List all available roles

To list RBAC roles available for assignment and to inspect the operations to which they grant access use:

Login-AzAccount

Get-AzRoleDefinition

Get-AzRoleDefinition | ft name, Description

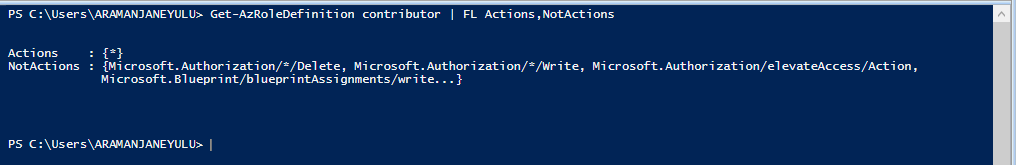


List actions of a role

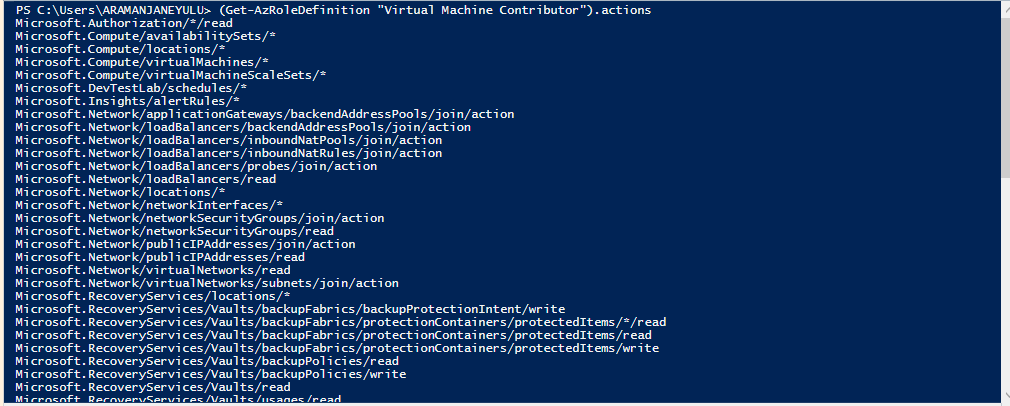
To list the actions for a specific role use:

Get-AzRoleDefinition <role name>

Get-AzRoleDefinition contributor | FL Actions,NotActions



(Get-AzRoleDefinition "Virtual Machine Contributor").actions



List Access

List all role assignments in the selected subscription

To list RBAC access assignments effective at the specified subscription, resource, or resource group use:

Get-AzRoleAssignment

List role assignments effective on a resource group

To list the access assignments for a resource group use:

Get-AzRoleAssignment -ResourceGroupName <resource group name>

Get-AzRoleAssignment -ResourceGroupName DEV\_CUSTOM\_TEST\_RG | FL DisplayName,RoleDefinitionName,Scope



List role assignments to a user, including ones assigned to users groups

To list access assignments to the specified user as well as to the groups of which the user is member use:

Get-AzRoleAssignment –ExpandPrincipalGroups

Get-AzRoleAssignment -SignInName "azadmin\_preddy@eastwestbank32.onmicrosoft.com"

PS C:\Users\ARAMANJANEYULU> Get-AzRoleAssignment -SignInName "azadmin\_preddy@eastwestbank32.onmicrosoft.com"

RoleAssignmentId : /subscriptions/3d8f9927-6b94-4173-88b8-cda9686ea102/providers/Microsoft.Authorization/roleAssignments/671fe5f6-d

645-4ec4-a0a4-22c7be20c427

Scope : /subscriptions/3d8f9927-6b94-4173-88b8-cda9686ea102

DisplayName : Pradeep Reddy

SignInName : azadmin\_preddy@eastwestbank32.onmicrosoft.com

RoleDefinitionName : Contributor

RoleDefinitionId : b24988ac-6180-42a0-ab88-20f7382dd24c

ObjectId : 7853f965-1a82-4d1f-8d57-b863a612e3b6

ObjectType : User

CanDelegate : False

Get-AzRoleAssignment -SignInName "azadmin\_preddy@eastwestbank32.onmicrosoft.com" -ExpandPrincipalGroups

PS C:\Users\ARAMANJANEYULU> Get-AzRoleAssignment -SignInName "azadmin\_preddy@eastwestbank32.onmicrosoft.com" -ExpandPrincipalGroups

RoleAssignmentId : /subscriptions/3d8f9927-6b94-4173-88b8-cda9686ea102/providers/Microsoft.Authorization/roleAssignments/671fe5f6-d

645-4ec4-a0a4-22c7be20c427

Scope : /subscriptions/3d8f9927-6b94-4173-88b8-cda9686ea102

DisplayName : Pradeep Reddy

SignInName : azadmin\_preddy@eastwestbank32.onmicrosoft.com

RoleDefinitionName : Contributor

RoleDefinitionId : b24988ac-6180-42a0-ab88-20f7382dd24c

ObjectId : 7853f965-1a82-4d1f-8d57-b863a612e3b6

ObjectType : User

CanDelegate : False

List classic service administrator and co-admin role assignments

To list access assignments for the classic subscription administrator and co-administrators use:

Get-AzRoleAssignment -IncludeClassicAdministrators

Grant access

Search for object IDs

In order to use the following command sequences, you must find the object IDs first. It is assumed that you already know the subscription ID that you are working with

Find the object ID for an Azure AD Group

To get the object ID for an Azure AD Group use:

Get-AzADGroup -SearchString <group name in quotes>

Find the object ID for an Azure AD Service Principal

To get the object ID for an Azure AD Service Principal use:

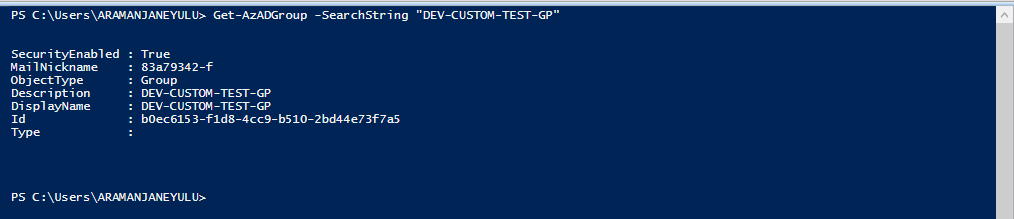
Get-AzADServicePrincipal -SearchString <service name in quotes>

Assign role to group at subscription scope

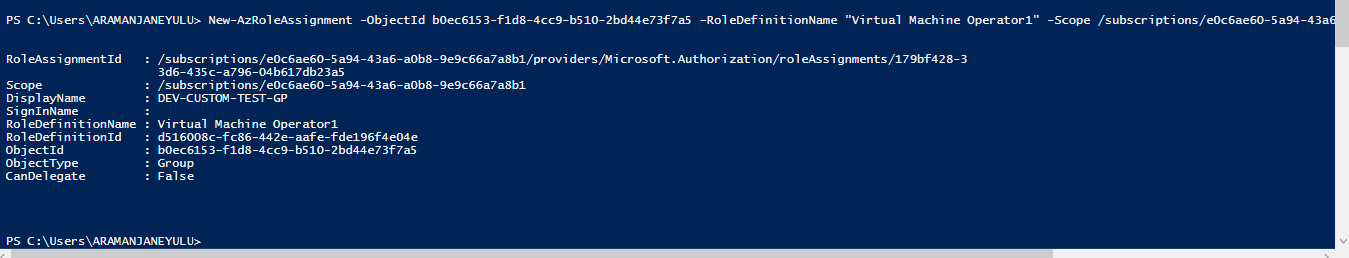
To grant access to a group at subscription scope use:

New-AzRoleAssignment -ObjectId <object id> -RoleDefinitionName <role name in quotes> -Scope <scope such as subscription/subscription id>

Get-AzADGroup -SearchString "DEV-CUSTOM-TEST-GP"



New-AzRoleAssignment -ObjectId b0ec6153-f1d8-4cc9-b510-2bd44e73f7a5 -RoleDefinitionName "Virtual Machine Operator1" -Scope /subscriptions/e0c6ae60-5a94-43a6-a0b8-9e9c66a7a8b1



Assign role to application at subscription scope

To grant access to an application at subscription scope use:

New-AzRoleAssignment -ObjectId <object id> -RoleDefinitionName <role name in quotes> -Scope <scope such as subscription/subscription id>

Assign role to user at resource group scope

To grant access to a user at resource group scope:

New-AzRoleAssignment -SignInName <email of user> -RoleDefinitionName <role name in quotes> -ResourceGroupName <resource group name>

Assign role to group at resource scope

To grant access to a group at the resource scope use:

New-AzRoleAssignment -ObjectId <object id> -RoleDefinitionName <role name in quotes> -ResourceName <resource name> -ResourceType <resource type> -ParentResource <parent resource> -ResourceGroupName <resource group name>

Remove access

To remove access for users, groups and applications use:

Remove-AzRoleAssignment -ObjectId <object id> -RoleDefinitionName <role name> -Scope <scope such as subscription/subscription id>

Remove-AzRoleAssignment -ObjectId "74b007ea-ab34-4209-981e-c538200fe251" -RoleDefinitionName "Virtual Machine Contributor" -Scope "/subscriptions/f0675ec9-480d-4c2a-982a-ed97983af390"

Create custom role

To create a custom role, use the New-AzRoleDefinition command.

The following example creates a custom role called Virtual Machine Operator1 that grants access to all read operations of Microsoft.Compute, Microsoft.Storage, and Microsoft.Network resource providers, and grants access to start, restart, and monitor virtual machines. The custom role can be used in one subscriptions.

Get-AzSubscription

Get-Azcontext

Get-AzSubscription -SubscriptionId e0c6ae60-5a94-43a6-a0b8-9e9c66a7a8b1 | set-Azcontext

Get-AzProviderOperation <operation> | FT OperationName, Operation, Description -AutoSize

Get-AzProviderOperation "Microsoft.Compute/virtualMachines/\*" | FT OperationName, Operation, Description -AutoSize

$role = Get-AzRoleDefinition "Virtual Machine Contributor"

$role.Id = $null

$role.Name = "Virtual Machine Operator1"

$role.Description = "Can monitor and restart virtual machines."

$role.Actions.Clear()

$role.Actions.Add("Microsoft.Storage/\*/read")

$role.Actions.Add("Microsoft.Network/\*/read")

$role.Actions.Add("Microsoft.Compute/\*/read")

$role.Actions.Add("Microsoft.Compute/virtualMachines/start/action")

$role.Actions.Add("Microsoft.Compute/virtualMachines/restart/action")

$role.Actions.Add("Microsoft.Authorization/\*/read")

$role.Actions.Add("Microsoft.ResourceHealth/availabilityStatuses/read")

$role.Actions.Add("Microsoft.Resources/subscriptions/resourceGroups/read")

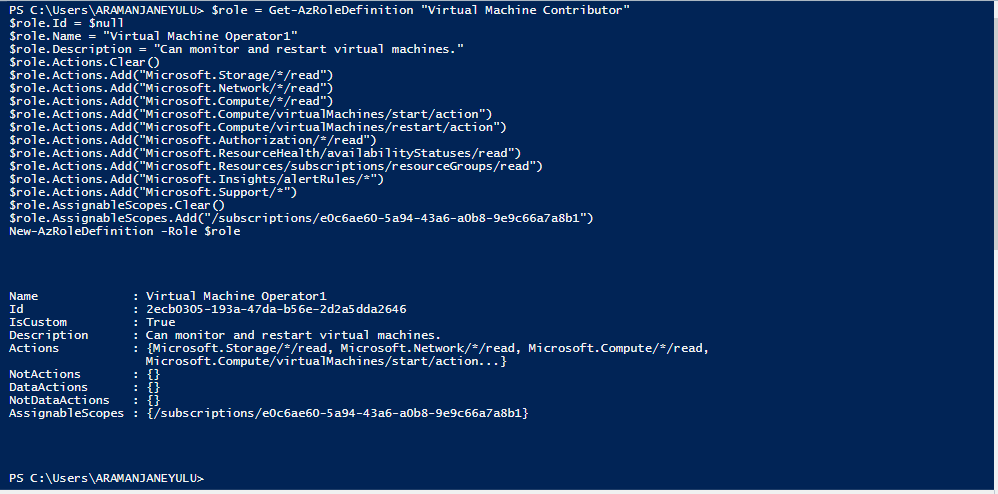
$role.Actions.Add("Microsoft.Insights/alertRules/\*")

$role.Actions.Add("Microsoft.Support/\*")

$role.AssignableScopes.Clear()

$role.AssignableScopes.Add("/subscriptions/e0c6ae60-5a94-43a6-a0b8-9e9c66a7a8b1")

New-AzRoleDefinition -Role $role



Modify a custom role

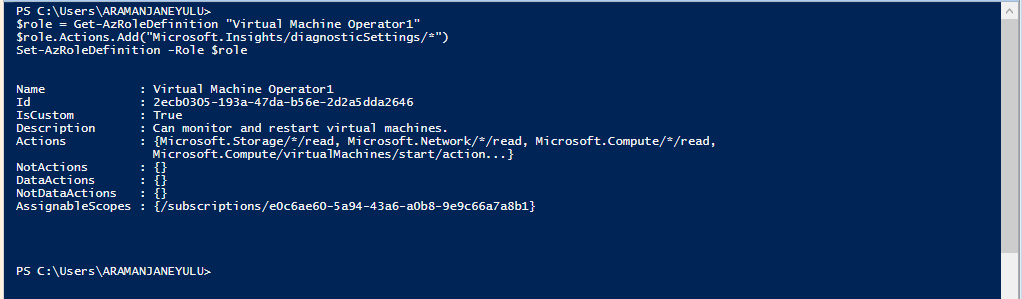
To modify a custom role first, use the **Get-AzRoleDefinition** command to retrieve role definition. Then, make desired changes to the role definition. Finally, use **Set-AzRoleDefinition**command to save the modified role definition.

The following example adds the **Microsoft.Insights/diagnosticSettings/\*** operation to the ***Virtual Machine Operator1*** custom role.

$role = Get-AzRoleDefinition "Virtual Machine Operator1"

$role.Actions.Add("Microsoft.Insights/diagnosticSettings/\*")

Set-AzRoleDefinition -Role $role

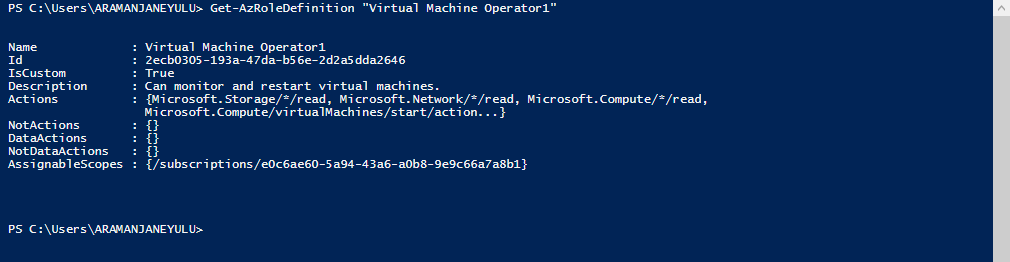


Delete a custom role

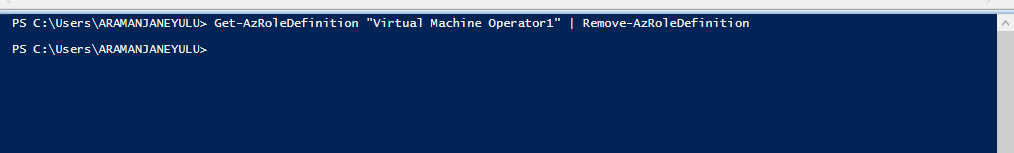
To delete a custom role, use the **Remove-AzRoleDefinition** command.

The following example removes the ***Virtual Machine Operator1*** custom role.

Get-AzRoleDefinition "Virtual Machine Operator1"



Get-AzRoleDefinition "Virtual Machine Operator1" | Remove-AzRoleDefinition

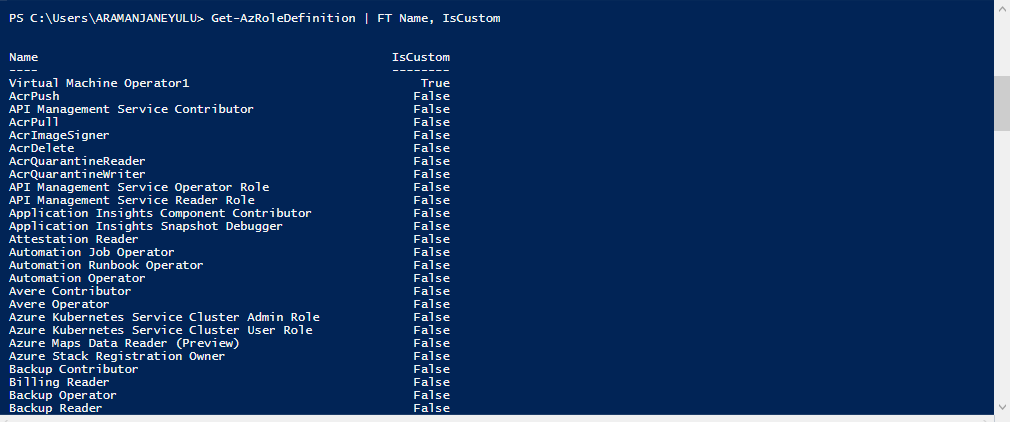


List custom roles

To list the roles that are available for assignment at a scope, use the **Get-AzRoleDefinition** command.

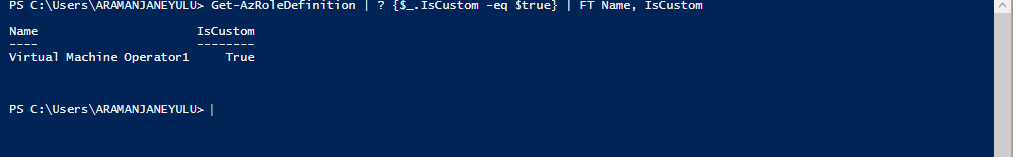
The following example lists all role available for assignment in the selected subscription.

Get-AzRoleDefinition | FT Name, IsCustom



In the following example, the ***Virtual Machine Operator1*** custom role is available in the **EWAZ Development Environment** subscription because that subscription is in the **AssignableScopes** of the role.

Get-AzRoleDefinition | ? {$\_.IsCustom -eq $true} | FT Name, IsCustom



Generate a role-based access control report

Create a report in detail showing all the permissions assigned to users, groups, and applications throughout a subscription.

$result=@()

$result+="displayname,objecttype,roledefinitionname,actions"

Get-AzRoleAssignment -scope "/subscriptions/3d8f9927-6b94-4173-88b8-cda9686ea102"|foreach{

$displayname=$\_.DisplayName

$objecttype=$\_.ObjectType

$roledefinitionname=$\_.RoleDefinitionName

$actions=(Get-AzRoleDefinition -Name $\_.roledefinitionname).actions

$result+="$displayname,$objecttype,$roledefinitionname,$actions"

}

$result | out-file C:\Users\admin\_aramanjaneyulu\Downloads\filename.csv

We’ve had a look at Azure roles and how we can easily manage role assignment tasks for Azure resources.

The following code would allow you to add group membership information if the object type is "user".

$result=@()

$groups=(Get-AzADUser | select id).id.guid

$result+="displayname,objecttype,roledefinitionname,actions,groups"

Get-AzRoleAssignment -scope "/subscriptions/SUBSCTIPTIONID" | foreach{

$displayname=$\_.DisplayName

$objecttype=$\_.ObjectType

if($objecttype -eq "user"){

$objectid=$\_.objectid

$groups=(Get-AzADUserMembership -ObjectId $objectid).objectid.guid

}

$roledefinitionname=$\_.RoleDefinitionName

$actions=(Get-AzRoleDefinition -Name $\_.roledefinitionname).actions

$result+="$displayname,$objecttype,$roledefinitionname,$actions,$groups"

}

$result | out-file C:\Users\admin\_aramanjaneyulu\Downloads\filename.csv

Get the Subscription information

$result=@()

$result+="displayname,objecttype,roledefinitionname,actions"

$subscriptions=(Get-AzSubscription).SubscriptionId

foreach($subscriptionid in $subscriptions){

Get-AzRoleAssignment -scope "/subscriptions/$subscriptionid" | foreach{

$displayname=$\_.DisplayName

$objecttype=$\_.ObjectType

$roledefinitionname=$\_.RoleDefinitionName

$actions=(Get-AzRoleDefinition -Name $\_.roledefinitionname).actions

$result+="$displayname,$objecttype,$roledefinitionname,$actions"

}

}

$result | out-file C:\Users\admin\_aramanjaneyulu\Downloads\filename.csv