



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

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

NAME

New-AzElasticSanVolumeGroup

SYNOPSIS

Create a Volume Group.

SYNTAX

```
New-AzElasticSanVolumeGroup -ElasticSanName <String> -Name <String> -ResourceGroupName <String>
[-SubscriptionId <String>] [-Encryption <String>]
[-EncryptionUserAssignedIdentity <String>] [-IdentityType <String>] [-IdentityUserAssignedIdentityId <String>] [-KeyName
<String>] [-KeyVaultUri <String>]
[-KeyVersion <String>] [-NetworkAclsVirtualNetworkRule <IVirtualNetworkRule[]>] [-ProtocolType <String>]
[-DefaultProfile <PSObject>] [-AsJob] [-Break]
[-HttpPipelineAppend <SendAsyncStep[]>] [-HttpPipelinePrepend <SendAsyncStep[]>] [-NoWait] [-Proxy <Uri>]
[-ProxyCredential <PSCredential>]
[-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm] [<CommonParameters>]
```

```
New-AzElasticSanVolumeGroup -Name <String> -ElasticSanInputObject <IElasticSanIdentity> [-Encryption <String>]
[-EncryptionUserAssignedIdentity <String>]
```

[-IdentityType <String>] [-IdentityUserAssignedIdentityId <String>] [-KeyName <String>] [-KeyVaultUri <String>]
 [-KeyVersion <String>] [-NetworkAclsVirtualNetworkRule
 <IVirtualNetworkRule[]>] [-ProtocolType <String>] [-DefaultProfile <PSObject>] [-AsJob] [-Break] [-HttpPipelineAppend
 <SendAsyncStep[]>] [-HttpPipelinePrepend
 <SendAsyncStep[]>] [-NoWait] [-Proxy <Uri>] [-ProxyCredential <PSCredential>] [-ProxyUseDefaultCredentials] [-WhatIf]
 [-Confirm] [<CommonParameters>]

New-AzElasticSanVolumeGroup -InputObject <IElasticSanIdentity> [-Encryption <String>
 [-EncryptionUserAssignedIdentity <String>] [-IdentityType <String>
 [-IdentityUserAssignedIdentityId <String>] [-KeyName <String>] [-KeyVaultUri <String>] [-KeyVersion <String>
 [-NetworkAclsVirtualNetworkRule <IVirtualNetworkRule[]>
 [-ProtocolType <String>] [-DefaultProfile <PSObject>] [-AsJob] [-Break] [-HttpPipelineAppend <SendAsyncStep[]>
 [-HttpPipelinePrepend <SendAsyncStep[]>] [-NoWait]
 [-Proxy <Uri>] [-ProxyCredential <PSCredential>] [-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm]
 [<CommonParameters>]

DESCRIPTION

Create a Volume Group.

PARAMETERS

-ElasticSanName <String>

The name of the ElasticSan.

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-Name <String>

The name of the VolumeGroup.

Required? true
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-ResourceGroupName <String>

The name of the resource group.
The name is case insensitive.

Required? true
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-SubscriptionId <String>

The ID of the target subscription.

Required? false
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-ElasticSanInputObject <IElasticSanIdentity>

Identity Parameter

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

Required? true
Position? named
Default value

Accept pipeline input? true (ByValue)

Accept wildcard characters? false

-InputObject <IElasticSanIdentity>

Identity Parameter

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

Required? true

Position? named

Default value

Accept pipeline input? true (ByValue)

Accept wildcard characters? false

-Encryption <String>

Type of encryption

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-EncryptionUserAssignedIdentity <String>

Resource identifier of the UserAssigned identity to be associated with server-side encryption on the volume group.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-IdentityType <String>

The identity type.

Required? false
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-IdentityUserAssignedIdentityId <String>

Gets or sets a list of key value pairs that describe the set of User Assigned identities that will be used with this volume group.

The key is the ARM resource identifier of the identity.

Required? false
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-KeyName <String>

The name of KeyVault key.

Required? false
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-KeyVaultUri <String>

The Uri of KeyVault.

Required? false
Position? named
Default value

Accept pipeline input? false

Accept wildcard characters? false

-KeyVersion <String>

The version of KeyVault key.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-NetworkAclsVirtualNetworkRule <IVirtualNetworkRule[]>

The list of virtual network rules.

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

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-ProtocolType <String>

Type of storage target

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

Microsoft.Azure.PowerShell.Cmdlets.ElasticSan.Models.IElasticSanIdentity

OUTPUTS

Microsoft.Azure.PowerShell.Cmdlets.ElasticSan.Models.IVolumeGroup

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.

ELASTICSANINPUTOBJECT <IElasticSanIdentity>: Identity Parameter To construct, see NOTES section for ELASTICSANINPUTOBJECT properties and create a hash table.

[ElasticSanName <String>]: The name of the ElasticSan.

[Id <String>]: Resource identity path

[PrivateEndpointConnectionName <String>]: The name of the Private Endpoint connection.

[ResourceGroupName <String>]: The name of the resource group. The name is case insensitive.

[SnapshotName <String>]: The name of the volume snapshot within the given volume group.

[SubscriptionId <String>]: The ID of the target subscription.

[VolumeGroupName <String>]: The name of the VolumeGroup.

[VolumeName <String>]: The name of the Volume.

INPUTOBJECT <IElasticSanIdentity>: Identity Parameter To construct, see NOTES section for INPUTOBJECT properties and create a hash table.

[ElasticSanName <String>]: The name of the ElasticSan.

[Id <String>]: Resource identity path

[PrivateEndpointConnectionName <String>]: The name of the Private Endpoint connection.

[ResourceGroupName <String>]: The name of the resource group. The name is case insensitive.

[SnapshotName <String>]: The name of the volume snapshot within the given volume group.

[SubscriptionId <String>]: The ID of the target subscription.

[VolumeGroupName <String>]: The name of the VolumeGroup.

[VolumeName <String>]: The name of the Volume.

NETWORKACLSVIRTUALNETWORKRULE <IVirtualNetworkRule[]>: The list of virtual network rules. To construct, see NOTES section for NETWORKACLSVIRTUALNETWORKRULE properties and create a hash table.

VirtualNetworkResourceId <String>: Resource ID of a subnet, for example:

/subscriptions/{subscriptionId}/resourceGroups/{groupName}/providers/Microsoft.Network/virtualNetworks/{vnetName}/subnets/{subnetName}.

[Action <String>]: The action of virtual network rule.

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

```
PS C:\>$virtualNetworkRule1 = New-AzElasticSanVirtualNetworkRuleObject -VirtualNetworkResourceId

"/subscriptions/xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myresourcegroup/providers/Microsoft.Network/virtualNetworks/myvnet/subnets/subnet1" -Action Allow

$virtualNetworkRule2 = New-AzElasticSanVirtualNetworkRuleObject -VirtualNetworkResourceId

"/subscriptions/xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myresourcegroup/providers/Microsoft.Network/virtualNetworks/myvnet/subnets/subnet2" -Action Allow

New-AzElasticSanVolumeGroup -ResourceGroupName myresourcegroup -ElasticSanName myelasticsan -Name myvolumeGroup -ProtocolType 'Iscsi' -NetworkAcIsVirtualNetworkRule

$virtualNetworkRule1,$virtualNetworkRule2
```

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

```
PS C:\>New-AzElasticSanVolumeGroup -ResourceGroupName myresourcegroup -ElasticSanName myelasticsan -Name  
myvolumegroup -ProtocolType 'Iscsi' `
```

```
-NetworkAclsVirtualNetworkRule (
```

```
@{VirtualNetworkResourceId="/subscriptions/xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myresourcegroup/pro  
viders/Microsoft.Network/virtualNetwo  
rks/myvnet/subnets/subnet1";  
    Action="Allow"},
```

```
@{VirtualNetworkResourceId="/subscriptions/xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myresourcegroup/pro  
viders/Microsoft.Network/virtualNetwo  
rks/myvnet/subnets/subnet2";  
    Action="Allow"})
```

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

```
PS C:\>New-AzElasticSanVolumeGroup -ResourceGroupName myresourcegroup -ElasticSanName myelasticsan -Name  
myvolumegroup -IdentityType SystemAssigned -ProtocolType  
Iscsi -Encryption EncryptionAtRestWithPlatformKey
```

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

```
PS C:\>$useridentity = Get-AzUserAssignedIdentity -ResourceGroupName myresoucegroup -Name myuai
```

```
New-AzElasticSanVolumeGroup -ResourceGroupName myresoucegroup -ElasticSanName myelasticsan -Name  
myvolumegroup -IdentityType UserAssigned
```

```
-IdentityUserAssignedIdentityId $useridentity.Id -Encryption EncryptionAtRestWithCustomerManagedKey -KeyName  
mykey -KeyVaultUri
```

```
"https://mykeyvault.vault.azure.net:443" -EncryptionUserAssignedIdentity $useridentity.Id -ProtocolType Iscsi
```

RELATED LINKS

<https://learn.microsoft.com/powershell/module/az.elasticsan/new-azelasticsanvolumegroup>