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

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

NAME

New-AzStorageAccount

SYNOPSIS

Creates a Storage account.

SYNTAX

New-AzStorageAccount [-ResourceGroupName] <System.String> [-Name] <System.String> [-SkuName] {Standard_LRS | Standard_ZRS | Standard_GRS | Standard_RAGRS |

Premium_LRS | Premium_ZRS | Standard_GZRS | Standard_RAGZRS} [-Location] <System.String> [-AccessTier {Hot | Cool}] [-ActiveDirectoryAccountType <System.String>]

[-ActiveDirectoryAzureStorageSid <System.String>] [-ActiveDirectoryDomainGuid <System.String>] [-ActiveDirectoryDomainName <System.String>] [-ActiveDirectoryDomainSid

<System.String>] [-ActiveDirectoryForestName <System.String>] [-ActiveDirectoryNetBiosDomainName <System.String>] [-ActiveDirectorySamAccountName <System.String>]

[-AllowBlobPublicAccess <System.Boolean>] [-AllowCrossTenantReplication <System.Boolean>] [-AllowedCopyScope <System.String>] [-AllowSharedKeyAccess <System.Boolean>]

[-AsJob] [-AssignIdentity] [-CustomDomainName <System.String>] [-DefaultProfile

<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>]

[-DefaultSharePermission {None | StorageFileDataSmbShareContributor |
StorageFileDataSmbShareReader | StorageFileDataSmbShareElevatedContributor}] [-DnsEndpointType <System.String>] [-EdgeZone <System.String>]
[-EnableAccountLevelImmutability] [-EnableActiveDirectoryDomainServicesForFile <System.Boolean>] [-EnableHierarchicalNamespace <System.Boolean>]
[-EnableHttpsTrafficOnly <System.Boolean>] [-EnableLargeFileShare] [-EnableLocalUser <System.Boolean>] [-EnableNfsV3 <System.Boolean>] [-EnableSftp <System.Boolean>]
[-EncryptionKeyTypeForQueue {Service | Account}] [-EncryptionKeyTypeForTable {Service | Account}] [-IdentityType {SystemAssigned | UserAssigned |
SystemAssignedUserAssigned | None}] [-ImmutabilityPeriod <System.Int32>] [-ImmutabilityPolicyState <System.String>] [-KeyExpirationPeriodInDay <System.Int32>]
[-KeyName <System.String>] [-KeyVaultFederatedClientId <System.String>] [-KeyVaultUri <System.String>] [-KeyVaultUserAssignedIdentityId <System.String>] [-KeyVersion
<System.String>] [-Kind {Storage | StorageV2 | BlobStorage | BlockBlobStorage | FileStorage}] [-MinimumTlsVersion {TLS1_0 | TLS1_1 | TLS1_2}] [-NetworkRuleSet
<Microsoft.Azure.Commands.Management.Storage.Models.PSNetworkRuleSet>] [-PublicNetworkAccess <System.String>] [-PublishInternetEndpoint <System.Boolean>]
[-PublishMicrosoftEndpoint <System.Boolean>] [-RequireInfrastructureEncryption] [-RoutingChoice {MicrosoftRouting | InternetRouting}] [-SasExpirationPeriod
<System.TimeSpan>] [-Tag <System.Collections.Hashtable>] [-UserAssignedIdentityId <System.String>] [-UseSubDomain <System.Nullable`1[System.Boolean]>]
[<CommonParameters>]

New-AzStorageAccount [-ResourceGroupName] <System.String> [-Name] <System.String> [-SkuName] {Standard_LRS | Standard_ZRS | Standard_GRS | Standard_RAGRS |
Premium_LRS | Premium_ZRS | Standard_GZRS | Standard_RAGZRS} [-Location] <System.String> [-AccessTier {Hot | Cool}] [-ActiveDirectoryDomainGuid <System.String>]
[-ActiveDirectoryDomainName <System.String>] [-AllowBlobPublicAccess <System.Boolean>] [-AllowCrossTenantReplication <System.Boolean>] [-AllowedCopyScope
<System.String>] [-AllowSharedKeyAccess <System.Boolean>] [-AsJob] [-AssignIdentity] [-CustomDomainName <System.String>] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>]
[-DefaultSharePermission {None | StorageFileDataSmbShareContributor |

StorageFileDataSmbShareReader | StorageFileDataSmbShareElevatedContributor}] [-DnsEndpointType <System.String>] [-EdgeZone <System.String>]

[-EnableAccountLevelImmutability] -EnableAzureActiveDirectoryKerberosForFile <System.Boolean> [-EnableHierarchicalNamespace <System.Boolean>] [-EnableHttpsTrafficOnly

<System.Boolean>] [-EnableLargeFileShare] [-EnableLocalUser <System.Boolean>] [-EnableNfsV3 <System.Boolean>] [-EnableSftp <System.Boolean>]

[-EncryptionKeyTypeForQueue {Service | Account}] [-EncryptionKeyTypeForTable {Service | Account}] [-IdentityType {SystemAssigned | UserAssigned |

SystemAssignedUserAssigned | None}] [-ImmutabilityPeriod <System.Int32>] [-ImmutabilityPolicyState <System.String>] [-KeyExpirationPeriodInDay <System.Int32>]

[-KeyName <System.String>] [-KeyVaultFederatedClientId <System.String>] [-KeyVaultUri <System.String>] [-KeyVaultUserAssignedIdentityId <System.String>] [-KeyVersion

<System.String>] [-Kind {Storage | StorageV2 | BlobStorage | BlockBlobStorage | FileStorage}] [-MinimumTlsVersion {TLS1_0 | TLS1_1 | TLS1_2}] [-NetworkRuleSet

<Microsoft.Azure.Commands.Management.Storage.Models.PSNetworkRuleSet>] [-PublicNetworkAccess <System.String>] [-PublishInternetEndpoint <System.Boolean>]

[-PublishMicrosoftEndpoint <System.Boolean>] [-RequireInfrastructureEncryption] [-RoutingChoice {MicrosoftRouting | InternetRouting}] [-SasExpirationPeriod

<System.TimeSpan>] [-Tag <System.Collections.Hashtable>] [-UserAssignedIdentityId <System.String>] [-UseSubDomain <System.Nullable`1[System.Boolean]>]

[<CommonParameters>]

New-AzStorageAccount [-ResourceGroupName] <System.String> [-Name] <System.String> [-SkuName] {Standard_LRS | Standard_ZRS | Standard_GRS | Standard_RAGRS |

Premium_LRS | Premium_ZRS | Standard_GZRS | Standard_RAGZRS} [-Location] <System.String> [-AccessTier {Hot | Cool}] [-AllowBlobPublicAccess <System.Boolean>]

[-AllowCrossTenantReplication <System.Boolean>] [-AllowedCopyScope <System.String>] [-AllowSharedKeyAccess <System.Boolean>] [-AsJob] [-AssignIdentity]

[-CustomDomainName <System.String>] [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>]

[-DefaultSharePermission {None | StorageFileDataSmbShareContributor | StorageFileDataSmbShareReader | StorageFileDataSmbShareElevatedContributor}] [-DnsEndpointType

<System.String>] [-EdgeZone <System.String>] [-EnableAccountLevelImmutability]

[-EnableAzureActiveDirectoryDomainServicesForFile <System.Boolean>]

[-EnableHierarchicalNamespace <System.Boolean>] [-EnableHttpsTrafficOnly <System.Boolean>] [-EnableLargeFileShare] [-EnableLocalUser <System.Boolean>] [-EnableNfsV3

<System.Boolean>] [-EnableSftp <System.Boolean>] [-EncryptionKeyTypeForQueue {Service | Account}] [-EncryptionKeyTypeForTable {Service | Account}] [-IdentityType

{SystemAssigned | UserAssigned | SystemAssignedUserAssigned | None}] [-ImmutabilityPeriod <System.Int32>] [-ImmutabilityPolicyState <System.String>]

[-KeyExpirationPeriodInDay <System.Int32>] [-KeyName <System.String>] [-KeyVaultFederatedClientId <System.String>] [-KeyVaultUri <System.String>]

[-KeyVaultUserAssignedIdentityId <System.String>] [-KeyVersion <System.String>] [-Kind {Storage | StorageV2 | BlobStorage | BlockBlobStorage | FileStorage}]

[-MinimumTlsVersion {TLS1_0 | TLS1_1 | TLS1_2}] [-NetworkRuleSet <Microsoft.Azure.Commands.Management.Storage.Models.PSNetworkRuleSet>] [-PublicNetworkAccess

<System.String>] [-PublishInternetEndpoint <System.Boolean>] [-PublishMicrosoftEndpoint <System.Boolean>] [-RequireInfrastructureEncryption] [-RoutingChoice

{MicrosoftRouting | InternetRouting}] [-SasExpirationPeriod <System.TimeSpan>] [-Tag <System.Collections.Hashtable>] [-UserAssignedIdentityId <System.String>]

[-UseSubDomain <System.Nullable`1[System.Boolean]>] [<CommonParameters>]

DESCRIPTION

The New-AzStorageAccount cmdlet creates an Azure Storage account.

PARAMETERS

-AccessTier <System.String>

Specifies the access tier of the Storage account that this cmdlet creates. The acceptable values for this parameter are: Hot and Cool. If you specify a value of

BlobStorage for the Kind parameter, you must specify a value for the AccessTier parameter. If you specify a value of Storage for this Kind parameter, do not

specify the AccessTier parameter.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ActiveDirectoryAccountType <System.String>

Specifies the Active Directory account type for Azure Storage. Possible values include: 'User', 'Computer'.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ActiveDirectoryAzureStorageSid <System.String>

Specifies the security identifier (SID) for Azure Storage. This parameter must be set when -EnableActiveDirectoryDomainServicesForFile is set to true.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ActiveDirectoryDomainGuid <System.String>

Specifies the domain GUID. This parameter must be set when -EnableActiveDirectoryDomainServicesForFile is set to true.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ActiveDirectoryDomainName <System.String>

Specifies the primary domain that the AD DNS server is authoritative for. This parameter must be set when -EnableActiveDirectoryDomainServicesForFile is set to

true.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ActiveDirectoryDomainSid <System.String>

Specifies the security identifier (SID). This parameter must be set when -EnableActiveDirectoryDomainServicesForFile is set to true.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ActiveDirectoryForestName <System.String>

Specifies the Active Directory forest to get. This parameter must be set when -EnableActiveDirectoryDomainServicesForFile is set to true.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ActiveDirectoryNetBiosDomainName <System.String>

Specifies the NetBIOS domain name. This parameter must be set when -EnableActiveDirectoryDomainServicesForFile is set to true.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ActiveDirectorySamAccountName <System.String>

Specifies the Active Directory SAMAccountName for Azure Storage.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-AllowBlobPublicAccess <System.Boolean>

Allow anonymous access to all blobs or containers in the storage account. The default interpretation is false for this property.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-AllowCrossTenantReplication <System.Boolean>

Gets or sets allow or disallow cross Microsoft Entra tenant object replication. The default interpretation is false for this property.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-AllowedCopyScope <System.String>

Set restrict copy to and from Storage Accounts within a Microsoft Entra tenant or with Private Links to the same VNet. Possible values include: 'PrivateLink',

'AAD'

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-AllowSharedKeyAccess <System.Boolean>

Indicates whether the storage account permits requests to be authorized with the account access key via Shared Key. If false, then all requests, including shared

access signatures, must be authorized with Microsoft Entra ID. The default value is null, which is equivalent to true.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-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

-AssignIdentity <System.Management.Automation.SwitchParameter>

Generate and assign a new Storage account Identity for this Storage account for use with key management services like Azure KeyVault.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-CustomDomainName <System.String>

Specifies the name of the custom domain of the Storage account. The default value is Storage.

Required? false

Position? named

Default value None

Accept pipeline input? False

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

-DefaultSharePermission <System.String>

Default share permission for users using Kerberos authentication if RBAC role is not assigned.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-DnsEndpointType <System.String>

Specify the type of endpoint. Set this to AzureDNSZone to create a large number of accounts in a single subscription, which creates accounts in an Azure DNS Zone

and the endpoint URL will have an alphanumeric DNS Zone identifier. Possible values include: 'Standard', 'AzureDnsZone'.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EdgeZone <System.String>

Set the extended location name for EdgeZone. If not set, the storage account will be created in Azure main region. Otherwise it will be created in the specified

extended location

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EnableAccountLevelImmutability <System.Management.Automation.SwitchParameter>

Enables account-level immutability, then all the containers under this account will have object-level immutability enabled by default.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-EnableActiveDirectoryDomainServicesForFile <System.Boolean>

Enable Azure Files Active Directory Domain Service Authentication for the storage account.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EnableAzureActiveDirectoryDomainServicesForFile <System.Boolean>

Enable Azure Files Microsoft Entra Domain Service Authentication for the storage account.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EnableAzureActiveDirectoryKerberosForFile <System.Boolean>

Enable Azure Files Active Directory Domain Service Kerberos Authentication for the storage account.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EnableHierarchicalNamespace <System.Boolean>

Indicates whether or not the Storage account enables Hierarchical Namespace.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EnableHttpsTrafficOnly <System.Boolean>

Indicates whether or not the Storage account only enables HTTPS traffic.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EnableLargeFileShare <System.Management.Automation.SwitchParameter>

Indicates whether or not the storage account can support large file shares with more than 5 TiB capacity. Once the account is enabled, the feature cannot be

disabled. Currently only supported for LRS and ZRS replication types, hence account conversions to geo-redundant accounts would not be possible. Learn more in

https://go.microsoft.com/fwlink/?linkid=2086047

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-EnableLocalUser <System.Boolean>

Enable local users feature for the Storage account.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EnableNfsV3 <System.Boolean>

Enable NFS 3.0 protocol support if sets to true

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EnableSftp <System.Boolean>

Enable Secure File Transfer Protocol for the Storage account.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EncryptionKeyTypeForQueue <System.String>

Set the Encryption KeyType for Queue. The default value is Service. -Account: Queue will be encrypted with account-scoped encryption key. -Service: Queue will

always be encrypted with Service-Managed keys.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EncryptionKeyTypeForTable <System.String>

Set the Encryption KeyType for Table. The default value is Service. - Account: Table will be encrypted with account-scoped encryption key.

- Service: Table will always be encrypted with Service-Managed keys.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-IdentityType <System.String>

Set the new Storage Account Identity type, the idenetity is for use with key management services like Azure KeyVault.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ImmutabilityPeriod <System.Int32>

The immutability period for the blobs in the container since the policy creation in days. This property can only be only be specified with

'-EnableAccountLevelImmutability'.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ImmutabilityPolicyState <System.String>

The mode of the policy. Possible values include: 'Unlocked', 'Disabled. Disabled state disablesthe policy. Unlocked state allows increase and decrease of

immutability retention time and also allows toggling allowProtectedAppendWrites property. A policy can only be created in a Disabled or Unlocked state and can be

toggled between the two states. This property can only be specified with '-EnableAccountLevelImmutability'.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-KeyExpirationPeriodInDay <System.Int32>

The Key expiration period of this account, it is accurate to days.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-KeyName <System.String>

Storage Account encryption keySource KeyVault KeyName

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-KeyVaultFederatedClientId <System.String>

Set ClientId of the multi-tenant application to be used in conjunction with the user-assigned identity for cross-tenant customer-managed-keys server-side

encryption on the storage account.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-KeyVaultUri <System.String>

Storage Account encryption keySource KeyVault KeyVaultUri

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-KeyVaultUserAssignedIdentityId <System.String>

Set resource id for user assigned Identity used to access Azure KeyVault of Storage Account Encryption, the id must in UserAssignIdentityId.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-KeyVersion <System.String>

Storage Account encryption keySource KeyVault KeyVersion

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Kind <System.String>

Specifies the kind of Storage account that this cmdlet creates. The acceptable values for this parameter are: - Storage. General purpose Storage account that

supports storage of Blobs, Tables, Queues, Files and Disks.

- StorageV2. General Purpose Version 2 (GPv2) Storage account that supports Blobs, Tables, Queues, Files, and Disks, with advanced features like data tiering.

- BlobStorage. Blob Storage account which supports storage of Blobs only.

- BlockBlobStorage. Block Blob Storage account which supports storage of Block Blobs only.

- FileStorage. File Storage account which supports storage of Files only.

The default value is StorageV2.

Required? false

Position? named

Default value StorageV2

Accept pipeline input? False

Accept wildcard characters? false

-Location <System.String>

Specifies the location of the Storage account to create.

Required? true

Position? 3

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-MinimumTlsVersion <System.String>

The minimum TLS version to be permitted on requests to storage. The default interpretation is TLS 1.0 for this property.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Name <System.String>

Specifies the name of the Storage account to create.

Required? true

Position? 1

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-NetworkRuleSet <Microsoft.Azure.Commands.Management.Storage.Models.PSNetworkRuleSet>

NetworkRuleSet is used to define a set of configuration rules for firewalls and virtual networks, as well as to set values for network properties such as services

allowed to bypass the rules and how to handle requests that don't match any of the defined rules.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PublicNetworkAccess <System.String>

Allow or disallow public network access to Storage Account.Possible values include: 'Enabled', 'Disabled'.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PublishInternetEndpoint <System.Boolean>

Indicates whether internet routing storage endpoints are to be published

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PublishMicrosoftEndpoint <System.Boolean>

Indicates whether microsoft routing storage endpoints are to be published

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-RequireInfrastructureEncryption <System.Management.Automation.SwitchParameter>

The service will apply a secondary layer of encryption with platform managed keys for data at rest.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-ResourceGroupName <System.String>

Specifies the name of the resource group in which to add the Storage account.

Required? true

Position? 0

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-RoutingChoice <System.String>

Routing Choice defines the kind of network routing opted by the user. Possible values include: 'MicrosoftRouting', 'InternetRouting'

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-SasExpirationPeriod <System.TimeSpan>

The SAS expiration period of this account, it is a timespan and accurate to seconds.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-SkuName <System.String>

Specifies the SKU name of the Storage account that this cmdlet creates. The acceptable values for this parameter are:

- Standard_LRS. Locally-redundant storage.

- Standard_ZRS. Zone-redundant storage.

- Standard_GRS. Geo-redundant storage.

- Standard_RAGRS. Read access geo-redundant storage.

- Premium_LRS. Premium locally-redundant storage.

- Premium_ZRS. Premium zone-redundant storage.

- Standard_GZRS - Geo-redundant zone-redundant storage.

- Standard_RAGZRS - Read access geo-redundant zone-redundant storage.

Required? true

Position? 2

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 set as tags on the server. For example: @{key0="value0";key1=$null;key2="value2"}

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-UserAssignedIdentityId <System.String>

Set resource ids for the new Storage Account user assigned Identity, the identity will be used with key management services like Azure KeyVault.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-UseSubDomain <System.Nullable`1[System.Boolean]>

Indicates whether to enable indirect CName validation.

Required? false

Position? named

Default value None

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

OUTPUTS

Microsoft.Azure.Commands.Management.Storage.Models.PSStorageAccount

NOTES

---------- Example 1: Create a Storage account ----------

New-AzStorageAccount -ResourceGroupName MyResourceGroup -Name mystorageaccount -Location westus -SkuName Standard_GRS

This command creates a Storage account for the resource group name MyResourceGroup.

Example 2: Create a Blob Storage account with BlobStorage Kind and hot AccessTier

New-AzStorageAccount -ResourceGroupName MyResourceGroup -Name mystorageaccount -Location westus -SkuName Standard_GRS -Kind BlobStorage -AccessTier Hot

This command creates a Blob Storage account that with BlobStorage Kind and hot AccessTier

Example 3: Create a Storage account with Kind StorageV2, and Generate and Assign an Identity for Azure KeyVault.

New-AzStorageAccount -ResourceGroupName MyResourceGroup -Name mystorageaccount -Location westus -SkuName Standard_GRS -Kind StorageV2 -AssignIdentity

This command creates a Storage account with Kind StorageV2. It also generates and assigns an identity that can be used to manage account keys through Azure KeyVault.

Example 4: Create a Storage account with NetworkRuleSet from JSON

```
New-AzStorageAccount -ResourceGroupName MyResourceGroup -Name mystorageaccount -Location westus -Type
Standard_LRS -NetworkRuleSet (@{bypass="Logging,Metrics";
    ipRules=(@{IPAddressOrRange="20.11.0.0/16";Action="allow"},
        @{IPAddressOrRange="10.0.0.0/7";Action="allow"});

virtualNetworkRules=(@{VirtualNetworkResourceId="/subscriptions/s1/resourceGroups/g1/providers/Microsoft.Network/virtu
alNetworks/vnet1/subnets/subnet1";Action="all
ow"},


@{VirtualNetworkResourceId="/subscriptions/s1/resourceGroups/g1/providers/Microsoft.Network/virtualNetworks/vnet2/sub
nets/subnet2";Action="allow"});
    defaultAction="Deny"})
```

This command creates a Storage account that has NetworkRuleSet property from JSON

Example 5: Create a Storage account with Hierarchical Namespace enabled, Sftp enabled, and localuser enabled.

```
New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -AccountName "mystorageaccount" -Location "US
West" -SkuName "Standard_GRS" -Kind StorageV2
-EnableHierarchicalNamespace $true -EnableSftp $true -EnableLocalUser $true
```

This command creates a Storage account with Hierarchical Namespace enabled, Sftp enabled, and localuser enabled.

Example 6: Create a Storage account with Azure Files Microsoft Entra Domain Services Authentication, and enable large file share.

```
New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -Name "mystorageaccount" -Location
"eastus2euap" -SkuName "Standard_LRS" -Kind StorageV2
```

-EnableAzureActiveDirectoryDomainServicesForFile $true -EnableLargeFileShare

This command creates a Storage account with Azure Files Microsoft Entra Domain Services Authentication, and enable large file share.

Example 7: Create a Storage account with enable Files Active Directory Domain Service Authentication and DefaultSharePermission.

New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -Name "mystorageaccount" -Location "eastus2euap" -SkuName "Standard_LRS" -Kind StorageV2
-EnableActiveDirectoryDomainServicesForFile $true `
-ActiveDirectoryDomainName "mydomain.com" `
-ActiveDirectoryNetBiosDomainName "mydomain.com" `
-ActiveDirectoryForestName "mydomain.com" `
-ActiveDirectoryDomainGuid "12345678-1234-1234-1234-123456789012" `
-ActiveDirectoryDomainSid "S-1-5-21-1234567890-1234567890-1234567890" `
-ActiveDirectoryAzureStorageSid "S-1-5-21-1234567890-1234567890-1234567890-1234" `
-ActiveDirectorySamAccountName "samaccountname" `
-ActiveDirectoryAccountType User `
-DefaultSharePermission StorageFileDataSmbShareElevatedContributor

This command creates a Storage account withenable Files Active Directory Domain Service Authentication and DefaultSharePermission.

Example 8: Create a Storage account with Queue and Table Service use account-scoped encryption key, and Require Infrastructure Encryption.

New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -Name "mystorageaccount" -Location "eastus2euap" -SkuName "Standard_LRS" -Kind StorageV2
-EncryptionKeyTypeForTable Account -EncryptionKeyTypeForQueue Account -RequireInfrastructureEncryption

```
$account = Get-AzStorageAccount -ResourceGroupName $rgname -Name $accountName

$account.Encryption.Services.Queue

Enabled LastEnabledTime     KeyType
------- ---------------     -------
   True 1/9/2020 6:09:11 AM Account

$account.Encryption.Services.Table

Enabled LastEnabledTime     KeyType
------- ---------------     -------
   True 1/9/2020 6:09:11 AM Account

$account.Encryption.RequireInfrastructureEncryption
True
```

This command creates a Storage account with Queue and Table Service use account-scoped encryption key and Require Infrastructure Encryption, so Queue and Table will use same encryption key with Blob and File service, and the service will apply a secondary layer of encryption with platform managed keys for data at rest. Then get the Storage account properties, and view the encryption keytype of Queue and Table Service, and RequireInfrastructureEncryption value.

Example 9: Create account MinimumTlsVersion and AllowBlobPublicAccess, and disable SharedKey Access

```
$account = New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -Name "mystorageaccount" -Location "eastus2euap" -SkuName "Standard_LRS" -Kind StorageV2
-MinimumTlsVersion TLS1_1 -AllowBlobPublicAccess $false -AllowSharedKeyAccess $false

$account.MinimumTlsVersion
TLS1_1
```

$account.AllowBlobPublicAccess

False

$a.AllowSharedKeyAccess

False

The command create account with MinimumTlsVersion, AllowBlobPublicAccess, and disable SharedKey access to the account, and then show the 3 properties of the created

account

Example 10: Create a Storage account with RoutingPreference setting

$account = New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -Name "mystorageaccount" -Location "eastus2euap" -SkuName "Standard_LRS"

-PublishMicrosoftEndpoint $true -PublishInternetEndpoint $true -RoutingChoice MicrosoftRouting

$account.RoutingPreference

RoutingChoice PublishMicrosoftEndpoints PublishInternetEndpoints

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

MicrosoftRouting True True

$account.PrimaryEndpoints

Blob : https://mystorageaccount.blob.core.windows.net/

Queue : https://mystorageaccount.queue.core.windows.net/

Table : https://mystorageaccount.table.core.windows.net/

File : https://mystorageaccount.file.core.windows.net/

Web : https://mystorageaccount.z2.web.core.windows.net/

Dfs : https://mystorageaccount.dfs.core.windows.net/

MicrosoftEndpoints :

{"Blob":"https://mystorageaccount-microsoftrouting.blob.core.windows.net/","Queue":"https://mystorageaccount-microsoftrouting.queue.core.windows.n

et/","Table":"https://mystorageaccount-microsoftrouting.table.core.windows.net/","File":"ht

tps://mystorageaccount-microsoftrouting.file.core.windows.net/","Web":"https://mystorageaccount-microsoftrouting.z2.web.core.windows.net/","Dfs":"

https://mystorageaccount-microsoftrouting.dfs.core.windows.net/"}

InternetEndpoints :

{"Blob":"https://mystorageaccount-internetrouting.blob.core.windows.net/","File":"https://mystorageaccount-internetrouting.file.core.windows.net/"

,"Web":"https://mystorageaccount-internetrouting.z2.web.core.windows.net/","Dfs":"https://w

eirp3-internetrouting.dfs.core.windows.net/"}

This command creates a Storage account with RoutingPreference setting: PublishMicrosoftEndpoint and PublishInternetEndpoint as true, and RoutingChoice as

MicrosoftRouting.

Example 11: Create a Storage account with EdgeZone and AllowCrossTenantReplication

$account = New-AzStorageAccount -ResourceGroupName "myresourcegroup" -Name "mystorageaccount" -SkuName Premium_LRS -Location westus -EdgeZone "microsoftlosangeles1"

-AllowCrossTenantReplication $false

$account.ExtendedLocation

Name Type

---- ----

microsoftlosangeles1 EdgeZone

$account.AllowCrossTenantReplication

False

This command creates a Storage account with EdgeZone as "microsoftlosangeles1" and AllowCrossTenantReplication as false, then show the created account related

properties.

Example 12: Create a Storage account with KeyExpirationPeriod and SasExpirationPeriod

```
$account = New-AzStorageAccount -ResourceGroupName "myresourcegroup" -Name "mystorageaccount" -SkuName Premium_LRS -Location eastus -KeyExpirationPeriodInDay 5
-SasExpirationPeriod "1.12:05:06"

$account.KeyPolicy.KeyExpirationPeriodInDays
5

$account.SasPolicy.SasExpirationPeriod
1.12:05:06
```

This command creates a Storage account with KeyExpirationPeriod and SasExpirationPeriod, then show the created account related properties.

Example 12: Create a Storage account with Keyvault encryption (access Keyvault with user assigned identity)

```
# Create KeyVault (no need if using exist keyvault)
$keyVault = New-AzKeyVault -VaultName $keyvaultName -ResourceGroupName $resourceGroupName -Location eastus2euap -EnablePurgeProtection
$key = Add-AzKeyVaultKey -VaultName $keyvaultName -Name $keyname -Destination 'Software'

# create user assigned identity and grant access to keyvault (no need if using exist user assigned identity)
$userId = New-AzUserAssignedIdentity -ResourceGroupName $resourceGroupName -Name $userIdName
Set-AzKeyVaultAccessPolicy -VaultName $keyvaultName -ResourceGroupName $resourceGroupName -ObjectId $userId.PrincipalId -PermissionsToKeys get,wrapkey,unwrapkey
-BypassObjectIdValidation
```

$useridentityId= $userId.Id

create Storage account with Keyvault encryption (access Keyvault with user assigned identity), then show properties

$account = New-AzStorageAccount -ResourceGroupName $resourceGroupName -Name $storageAccountName -Kind StorageV2 -SkuName Standard_LRS -Location eastus2euap `

-IdentityType SystemAssignedUserAssigned -UserAssignedIdentityId $useridentityId `

-KeyVaultUri $keyVault.VaultUri -KeyName $keyname -KeyVaultUserAssignedIdentityId $useridentityId

$account.Encryption.EncryptionIdentity

EncryptionUserAssignedIdentity

/subscriptions/{subscription-id}/resourceGroups/myresourcegroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/myuserid

$account.Encryption.KeyVaultProperties

KeyName : wrappingKey

KeyVersion :

KeyVaultUri : https://mykeyvault.vault.azure.net:443

CurrentVersionedKeyIdentifier : https://mykeyvault.vault.azure.net/keys/wrappingKey/8e74036e0d534e58b3bd84b319e31d8f

LastKeyRotationTimestamp : 4/12/2021 8:17:57 AM

This command first create a keyvault and a user assigned identity, then create a storage account with keyvault encryption (the storage access access keyvault with the

user assigned identity).

--------- Example 13: Create account with EnableNfsV3 ---------

$account = New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -Name "mystorageaccount" -SkuName

Standard_LRS -Location centraluseuap -Kind StorageV2

-EnableNfsV3 $true -EnableHierarchicalNamespace $true -EnableHttpsTrafficOnly $false -NetworkRuleSet (@{bypass="Logging,Metrics";

virtualNetworkRules=(@{VirtualNetworkResourceId="$vnet1";Action="allow"});

defaultAction="deny"})

$account.EnableNfsV3

True

The command create account with EnableNfsV3 as true, and then show the EnableNfsV3 property of the created account

- Example 14: Create account with disable PublicNetworkAccess -

$account = New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -Name "mystorageaccount" -SkuName Standard_LRS -Location centraluseuap -Kind StorageV2

-PublicNetworkAccess Disabled

$account.PublicNetworkAccess

Disabled

The command creates account with disable PublicNetworkAccess of the account.

Example 15: Create account with account level mmutability policy

$account = New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -Name "mystorageaccount" -SkuName Standard_LRS -Location centraluseuap -Kind StorageV2

-EnableAccountLevelImmutability -ImmutabilityPeriod 1 -ImmutabilityPolicyState Unlocked

$account.ImmutableStorageWithVersioning.Enabled

True

$account.ImmutableStorageWithVersioning.ImmutabilityPolicy

ImmutabilityPeriodSinceCreationInDays State

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

1 Unlocked

The command creates an account and enable account level immutability with versioning by '-EnableAccountLevelImmutability', then all the containers under this account

will have object-level immutability enabled by default. The account is also created with a default account-level immutability policy which is inherited and applied to

objects that do not possess an explicit immutability policy at the object level.

Example 16: Create a Storage account with enable Azure Files Active Directory Domain Service Kerberos Authentication.

New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -Name "mystorageaccount" -Location "eastus2euap" -SkuName "Standard_LRS" -Kind StorageV2

-EnableAzureActiveDirectoryKerberosForFile $true `

-ActiveDirectoryDomainName "mydomain.com" `

-ActiveDirectoryDomainGuid "12345678-1234-1234-1234-123456789012"

This command creates a Storage account with enable Azure Files Active Directory Domain Service Kerberos Authentication.

Example 17: Create a Storage account with Keyvault from another tenant (access Keyvault with FederatedClientId)

create Storage account with Keyvault encryption (access Keyvault with FederatedClientId), then show properties

$account = New-AzStorageAccount -ResourceGroupName $resourceGroupName -Name $storageAccountName -Kind StorageV2 -SkuName Standard_LRS -Location eastus2euap `

-IdentityType SystemAssignedUserAssigned -UserAssignedIdentityId $useridentityId `

-KeyVaultUri $keyVault.VaultUri -KeyName $keyname -KeyVaultUserAssignedIdentityId $useridentityId

-KeyVaultFederatedClientId $federatedClientId

$account.Encryption.EncryptionIdentity

EncryptionUserAssignedIdentity

EncryptionFederatedIdentityClientId

/subscriptions/{subscription-id}/resourceGroups/myresourcegroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/myuserid

********-****-****-****-************

$account.Encryption.KeyVaultProperties

KeyName : wrappingKey

KeyVersion :

KeyVaultUri : https://mykeyvault.vault.azure.net:443

CurrentVersionedKeyIdentifier : https://mykeyvault.vault.azure.net/keys/wrappingKey/8e74036e0d534e58b3bd84b319e31d8f

LastKeyRotationTimestamp : 3/3/2022 2:07:34 AM

This command creates a storage account with Keyvault from another tenant (access Keyvault with FederatedClientId).

Example 18: Create account with DnsEndpointType as AzureDnsZone

New-AzStorageAccount -ResourceGroupName "MyResourceGroup" -AccountName "mystorageaccount" -SkuName Standard_LRS -Location centraluseuap -Kind StorageV2

-DnsEndpointType AzureDnsZone

The command creates a storage account with DnsEndpointType as AzureDnsZone to create a large number of accounts in a single subscription, which creates accounts in an

Azure DNS Zone and the endpoint URL will have an alphanumeric DNS Zone identifier.

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/az.storage/new-azstorageaccount

Get-AzStorageAccount

Remove-AzStorageAccount

Set-AzStorageAccount