Windows PowerShell Get-Help on Cmdlet 'New-AzLoadBalancer'

PS:\>Get-HELP New-AzLoadBalancer -Full

WARNING: The names of some imported commands from the module 'Microsoft.Azure.PowerShell.Cmdlets.Network' include unapproved verbs that might make them less discoverable.

To find the commands with unapproved verbs, run the Import-Module command again with the Verbose parameter. For a list of approved verbs, type Get-Verb.

NAME

New-AzLoadBalancer

SYNOPSIS

Creates a load balancer.

SYNTAX

New-AzLoadBalancer [-AsJob] [-BackendAddressPool <Microsoft.Azure.Commands.Network.Models.PSBackendAddressPool[]>] [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-EdgeZone <System.String>] [-Force] [-FrontendIpConfiguration <Microsoft.Azure.Commands.Network.Models.PSFrontendIPConfiguration[]>] [-InboundNatPool <Microsoft.Azure.Commands.Network.Models.PSInboundNatPool[]>] [-InboundNatRule <Microsoft.Azure.Commands.Network.Models.PSInboundNatRule[]>] [-LoadBalancingRule <Microsoft.Azure.Commands.Network.Models.PSLoadBalancingRule[]>] -Location

<System.String> -Name <System.String> [-OutboundRule <Microsoft.Azure.Commands.Network.Models.PSOutboundRule[]>] [-Probe <Microsoft.Azure.Commands.Network.Models.PSProbe[]>] -ResourceGroupName <System.String> [-Sku <System.String>] [-Tag <System.Collections.Hashtable>] [-Tier <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

DESCRIPTION

The New-AzLoadBalancer cmdlet creates an Azure load balancer.

PARAMETERS

-AsJob <System.Management.Automation.SwitchParameter>

Run cmdlet in the background

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-BackendAddressPool <Microsoft.Azure.Commands.Network.Models.PSBackendAddressPool[]>

Specifies a backend address pool to associate with a load balancer.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>

The credentials, account, tenant, and subscription used for communication with azure.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EdgeZone <System.String>

The edge zone of the load balancer

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Force <System.Management.Automation.SwitchParameter>

Indicates that this cmdlet creates a load balancer even if a load balancer with the same name already exists.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-FrontendIpConfiguration <Microsoft.Azure.Commands.Network.Models.PSFrontendIPConfiguration[]>

Specifies a list of front-end IP addresses to associate with a load balancer.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-InboundNatPool <Microsoft.Azure.Commands.Network.Models.PSInboundNatPool[]>

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-InboundNatRule <Microsoft.Azure.Commands.Network.Models.PSInboundNatRule[]>

Specifies a list of inbound network address translation (NAT) rules to associate with a load balancer.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-LoadBalancingRule <Microsoft.Azure.Commands.Network.Models.PSLoadBalancingRule[]>

Specifies a list of load balancing rules to associate with a load balancer.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Location <System.String>

Specifies the region in which to create a load balancer.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Name <System.String>

Specifies the name of the load balancer that this creates.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-OutboundRule <Microsoft.Azure.Commands.Network.Models.PSOutboundRule[]>

The outbound rules.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Probe <Microsoft.Azure.Commands.Network.Models.PSProbe[]>

Specifies a list of probes to associate with a load balancer.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-ResourceGroupName <System.String>

Specifies the name of the resource group in which to create a load balancer.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Sku <System.String>

The load balancer Sku name.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Tag <System.Collections.Hashtable>

Key-value pairs in the form of a hash table. For example: @{key0="value0";key1=$null;key2="value2"}

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Tier <System.String>

The load balancer Sku Tier.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Confirm <System.Management.Automation.SwitchParameter>

Prompts you for confirmation before running the cmdlet.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-WhatIf <System.Management.Automation.SwitchParameter>

Shows what would happen if the cmdlet runs. The cmdlet is not run.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

<CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug,

ErrorAction, ErrorVariable, WarningAction, WarningVariable,

OutBuffer, PipelineVariable, and OutVariable. For more information, see

about_CommonParameters (https:/go.microsoft.com/fwlink/?LinkID=113216).

INPUTS

System.String

System.Collections.Hashtable

Microsoft.Azure.Commands.Network.Models.PSFrontendIPConfiguration[]

Microsoft.Azure.Commands.Network.Models.PSBackendAddressPool[]

Microsoft.Azure.Commands.Network.Models.PSLoadBalancingRule[]

Microsoft.Azure.Commands.Network.Models.PSProbe[]

Microsoft.Azure.Commands.Network.Models.PSInboundNatRule[]

Microsoft.Azure.Commands.Network.Models.PSInboundNatPool[]

Microsoft.Azure.Commands.Network.Models.PSOutboundRule[]

OUTPUTS

Microsoft.Azure.Commands.Network.Models.PSLoadBalancer

NOTES

---------- Example 1: Create a load balancer ----------

$publicip = New-AzPublicIpAddress -ResourceGroupName "MyResourceGroup" -Name "MyPublicIp" -Location "West

US" -AllocationMethod "Dynamic"

$frontend = New-AzLoadBalancerFrontendIpConfig -Name "MyFrontEnd" -PublicIpAddress $publicip

$backendAddressPool = New-AzLoadBalancerBackendAddressPoolConfig -Name "MyBackendAddPoolConfig02"

$probe = New-AzLoadBalancerProbeConfig -Name "MyProbe" -Protocol "http" -Port 80 -IntervalInSeconds 15 -ProbeCount 2 -ProbeThreshold 2 -RequestPath "healthcheck.aspx"

$inboundNatRule1 = New-AzLoadBalancerInboundNatRuleConfig -Name "MyinboundNatRule1" -FrontendIPConfiguration $frontend -Protocol "Tcp" -FrontendPort 3389 -BackendPort

3389 -IdleTimeoutInMinutes 15 -EnableFloatingIP

$inboundNatRule2 = New-AzLoadBalancerInboundNatRuleConfig -Name "MyinboundNatRule2" -FrontendIPConfiguration $frontend -Protocol "Tcp" -FrontendPort 3391 -BackendPort

3392

$lbrule = New-AzLoadBalancerRuleConfig -Name "MyLBruleName" -FrontendIPConfiguration $frontend -BackendAddressPool $backendAddressPool -Probe $probe -Protocol "Tcp"

-FrontendPort 80 -BackendPort 80 -IdleTimeoutInMinutes 15 -EnableFloatingIP -LoadDistribution SourceIP

$lb = New-AzLoadBalancer -Name "MyLoadBalancer" -ResourceGroupName "MyResourceGroup" -Location "West US" -FrontendIpConfiguration $frontend -BackendAddressPool

$backendAddressPool -Probe $probe -InboundNatRule $inboundNatRule1,$inboundNatRule2 -LoadBalancingRule $lbrule

Get-AzLoadBalancer -Name "MyLoadBalancer" -ResourceGroupName "MyResourceGroup"

Deploying a load balancer requires that you first create several objects, and the first seven commands show how to create those objects. The eighth command creates a

load balancer named MyLoadBalancer in the resource group named MyResourceGroup. The ninth and last command gets the new load balancer to ensure it was successfully

created. Note that this example only shows how to create a load balancer. You must also configure it using the Add-AzNetworkInterfaceIpConfig cmdlet to assign the

NICs to different virtual machines.

---------- Example 2: Create a global load balancer ----------

$publicip = New-AzPublicIpAddress -ResourceGroupName "MyResourceGroup" -name "MyPublicIp" -Location "West US" -AllocationMethod Static -DomainNameLabel

$domainNameLabel -Sku Standard -Tier Global

$frontend = New-AzLoadBalancerFrontendIpConfig -Name $frontendName -PublicIpAddress $publicip

$backendAddressPool = New-AzLoadBalancerBackendAddressPoolConfig -Name "MyBackendAddPoolConfig01"

$probe = New-AzLoadBalancerProbeConfig -Name "MyProbe" -RequestPath healthcheck.aspx -Protocol http -Port 80 -IntervalInSeconds 15 -ProbeCount 2 -ProbeThreshold 2

$lbrule = New-AzLoadBalancerRuleConfig -Name "MyLBruleName" -FrontendIPConfiguration $frontend -BackendAddressPool $backendAddressPool -Probe $probe -Protocol Tcp

-FrontendPort 80 -BackendPort 80 -IdleTimeoutInMinutes 15 -EnableFloatingIP -LoadDistribution SourceIP -DisableOutboundSNAT

$lb = New-AzLoadBalancer -Name "MyLoadBalancer" -ResourceGroupName "MyResourceGroup" -Location "West US" -FrontendIpConfiguration $frontend -BackendAddressPool

$backendAddressPool -Probe $probe -LoadBalancingRule $lbrule -Sku Standard -Tier Global

Get-AzLoadBalancer -Name "MyLoadBalancer" -ResourceGroupName "MyResourceGroup"

Deploying a global load balancer requires that you first create several objects, and the first five commands show how to create those objects. The sixth command

creates a load balancer named MyLoadBalancer in the resource group named MyResourceGroup. The seventh and last command gets the new load balancer to ensure it was

successfully created. Note that this example only shows how to create a global load balancer. You must also configure it using the

New-AzLoadBalancerBackendAddressConfig cmdlet to assign regional load balancer frontend ipconfig ids to its backend address pool

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/az.network/new-azloadbalancer

Add-AzNetworkInterfaceIpConfig

Get-AzLoadBalancer

Remove-AzLoadBalancer

Set-AzLoadBalancer