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

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

NAME

New-AzCloudService

SYNOPSIS

Create or update a cloud service.

Please note some properties can be set only during cloud service creation.

SYNTAX

New-AzCloudService -Name <String> -ResourceGroupName <String> [-SubscriptionId <String>] -Location <String> [-AllowModelOverride] [-Configuration <String>]

[-ConfigurationUrl <String>] [-ExtensionProfile <ICloudServiceExtensionProfile>] [-NetworkProfile <ICloudServiceNetworkProfile>] [-OSProfile <ICloudServiceOSProfile>]

[-PackageUrl <String>] [-RoleProfile <ICloudServiceRoleProfile>] [-StartCloudService] [-Tag <Hashtable>] [-UpgradeMode <CloudServiceUpgradeMode>] [-Zone <String[]>]

[-DefaultProfile <PSObject>] [-AsJob] [-Break] [-HttpPipelineAppend <SendAsyncStep[]>] [-HttpPipelinePrepend <SendAsyncStep[]>] [-NoWait] [-Proxy <Uri>]

[-ProxyCredential <PSCredential>] [-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm] [<CommonParameters>]

New-AzCloudService -Name <String> -ResourceGroupName <String> [-SubscriptionId <String>] -Location <String>

[-ExtensionProfile <ICloudServiceExtensionProfile>]

-PackageUrl <String> [-StartCloudService] [-Tag <Hashtable>] [-UpgradeMode <CloudServiceUpgradeMode>] -ConfigurationFile <String> -DefinitionFile <String> [-DnsName

<String>] [-KeyVaultName <String>] [-WhatIf] [-Confirm] [<CommonParameters>]

New-AzCloudService -Name <String> -ResourceGroupName <String> [-SubscriptionId <String>] -Location <String> [-ExtensionProfile <ICloudServiceExtensionProfile>]

[-StartCloudService] [-Tag <Hashtable>] [-UpgradeMode <CloudServiceUpgradeMode>] -ConfigurationFile <String> -DefinitionFile <String> -PackageFile <String>

-StorageAccount <String> [-DnsName <String>] [-KeyVaultName <String>] [-WhatIf] [-Confirm] [<CommonParameters>]

DESCRIPTION

Create or update a cloud service.

Please note some properties can be set only during cloud service creation.

PARAMETERS

-Name <String>

Name of the cloud service.

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-ResourceGroupName <String>

Name of the resource group.

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-SubscriptionId <String>

Subscription credentials which uniquely identify Microsoft Azure subscription.

The subscription ID forms part of the URI for every service call.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-Location <String>

Resource location.

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-AllowModelOverride [<SwitchParameter>]

(Optional) Indicates whether the role sku properties (roleProfile.roles.sku) specified in the model/template should override the role instance count and vm size

specified in the .cscfg and .csdef respectively.The default value is `false`.

Required? false

Position? named

Default value False

Accept pipeline input? false

Accept wildcard characters? false

-Configuration <String>

Specifies the XML service configuration (.cscfg) for the cloud service.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-ConfigurationUrl <String>

Specifies a URL that refers to the location of the service configuration in the Blob service.

The service package URL can be Shared Access Signature (SAS) URI from any storage account.This is a write-only property and is not returned in GET calls.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-ExtensionProfile <ICloudServiceExtensionProfile>

Describes a cloud service extension profile.

To construct, see NOTES section for EXTENSIONPROFILE properties and create a hash table.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-NetworkProfile <ICloudServiceNetworkProfile>

Network Profile for the cloud service.

To construct, see NOTES section for NETWORKPROFILE properties and create a hash table.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-OSProfile <ICloudServiceOSProfile>

Describes the OS profile for the cloud service.

To construct, see NOTES section for OSPROFILE properties and create a hash table.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-PackageUrl <String>

Specifies a URL that refers to the location of the service package in the Blob service.

The service package URL can be Shared Access Signature (SAS) URI from any storage account.This is a write-only property and is not returned in GET calls.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-RoleProfile <ICloudServiceRoleProfile>

Describes the role profile for the cloud service.

To construct, see NOTES section for ROLEPROFILE properties and create a hash table.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-StartCloudService [<SwitchParameter>]

(Optional) Indicates whether to start the cloud service immediately after it is created.

The default value is `true`.If false, the service model is still deployed, but the code is not run immediately.

Instead, the service is PoweredOff until you call Start, at which time the service will be started.

A deployed service still incurs charges, even if it is poweredoff.

Required? false

Position? named

Default value False

Accept pipeline input? false

Accept wildcard characters? false

-Tag <Hashtable>

Resource tags.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-UpgradeMode <CloudServiceUpgradeMode>

Update mode for the cloud service.

Role instances are allocated to update domains when the service is deployed.

Updates can be initiated manually in each update domain or initiated automatically in all update domains.Possible Values are

Auto

Manual

Simultaneous

If not specified, the default value is Auto.

If set to Manual, PUT UpdateDomain must be called to apply the update.

If set to Auto, the update is automatically applied to each update domain in sequence.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-Zone <String[]>

List of logical availability zone of the resource.

List should contain only 1 zone where cloud service should be provisioned.

This field is optional.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-ConfigurationFile <String>

Specifies the XML service configuration (.cscfg) for the cloud service.

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-DefinitionFile <String>

Path to .csdef file.

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-PackageFile <String>

Path to .cspkg file.

It will be uploaded to a blob

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-StorageAccount <String>

Name of the storage account that will store the Package file.

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-DnsName <String>

Name of Dns to be used for the CloudService resource.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-KeyVaultName <String>

Name of the KeyVault to be used for the CloudService resource.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-DefaultProfile <PSObject>

The DefaultProfile parameter is not functional.

Use the SubscriptionId parameter when available if executing the cmdlet against a different subscription.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-AsJob [<SwitchParameter>]

Run the command as a job

Required? false

Position? named

Default value False

Accept pipeline input? false

Accept wildcard characters? false

-Break [<SwitchParameter>]

Wait for .NET debugger to attach

Required? false

Position? named

Default value False

Accept pipeline input? false

Accept wildcard characters? false

-HttpPipelineAppend <SendAsyncStep[]>

SendAsync Pipeline Steps to be appended to the front of the pipeline

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-HttpPipelinePrepend <SendAsyncStep[]>

SendAsync Pipeline Steps to be prepended to the front of the pipeline

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-NoWait [<SwitchParameter>]

Run the command asynchronously

Required? false

Position? named

Default value False

Accept pipeline input? false

Accept wildcard characters? false

-Proxy <Uri>

The URI for the proxy server to use

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-ProxyCredential <PSCredential>

Credentials for a proxy server to use for the remote call

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-ProxyUseDefaultCredentials [<SwitchParameter>]

Use the default credentials for the proxy

Required? false

Position? named

Default value False

Accept pipeline input? false

Accept wildcard characters? false

-WhatIf [<SwitchParameter>]

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-Confirm [<SwitchParameter>]

Required? false

Position? named

Default value

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

OUTPUTS

Microsoft.Azure.PowerShell.Cmdlets.CloudService.Models.Api20220904.ICloudService

NOTES

COMPLEX PARAMETER PROPERTIES

To create the parameters described below, construct a hash table containing the appropriate properties. For information on hash tables, run Get-Help

about_Hash_Tables.

EXTENSIONPROFILE <ICloudServiceExtensionProfile>: Describes a cloud service extension profile.

[Extension <IExtension[]>]: List of extensions for the cloud service.

[AutoUpgradeMinorVersion <Boolean?>]: Explicitly specify whether platform can automatically upgrade typeHandlerVersion to higher minor versions when they

become available.

[ForceUpdateTag <String>]: Tag to force apply the provided public and protected settings. Changing the tag value allows for re-running the extension

without changing any of the public or protected settings. If forceUpdateTag is not changed, updates to public or protected settings would still be applied

by the handler. If neither forceUpdateTag nor any of public or protected settings change, extension would flow to the role instance with the same

sequence-number, and it is up to handler implementation whether to re-run it or not

[Name <String>]: The name of the extension.

[ProtectedSetting <String>]: Protected settings for the extension which are encrypted before sent to the role instance.

[ProtectedSettingFromKeyVaultSecretUrl <String>]: Secret URL which contains the protected settings of the extension

[Publisher <String>]: The name of the extension handler publisher.

[RolesAppliedTo <String[]>]: Optional list of roles to apply this extension. If property is not specified or '*' is specified, extension is applied to all

roles in the cloud service.

[Setting <String>]: Public settings for the extension. For JSON extensions, this is the JSON settings for the extension. For XML Extension (like RDP), this is

the XML setting for the extension.

[SourceVaultId <String>]: Resource Id

[Type <String>]: Specifies the type of the extension.

[TypeHandlerVersion <String>]: Specifies the version of the extension. Specifies the version of the extension. If this element is not specified or an asterisk

(*) is used as the value, the latest version of the extension is used. If the value is specified with a major version number and an asterisk as the minor version

number (X.), the latest minor version of the specified major version is selected. If a major version number and a minor version number are specified (X.Y), the

specific extension version is selected. If a version is specified, an auto-upgrade is performed on the role instance.

NETWORKPROFILE <ICloudServiceNetworkProfile>: Network Profile for the cloud service.

[LoadBalancerConfiguration <ILoadBalancerConfiguration[]>]: List of Load balancer configurations. Cloud service can have up to two load balancer configurations,

corresponding to a Public Load Balancer and an Internal Load Balancer.

FrontendIPConfiguration <ILoadBalancerFrontendIPConfiguration[]>: Specifies the frontend IP to be used for the load balancer. Only IPv4 frontend IP address is

supported. Each load balancer configuration must have exactly one frontend IP configuration.

Name <String>: The name of the resource that is unique within the set of frontend IP configurations used by the load balancer. This name can be used to

access the resource.

[PrivateIPAddress <String>]: The virtual network private IP address of the IP configuration.

[PublicIPAddressId <String>]: Resource Id

[SubnetId <String>]: Resource Id

Name <String>: The name of the Load balancer

[Id <String>]: Resource Id

[SlotType <CloudServiceSlotType?>]: Slot type for the cloud service. Possible values are

Production

Staging

If not specified, the default value is Production.

[SwappableCloudService <ISubResource>]: The id reference of the cloud service containing the target IP with which the subject cloud service can perform a swap.

This property cannot be updated once it is set. The swappable cloud service referred by this id must be present otherwise an error will be thrown.

[Id <String>]: Resource Id

OSPROFILE <ICloudServiceOSProfile>: Describes the OS profile for the cloud service.

[Secret <ICloudServiceVaultSecretGroup[]>]: Specifies set of certificates that should be installed onto the role instances.

[SourceVaultId <String>]: Resource Id

[VaultCertificate <ICloudServiceVaultCertificate[]>]: The list of key vault references in SourceVault which contain certificates.

[CertificateUrl <String>]: This is the URL of a certificate that has been uploaded to Key Vault as a secret.

ROLEPROFILE <ICloudServiceRoleProfile>: Describes the role profile for the cloud service.

[Role <ICloudServiceRoleProfileProperties[]>]: List of roles for the cloud service.

[Name <String>]: Resource name.

[SkuCapacity <Int64?>]: Specifies the number of role instances in the cloud service.

[SkuName <String>]: The sku name. NOTE: If the new SKU is not supported on the hardware the cloudservice is

currently on, you need to delete and recreate the

cloud service or move back to the old sku.

[SkuTier <String>]: Specifies the tier of the cloud service. Possible Values are

 Standard

 Basic

---------- EXAMPLE 1 ----------

PS C:\># Create role profile object

$role = New-AzCloudServiceRoleProfilePropertiesObject -Name 'ContosoFrontend' -SkuName 'Standard_D1_v2' -SkuTier 'Standard' -SkuCapacity 2

$roleProfile = @{role = @($role)}

Create network profile object

$publicIp = Get-AzPublicIpAddress -ResourceGroupName ContosOrg -Name ContosIp

$feIpConfig = New-AzCloudServiceLoadBalancerFrontendIPConfigurationObject -Name 'ContosoFe' -PublicIPAddressId $publicIp.Id

$loadBalancerConfig = New-AzCloudServiceLoadBalancerConfigurationObject -Name 'ContosoLB' -FrontendIPConfiguration $feIpConfig

$networkProfile = @{loadBalancerConfiguration = $loadBalancerConfig}

Read Configuration File

$cscfgFile = "<Path to cscfg configuration file>"

$cscfgContent = Get-Content $cscfgFile | Out-String

Create cloud service

$cloudService = New-AzCloudService `

-Name ContosoCS `

-ResourceGroupName ContosOrg `

-Location EastUS `

-PackageUrl "https://xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx" `

-Configuration $cscfgContent `

-UpgradeMode 'Auto' `

```
                -RoleProfile $roleProfile                                    `
                -NetworkProfile $networkProfile
```

---------- EXAMPLE 2 ----------

```
PS C:\># Create role profile object

    $role = New-AzCloudServiceRoleProfilePropertiesObject -Name 'ContosoFrontend' -SkuName 'Standard_D1_v2'
-SkuTier 'Standard' -SkuCapacity 2
$roleProfile = @{role = @($role)}

# Create network profile object
$publicIp = Get-AzPublicIpAddress -ResourceGroupName ContosoOrg -Name ContosIp
 $feIpConfig = New-AzCloudServiceLoadBalancerFrontendIPConfigurationObject -Name 'ContosoFe' -PublicIPAddressId
$publicIp.Id
            $loadBalancerConfig = New-AzCloudServiceLoadBalancerConfigurationObject -Name 'ContosoLB'
-FrontendIPConfiguration $feIpConfig
$networkProfile = @{loadBalancerConfiguration = $loadBalancerConfig}

# Create RDP extension object
$credential = Get-Credential
$expiration = (Get-Date).AddYears(1)
    $extension = New-AzCloudServiceRemoteDesktopExtensionObject -Name 'RDPExtension' -Credential $credential
-Expiration $expiration -TypeHandlerVersion '1.2.1'
$extensionProfile = @{extension = @($extension)}

# Read Configuration File
$cscfgFile = "<Path to cscfg configuration file>"
$cscfgContent = Get-Content $cscfgFile | Out-String
```

```
    # Create cloud service
    $cloudService = New-AzCloudService                                  `
                        -Name ContosoCS                                 `
                        -ResourceGroupName ContosOrg                    `
                        -Location EastUS                                `
                        -PackageUrl "https://xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"    `
                        -Configuration $cscfgContent                    `
                        -UpgradeMode 'Auto'                             `
                        -RoleProfile $roleProfile                       `
                        -NetworkProfile $networkProfile                 `
                        -ExtensionProfile $extensionProfile




    ------------------------- EXAMPLE 3 -------------------------


    PS C:\># Create role profile object


        $role = New-AzCloudServiceRoleProfilePropertiesObject -Name 'ContosoFrontend' -SkuName 'Standard_D1_v2'
-SkuTier 'Standard' -SkuCapacity 2
    $roleProfile = @{role = @($role)}


    # Create OS profile object
    $keyVault = Get-AzKeyVault -ResourceGroupName ContosOrg -VaultName ContosKeyVault
    $certificate=Get-AzKeyVaultCertificate -VaultName ContosKeyVault -Name ContosCert
            $secretGroup = New-AzCloudServiceVaultSecretGroupObject -Id $keyVault.ResourceId -CertificateUrl
$certificate.SecretId
    $osProfile = @{secret = @($secretGroup)}


    # Create network profile object
    $publicIp = Get-AzPublicIpAddress -ResourceGroupName ContosOrg -Name ContosIp
     $feIpConfig = New-AzCloudServiceLoadBalancerFrontendIPConfigurationObject -Name 'ContosoFe' -PublicIPAddressId
```

```
$publicIp.Id

        $loadBalancerConfig = New-AzCloudServiceLoadBalancerConfigurationObject -Name 'ContosoLB'
-FrontendIPConfiguration $feIpConfig
    $networkProfile = @{loadBalancerConfiguration = $loadBalancerConfig}

    # Read Configuration File
    $cscfgFile = "<Path to cscfg configuration file>"
    $cscfgContent = Get-Content $cscfgFile | Out-String

    # Create cloud service
    $cloudService = New-AzCloudService                                  `
                        -Name ContosoCS                                 `
                        -ResourceGroupName ContosOrg                    `
                        -Location EastUS                                `
                        -PackageUrl "https://xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"    `
                        -Configuration $cscfgContent                    `
                        -UpgradeMode 'Auto'                             `
                        -RoleProfile $roleProfile                       `
                        -NetworkProfile $networkProfile                 `
                        -OSProfile $osProfile




    ---------------------------- EXAMPLE 4 ----------------------------

    PS C:\># Create role profile object

        $role1 = New-AzCloudServiceRoleProfilePropertiesObject -Name 'ContosoFrontend' -SkuName 'Standard_D1_v2'
-SkuTier 'Standard' -SkuCapacity 2
        $role2 = New-AzCloudServiceRoleProfilePropertiesObject -Name 'ContosoBackend' -SkuName 'Standard_D1_v2'
-SkuTier 'Standard' -SkuCapacity 2
    $roleProfile = @{role = @($role1, $role2)}
```

```
    # Create network profile object
    $publicIp = Get-AzPublicIpAddress -ResourceGroupName ContosOrg -Name ContosIp
     $feIpConfig = New-AzCloudServiceLoadBalancerFrontendIPConfigurationObject -Name 'ContosoFe' -PublicIPAddressId
$publicIp.Id
                $loadBalancerConfig = New-AzCloudServiceLoadBalancerConfigurationObject -Name 'ContosoLB'
-FrontendIPConfiguration $feIpConfig
    $networkProfile = @{loadBalancerConfiguration = $loadBalancerConfig}

    # Create RDP extension object
    $credential = Get-Credential
    $expiration = (Get-Date).AddYears(1)
      $rdpExtension = New-AzCloudServiceRemoteDesktopExtensionObject -Name 'RDPExtension' -Credential $credential
-Expiration $expiration -TypeHandlerVersion '1.2.1'

    # Create Geneva extension object
     $genevaExtension = New-AzCloudServiceExtensionObject -Name GenevaExtension -Publisher Microsoft.Azure.Geneva
-Type GenevaMonitoringPaaS -TypeHandlerVersion "2.14.0.2"
    $extensionProfile = @{extension = @($rdpExtension, $genevaExtension)}

    # Add tags
    $tag=@{"Owner" = "Contoso"}

    # Read Configuration File
    $cscfgFile = "<Path to cscfg configuration file>"
    $cscfgContent = Get-Content $cscfgFile | Out-String

    # Create cloud service
    $cloudService = New-AzCloudService                                  `
                        -Name ContosoCS                                 `
                        -ResourceGroupName ContosOrg                    `
                        -Location EastUS                                `
                        -PackageUrl "https://xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"    `
```

```
                -Configuration $cscfgContent                    `
                -UpgradeMode 'Auto'                             `
                -RoleProfile $roleProfile                       `
                -NetworkProfile $networkProfile                 `
                -ExtensionProfile $extensionProfile             `
                -Tag $tag
```

---------- EXAMPLE 5 ----------

```
PS C:\># Set up a storage account if you have not

 $storageAccount = New-AzStorageAccount -ResourceGroupName ContosoOrg -Name ContosoStorAcc -Location "East US" -SkuName "Standard_RAGRS" -Kind "StorageV2"

# Create cloud service
$cloudService = New-AzCloudService                              `
                -Name ContosoCS                                 `
                -ResourceGroupName ContosOrg                    `
                -Location EastUS                                `
                -ConfigurationFile C:\files\CS.cscfg            `
                -DefinitionFile C:\files\CS.csdef               `
                -PackageFile C:\CS.cspkg                        `
                -StorageAccount ContosoStorAcc                  `
                -KeyVaultName ContosoKV
```

---------- EXAMPLE 6 ----------

```
    PS C:\># getting Package URL

    $tokenStartTime = Get-Date
    $tokenEndTime = $tokenStartTime.AddYears(1)
    $storAcc = Get-AzStorageAccount -ResourceGroupName ContosoOrg -Name ContosoStorAcc
    $csPkgBlob = Get-AzStorageBlob -Container Contoso-Container -Blob ContosoBlob.cspkg -Context $storAcc.Context
    $csPkgToken = New-AzStorageBlobSASToken -Container Contoso-Container -Blob ContosoBlob.cspkg -Permission rwd
-StartTime $tokenStartTime -ExpiryTime $tokenEndTime
    -Context $storAcc.Context
    $cspkgUrl = $csPkgBlob.ICloudBlob.Uri.AbsoluteUri + $csPkgToken

    # Create cloud service
    $cloudService = New-AzCloudService                    `
                        -Name ContosoCS                   `
                        -ResourceGroupName ContosOrg      `
                        -Location EastUS                  `
                        -ConfigurationFile C:\files\CS.cscfg  `
                        -DefinitionFile C:\files\CS.csdef     `
                        -packageUrl $cspkgUrl             `
```

RELATED LINKS

https://learn.microsoft.com/powershell/module/az.cloudservice/new-azcloudservice