM health status	The VMs you want to move must be in a healthy state before attempting the zonal move. Ensure that all pending reboots and mandatory updates are

Requirement	Description
	complete and the Virtual Machine is working and is in a healthy state before attempting the VM zonal move.

Review PowerShell and CLI requirements

Most move resources operations are the same whether using the Azure portal or PowerShell or CLI, with a couple of exceptions.

Operation	Portal	PowerShell/CLI
Create a move collection	A move collection (a list of all the regional VMs that you're moving) is created automatically. Required identity permissions are assigned in the backend by the portal.	You can use PowerShell cmdlets or CLI cmdlets to: - Assign a managed identity to the collection. - Add regional VMs to the collection.
Resource move operations	Validate steps and validates the *User* setting changes. **Initiate move** starts the move process and creates a copy of source VM in the target zone. It also finalizes the move of the newly created VM in the target zone.	PowerShell cmdlets or CLI cmdlets to: - Add regional VMs to the collection - Resolve dependencies - Perform the move. - Commit the move.

Sample values

We use these values in our script examples:

Setting	Value
Subscription ID	subscription-id
Move Region	East US
Resource group (holding metadata for move collection)	RegionToZone-DemoMCRG
Move collection name	RegionToZone-DemoMC
Location of the move collection	eastus2euap
IdentityType	SystemAssigned

Setting	Value
VM name	demoVM-MoveResource
Move Type	RegionToZone

Sign in to Azure

Sign in to your Azure subscription with the `Connect-AzAccount` command and follow the on-screen directions.

PowerShell

```
Connect-AzAccount –Subscription "<subscription-id>"
```

Set up the move collection

The MoveCollection object stores metadata and configuration information about the resources you want to move. To set up a move collection, do the following:

- Create a resource group for the move collection.
- Register the service provider to the subscription, so that the MoveCollection resource can be created.
- Create the MoveCollection object with managed identity. For the MoveCollection object to access the subscription in which the Resource Mover service is located, it needs a system-assigned managed identity (formerly known as Managed Service Identity (MSI)) that's trusted by the subscription.
- Grant access to the Resource Mover subscription for the managed identity.

Create the resource group

Use the following cmdlet to create a resource group for the move collection metadata and configuration information with New-AzResourceGroup. A resource group is a logical container into which Azure resources are deployed and managed.

PowerShell

PowerShell

```
New-AzResourceGroup -Name "RegionToZone-DemoMCRG" -Location "EastUS"
```

Output:

The output shows that the managed disk is in the same availability zone as the VM:

PowerShell

```
ResourceGroupName : RegionToZone-DemoMCRG
Location          : eastus
ProvisioningState : Succeeded
Tags              :
                    Name     Value
                    =======  ========
                    Created  20230908

ResourceId        : /subscriptions/<Subscription-
id>/resourceGroups/RegionToZone-DemoMCRG
```

Register the resource provider

1. Register the resource provider Microsoft.Migrate, so that the MoveCollection resource can be created, as follows:

Azure PowerShell

```
Register-AzResourceProvider -ProviderNamespace Microsoft.Migrate
```

2. Wait for registration:

Azure PowerShell

```
While(((Get-AzResourceProvider -ProviderNamespace Microsoft.Migrate)|
where {$_.RegistrationState -eq "Registered" -and
$_.ResourceTypes.ResourceTypeName -eq
"moveCollections"}|measure).Count -eq 0)
{
    Start-Sleep -Seconds 5
    Write-Output "Waiting for registration to complete."
}
```

Create a MoveCollection object

Create a MoveCollection object, and assign a managed identity to it, as follows:

PowerShell

Azure PowerShell

```
New-AzResourceMoverMoveCollection -Name "RegionToZone-DemoMC"  -
ResourceGroupName "RegionToZone-DemoMCRG" -MoveRegion "eastus" -
Location "eastus2euap" -IdentityType "SystemAssigned" -MoveType
"RegionToZone"
```

Output:

PowerShell

```
Etag                                    Location    Name
----                                    --------    ----
"3a00c441-0000-3400-0000-64fac1b30000" eastus2euap RegionToZone-DemoMC
```

ⓘ **Note**

For Regional to zonal move, the `MoveType` parameter should be set as *RegionToZone* and `MoveRegion` parameter should be set as the location where resources undergoing zonal move reside. Ensure that the parameters `SourceRegion` and `TargetRegion` are not required and should be set to *null*.

Grant access to the managed identity

Grant the managed identity access to the Resource Mover subscription as follows. You must be the subscription owner.

1. Retrieve identity details from the MoveCollection object.

   Azure PowerShell

   ```
   $moveCollection = Get-AzResourceMoverMoveCollection -Name
   "RegionToZone-DemoMC" -ResourceGroupName "RegionToZone-DemoMCRG"
   $identityPrincipalId = $moveCollection.IdentityPrincipalId
   ```

2. Assign the required roles to the identity so Azure Resource Mover can access your subscription to help move resources. Review the list of required permissions for the move.

   PowerShell

Azure PowerShell

```
New-AzRoleAssignment -ObjectId $identityPrincipalId -
RoleDefinitionName Contributor -Scope
"/subscriptions/<subscription-id>""
New-AzRoleAssignment -ObjectId $identityPrincipalId -
RoleDefinitionName "User Access Administrator" -Scope
"/subscriptions/<subscription-id>"
```

Add regional VMs to the move collection

Retrieve the IDs for existing source resources that you want to move. Create the destination resource settings object, then add resources to the move collection.

> ⓘ **Note**
>
> Resources added to a move collection must be in the same subscription but can be in different resource groups.

1. Create target resource setting object as follows:

Azure PowerShell

```
$targetResourceSettingsObj = New-Object
Microsoft.Azure.PowerShell.Cmdlets.ResourceMover.Models.Api20230801.Vir
tualMachineResourceSettings
$targetResourceSettingsObj.ResourceType =
"Microsoft.Compute/virtualMachines"
$targetResourceSettingsObj.TargetResourceName = "RegionToZone-demoTar-
getVm"
$targetResourceSettingsObj.TargetAvailabilityZone = "2"
```

Output

PowerShell

```
ResourceType                      TargetResourceGroupName
TargetResourceName        TargetAvailabilitySetId
TargetAvailabilityZone TargetVMSize UserManagedIdentity
------------                      ----------------------- -------------
-----        ---------------------- --------------------- -----------
- -------------------
Microsoft.Compute/virtualMachines                          RegionToZone-
demoTargetVm                          2
```

2. Add resources

PowerShell

Azure PowerShell

```
Add-AzResourceMoverMoveResource -ResourceGroupName "RegionToZone-
DemoMCRG" -MoveCollectionName "RegionToZone-DemoMC" -SourceId
"/subscriptions/<Subscription-id>/resourcegroups/PS-demo-
RegionToZone-
RG/providers/Microsoft.Compute/virtualMachines/RegionToZone-de-
moSourceVm" -Name "demoVM-MoveResource" -ResourceSetting
$targetResourceSettingsObj
```

Output

PowerShell

```
DependsOn                         : {}
DependsOnOverride                 : {}
ErrorsPropertiesCode              :
ErrorsPropertiesDetail            :
ErrorsPropertiesMessage           :
ErrorsPropertiesTarget            :
ExistingTargetId                  :
Id                                : /subscriptions/<Subscription-
id>/resourceGroups/RegionToZone-
DemoMCRG/providers/Microsoft.Migrate/moveCollections/Re
                                    gionToZone-
DemoMC/moveResources/demoVM-MoveResource
IsResolveRequired                 : False
JobStatusJobName                  :
JobStatusJobProgress              :
MoveStatusErrorsPropertiesCode    : DependencyComputationPending
MoveStatusErrorsPropertiesDetail  : {}
MoveStatusErrorsPropertiesMessage : The dependency computation is
not completed for resource - /subscriptions/<Subscription-
id>/resourcegroups/PS-demo-R
                                    egionToZone-
RG/providers/Microsoft.Compute/virtualMachines/RegionToZone-de-
moSourceVm'.
                                        Possible Causes: Dependency
computation is pending for resource.
                                        Recommended Action:
Validate dependencies to compute the dependencies.

MoveStatusErrorsPropertiesTarget  :
MoveStatusMoveState               : MovePending
Name                              : demoVM-MoveResource
ProvisioningState                 : Succeeded
ResourceSetting                   :
```

```
Microsoft.Azure.PowerShell.Cmdlets.ResourceMover.Models.Api20230801
.VirtualMachineResourceSettings
SourceId                          : /subscriptions/<Subscription-
id>/resourcegroups/PS-demo-RegionToZone-
RG/providers/Microsoft.Compute/virtualMachines/
                                    RegionToZone-demoSourceVm
SourceResourceSetting             :
Microsoft.Azure.PowerShell.Cmdlets.ResourceMover.Models.Api20230801
.VirtualMachineResourceSettings
SystemDataCreatedAt               : 9/8/2023 6:48:11 AM
SystemDataCreatedBy               : xxxxx@microsoft.com
SystemDataCreatedByType           : User
SystemDataLastModifiedAt          : 9/8/2023 6:48:11 AM
SystemDataLastModifiedBy          : xxxxx@microsoft.com
SystemDataLastModifiedByType      : User
TargetId                          :
Type                              :
```

Modify settings

You can modify destination settings when moving Azure VMs and associated resources. We recommend that you only change destination settings before you validate the move collection.

Settings that you can modify are:

- **Virtual machine settings:** Resource group, VM name, VM availability zone, VM SKU, VM key vault, and Disk encryption set.
- **Networking resource settings:** For Network interfaces, virtual networks (VNets/), and network security groups/network interfaces, you can either:
  - Use an existing networking resource in the destination region.
  - Create a new resource with a different name.
- **Public IP/Load Balancer:** SKU and Zone

Modify settings as follows:

1. Retrieve the move resource for which you want to edit properties. For example, to retrieve a VM run:

   Azure PowerShell

   ```
   $moveResourceObj = Get-AzResourceMoverMoveResource -MoveCollectionName
   " RegionToZone-DemoMCRG " -ResourceGroupName " RegionToZone-DemoMC " -
   Name "PSDemoVM"
   ```

2. Copy the resource setting to a target resource setting object.

Azure PowerShell

```
$TargetResourceSettingObj = $moveResourceObj.ResourceSetting
```

3. Set the parameter in the target resource setting object. For example, to change the name of the destination VM:

Azure PowerShell

```
$TargetResourceSettingObj.TargetResourceName="PSDemoVM-target"
```

4. Update the move resource destination settings. In this example, we change the name of the VM from PSDemoVM to PSDemoVMTarget.

Azure PowerShell

```
Update-AzResourceMoverMoveResource -ResourceGroupName " RegionToZone-
DemoMCRG " -MoveCollectionName " RegionToZone-DemoMC -SourceId
"/subscriptions/<Subscription-
d>/resourceGroups/PSDemoRM/providers/Microsoft.Compute/virtualMachines/
PSDemoVM" -Name "PSDemoVM" -ResourceSetting $TargetResourceSettingObj
```

Resolve dependencies

Check whether the regional VMs you added have any dependencies on other resources, and add as needed.

1. Resolve dependencies as follows:

PowerShell

```
Resolve-AzResourceMoverMoveCollectionDependency -ResourceGroupName
"RegionToZone-DemoMCRG" -MoveCollectionName "RegionToZone-DemoMC"
```

Output (when dependencies exist)

PowerShell

```
AdditionalInfo :
Code           :
Detail         :
EndTime        : 9/8/2023 6:52:14 AM
```

```
Id               : /subscriptions/<Subscription-
id>/resourceGroups/RegionToZone-
DemoMCRG/providers/Microsoft.Migrate/moveCollections/RegionToZone-
DemoMC/o
                   perations/bc68354b-ec1f-44cb-92ab-fb3b4ad90229
Message          :
Name             : bc68354b-ec1f-44cb-92ab-fb3b4ad90229
Property         :
Microsoft.Azure.PowerShell.Cmdlets.ResourceMover.Models.Any
StartTime        : 9/8/2023 6:51:50 AM
Status           : Succeeded
```

2. To get a list of resources added to the move collection:

PowerShell

Azure PowerShell

```
$list = Get-AzResourceMoverMoveResource -ResourceGroupName
"RegionToZone-DemoMCRG" -MoveCollectionName "RegionToZone-DemoMC"
$list.Name
```

Output:

PowerShell

```
demoVM-MoveResource
mr_regiontozone-demosourcevm661_d6f18900-3b87-4fb5-9bdf-
12da2f9fb185
mr_regiontozone-demosourcevm-vnet_d8536bf5-2d5f-4778-9650-
32d0570bc41a
mr_regiontozone-demosourcevm-ip_6af03f1f-eae8-4541-83f5-
97a2506cfc3e
mr_regiontozone-demosourcevm-nsg_98d68420-d7ff-4e2d-b758-
25a6df80fca7
mr_nrms-timkbo3hy3nnmregiontozone-demosourcevm-vnet_f474c880-4823-
4ed3-b761-96df6500f6a3
```

3. To remove resources from the resource collection, follow these instructions.

Initiate move of VM resources

PowerShell

Azure PowerShell

```
Invoke-AzResourceMoverInitiateMove -ResourceGroupName "RegionToZone-
DemoMCRG" -MoveCollectionName "RegionToZone-DemoMC" -MoveResource $("de-
moVM-MoveResource") -MoveResourceInputType "MoveResourceId"
```

Output

PowerShell

```
AdditionalInfo :
Code           :
Detail         :
EndTime        : 9/8/2023 7:07:58 AM
Id             : /subscriptions/<Subscription-
id>/resourceGroups/RegionToZone-
DemoMCRG/providers/Microsoft.Migrate/moveCollections/RegionToZone-
DemoMC/o
                 perations/d3e06ac3-a961-4045-8301-aee7f6911160
Message        :
Name           : d3e06ac3-a961-4045-8301-aee7f6911160
Property       :
Microsoft.Azure.PowerShell.Cmdlets.ResourceMover.Models.Any
StartTime      : 9/8/2023 7:01:31 AM
Status         : Succeeded
```

Commit

After the initial move, you must commit the move or discard it. **Commit** completes the move to the target region.

Commit the move as follows:

PowerShell

```
Invoke-AzResourceMover-VMZonalMoveCommit -ResourceGroupName "RG-
MoveCollection-demoRMS" -MoveCollectionName "PS-centralus-westcentralus-
demoRMS" -MoveResource $('psdemovm111', 'PSDemoRM-vnet','PSDemoVM-nsg',
‘PSDemoVM’) -MoveResourceInputType "MoveResourceId"
```

Output:

PowerShell

```
AdditionalInfo :
Code           :
Detail         :
EndTime        : 9/22/2023 5:26:55 AM
Id             : /subscriptions/e80eb9fa-c996-4435-aa32-
5af6f3d3077c/resourceGroups/RegionToZone-
DemoMCRG/providers/Microsoft.Migrate/moveCollections/RegionToZone-
DemoMC/operations/35dd1d93-ba70-4dc9-a17f-7d8ba48678d8
Message        :
Name           : 35dd1d93-ba70-4dc9-a17f-7d8ba48678d8
Property       :
Microsoft.Azure.PowerShell.Cmdlets.ResourceMover.Models.Any
StartTime      : 9/22/2023 5:26:54 AM
Status         : Succeeded
```

Delete source regional VMs

After you commit the move and verify that the resources work as expected in the target region, you can delete each source resource using:

- Azure portal
- PowerShell
- Azure CLI

Next steps

Learn how to move single instance Azure VMs from regional to zonal configuration via portal.