



Windows PowerShell Get-Help on Cmdlet 'Set-AzStorageBlobInventoryPolicy'

PS:\>Get-HELP Set-AzStorageBlobInventoryPolicy -Full

NAME

Set-AzStorageBlobInventoryPolicy

SYNOPSIS

Creates or updates blob inventory policy in a Storage account.

SYNTAX

```
Set-AzStorageBlobInventoryPolicy [-ResourceGroupName] <System.String> [-StorageAccountName] <System.String>
[-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Disabled] -Rule
<Microsoft.Azure.Commands.Management.Storage.Models.PSBlobInventoryPolicyRule[]> [-Confirm] [-WhatIf]
[<CommonParameters>]
```

Set-AzStorageBlobInventoryPolicy [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Disabled] -Rule
 <Microsoft.Azure.Commands.Management.Storage.Models.PSBlobInventoryPolicyRule[]> -StorageAccount
 <Microsoft.Azure.Commands.Management.Storage.Models.PSStorageAccount>
 [-Confirm] [-WhatIf] [<CommonParameters>]

```
Set-AzStorageBlobInventoryPolicy [-StorageAccountResourceId] <System.String> [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Disabled] -Rule
    <Microsoft.Azure.Commands.Management.Storage.Models.PSBlobInventoryPolicyRule[]> [-Confirm] [-WhatIf]
[<CommonParameters>]
```

```
Set-AzStorageBlobInventoryPolicy [-ResourceGroupName] <System.String> [-StorageAccountName] <System.String>
[-DefaultProfile
    <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -Policy
    <Microsoft.Azure.Commands.Management.Storage.Models.PSBlobInventoryPolicy> [-Confirm] [-WhatIf]
[<CommonParameters>]
```

```
Set-AzStorageBlobInventoryPolicy [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -Policy
    <Microsoft.Azure.Commands.Management.Storage.Models.PSBlobInventoryPolicy> -StorageAccount
    <Microsoft.Azure.Commands.Management.Storage.Models.PSStorageAccount>
[-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Set-AzStorageBlobInventoryPolicy [-StorageAccountResourceId] <System.String> [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -Policy
    <Microsoft.Azure.Commands.Management.Storage.Models.PSBlobInventoryPolicy> [-Confirm] [-WhatIf]
[<CommonParameters>]
```

DESCRIPTION

The Set-AzStorageBlobInventoryPolicy cmdlet creates or updates blob inventory policy in a Storage account.

PARAMETERS

-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

-Disabled <System.Management.Automation.SwitchParameter>

The Blob Inventory Policy is enabled by default, specify this parameter to disable it.

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

-Policy <Microsoft.Azure.Commands.Management.Storage.Models.PSBlobInventoryPolicy>

Blob Inventory Policy Object to Set

Required? true
Position? named
Default value None
Accept pipeline input? True (ByPropertyName, ByValue)
Accept wildcard characters? false

-ResourceGroupName <System.String>

Resource Group Name.

Required? true
Position? 0
Default value None
Accept pipeline input? False
Accept wildcard characters? false

-Rule <Microsoft.Azure.Commands.Management.Storage.Models.PSBlobInventoryPolicyRule[]>

The Blob Inventory Policy rules. Get the object with New-AzStorageBlobInventoryPolicyRule cmdlet.

Required? true
Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-StorageAccount <Microsoft.Azure.Commands.Management.Storage.Models.PSStorageAccount>

Storage account object

Required? true
Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-StorageAccountName <System.String>

Storage Account Name.

Required? true
Position? 1
Default value None
Accept pipeline input? False
Accept wildcard characters? false

-StorageAccountResourceId <System.String>

Storage Account Resource Id.

Required? true
Position? 0
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

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

System.String

Microsoft.Azure.Commands.Management.Storage.Models.PSBlobInventoryPolicySchema

OUTPUTS

Microsoft.Azure.Commands.Management.Storage.Models.PSManagementPolicy

NOTES

Example 1: Create or update the blob inventory policy with BlobInventoryPolicy rule objects.

```
$rule1 = New-AzStorageBlobInventoryPolicyRule -Name Test1 -Destination $containerName -Disabled -Format Csv  
-Schedule Daily -ContainerSchemaField
```

```
Name,Metadata,PublicAccess,Last-Modified,LeaseStatus,LeaseState,LeaseDuration,HasImmutabilityPolicy,HasLegalHold  
-PrefixMatch con1,con2
```

```
$rule2 = New-AzStorageBlobInventoryPolicyRule -Name Test2 -Destination $containerName -Format Parquet -Schedule  
Weekly -IncludeBlobVersion -IncludeSnapshot -BlobType  
blockBlob,appendBlob -PrefixMatch aaa,bbb `  
-BlobSchemaField
```

```
name,Creation-Time,Last-Modified,Content-Length,Content-MD5,BlobType,AccessTier,AccessTierChangeTime,Expiry-Time,  
e,hdi_isfolder,Owner,Group,Permissions,Acl,Metadata
```

```
$rule3 = New-AzStorageBlobInventoryPolicyRule -Name Test3 -Destination $containerName -Format Parquet -Schedule
```

Weekly -IncludeBlobVersion -IncludeSnapshot
-IncludeDeleted -BlobType blockBlob,appendBlob -PrefixMatch aaa,bbb `
-ExcludePrefix ccc,ddd -BlobSchemaField

name,Last-Modified,BlobType,AccessTier,AccessTierChangeTime,Content-Type,Content-CRC64,CopyId,x-ms-blob-sequence-number,TagCount

\$policy = Set-AzStorageBlobInventoryPolicy -ResourceGroupName "myresourcegroup" -AccountName "mystorageaccount" -Disabled -Rule \$rule1,\$rule2,\$rule3

\$policy

StorageAccountName : mystorageaccount
ResourceGroupName : myresourcegroup
Name : DefaultInventoryPolicy
Id :

/subscriptions/{subscription-Id}/resourceGroups/myresourcegroup/providers/Microsoft.Storage/storageAccounts/mystorageaccount/inventoryPolicies/default

Type : Microsoft.Storage/storageAccounts/inventoryPolicies
LastModifiedTime : 5/12/2021 8:53:38 AM
Enabled : False
Rules : {Test1, Test2, Test3}

\$policy.Rules

Name	Enabled	Destination	ObjectType	Format	Schedule	IncludeSnapshots	IncludeBlobVersions	IncludeDeleted
BlobTypes	PrefixMatch	ExcludePrefix	SchemaFields					

Test1	False	containername	Container	Csv	Daily			{con1, con2}
{Name,								

```

Metadata, PublicAccess, Last-Modified...}

Test2 True    containername Blob    Parquet Weekly    True    True    {blockBlob, appendBlob}
{aaa, bbb}    {Name,
Creation-Time, Last-Modified, Content-Length...}

Test3 True    containername Blob    Parquet Weekly    True    True    True    {blockBlob, appendBlob}
{aaa, bbb} {ccc, ddd}    {Name,
Content-Type, Content-CRC64, Last-Modified...}

```

This first 2 commands create 3 BlobInventoryPolicy rule objects: rule "Test1" for container inventory; rule "Test2" and "Test3" for blob inventory. The following command sets blob inventory policy to a Storage account with the 2 rule objects, then show the updated policy and rules properties.

Example 2: Create or update the blob inventory policy of a Storage account with a Json format policy.

```

$policy = Set-AzStorageBlobInventoryPolicy -ResourceGroupName $resourceGroupName -StorageAccountName
$accountName -Policy (@{
    Enabled=$true;
    Rules=(@{
        Enabled=$true;
        Name="Test1";
        Destination=$containerName;
        Definition=(@{
            ObjectType="Blob";
            Format="Csv";
            Schedule="Weekly";
            SchemaFields=@("name","Content-Length","BlobType","Snapshot","VersionId","IsCurrentVersion");
            Filters=(@{
                BlobTypes=@("blockBlob","appendBlob");
                PrefixMatch=@("prefix1","prefix2");
                IncludeSnapshots=$true;
                IncludeBlobVersions=$true;

```

```

    })
  })
},
@{
  Enabled=$false;
  Name="Test2";
  Destination=$containerName;
  Definition=(@{
    ObjectType="Container";
    Format="Parquet";
    Schedule="Daily";

```

```
SchemaFields=@("name","Metadata","PublicAccess","DefaultEncryptionScope","DenyEncryptionScopeOverride");
```

```

  Filters=(@{
    PrefixMatch=@("conpre1","conpre2");
  })
})
},
@{

```

```

  Enabled=$false;
  Name="Test3";
  Destination=$containerName;
  Definition=(@{
    ObjectType="Blob";
    Format="Csv";
    Schedule="Weekly";

```

```
SchemaFields=@("name","Deleted","RemainingRetentionDays","Content-Type","Content-Language","Cache-Control","Content-Disposition");
```

```

  Filters=(@{
    BlobTypes=@("blockBlob","appendBlob");
    PrefixMatch=@("conpre1","conpre2");
    ExcludePrefix=@("expre1","expre2");

```

```

        IncludeDeleted=$true
    })
})
})
})

```

\$policy

StorageAccountName : weiadlscanary1

ResourceGroupName : weitry

Name : DefaultInventoryPolicy

Id :

/subscriptions/{subscription-Id}/resourceGroups/weitry/providers/Microsoft.Storage/storageAccounts/weiadlscanary1/invento
ryPolicies/default

Type : Microsoft.Storage/storageAccounts/inventoryPolicies

LastModifiedTime : 5/12/2021 9:02:21 AM

Enabled : True

Rules : {Test1, Test2, Test3}

\$policy.Rules

Name	Enabled	Destination	ObjectType	Format	Schedule	IncludeSnapshots	IncludeBlobVersions	IncludeDeleted
BlobTypes	PrefixMatch	ExcludePrefix	SchemaFields					

Test1	True	containername	Blob	Csv	Weekly	True		
-------	------	---------------	------	-----	--------	------	--	--

{prefix1, prefix2}

{Name, Content-Length, BlobType, Snapshot...}

Test2	False	containername	Container	Parquet	Daily			{conpre1, conpre2}
-------	-------	---------------	-----------	---------	-------	--	--	-----------------------

{Name, Metadata, PublicAccess}

```
Test3 False    containername Blob    Csv    Weekly    True    {blockBlob, appendBlob}
{conpre1, conpre2} {expre1,
    expre2} {Name, Content-Type, Content-Cache, Content-Language...}
{name, Metadata,
    PublicAccess}
```

This command creates or updates the blob inventory policy of a Storage account with a json format policy.

Example 3: Get the blob inventory policy from a Storage account, then set it to another Storage account.

```
$policy = Get-AzStorageBlobInventoryPolicy -ResourceGroupName "myresourcegroup" -AccountName
"mystorageaccount" | Set-AzStorageBlobInventoryPolicy -ResourceGroupName
"myresourcegroup2" -AccountName "mystorageaccount2"
```

This command first gets the blob inventory policy from a Storage account, then set it to another Storage account. The properties: Destination, Enabled, and Rules of the policy will be set to the destination account.

Example 4: Get the blob inventory policy rules from a Storage account, then set it to another Storage account.

```
$policy = ,((Get-AzStorageBlobInventoryPolicy -ResourceGroupName "myresourcegroup" -AccountName
"mystorageaccount").Rules) | Set-AzStorageBlobInventoryPolicy
-ResourceGroupName "myresourcegroup2" -AccountName "mystorageaccount2" -Disabled
```

This command first gets the blob inventory policy from a Storage account, then set its rules to another Storage account.

RELATED LINKS

Online Version: <https://learn.microsoft.com/powershell/module/az.storage/set-azstorageblobinventorypolicy>

