Windows PowerShell Get-Help on Cmdlet 'New-AzNetworkWatcherPacketCaptureV2'

PS:\>Get-HELP New-AzNetworkWatcherPacketCaptureV2 -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-AzNetworkWatcherPacketCaptureV2

SYNOPSIS

V2 Version of Packet Capture Cmdlet which creates a new packet capture resource and starts a packet capture session on a VM, VMSS or few instances of VMSS.

SYNTAX

New-AzNetworkWatcherPacketCaptureV2 [-AsJob] [-BytesToCapturePerPacket <System.Nullable`1[System.Int32]>] [-DefaultProfile

<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Filter <Microsoft.Azure.Commands.Network.Models.PSPacketCaptureFilter[]>]

[-LocalFilePath <System.String>] -Location <System.String> -Name <System.String> [-Scope <Microsoft.Azure.Commands.Network.Models.PSPacketCaptureMachineScope>]

[-StorageAccountId <System.String>] [-StoragePath <System.String>] -TargetId <System.String> [-TargetType

<System.String>] [-TimeLimitInSecond
<System.Nullable`1[System.Int32]>] [-TotalBytesPerSession <System.Nullable`1[System.UInt32]>] [-Confirm] [-WhatIf] [<CommonParameters>]

New-AzNetworkWatcherPacketCaptureV2 [-AsJob] [-BytesToCapturePerPacket <System.Nullable`1[System.Int32]>] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Filter <Microsoft.Azure.Commands.Network.Models.PSPacketCaptureFilter[]>]
[-LocalFilePath <System.String>] -Name <System.String> -NetworkWatcher <Microsoft.Azure.Commands.Network.Models.PSNetworkWatcher> [-Scope
<Microsoft.Azure.Commands.Network.Models.PSPacketCaptureMachineScope>] [-StorageAccountId <System.String>] [-StoragePath <System.String>] -TargetId <System.String>
[-TargetType <System.String>] [-TimeLimitInSecond <System.Nullable`1[System.Int32]>] [-TotalBytesPerSession <System.Nullable`1[System.UInt32]>] [-Confirm] [-WhatIf]
[<CommonParameters>]

New-AzNetworkWatcherPacketCaptureV2 [-AsJob] [-BytesToCapturePerPacket <System.Nullable`1[System.Int32]>] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Filter <Microsoft.Azure.Commands.Network.Models.PSPacketCaptureFilter[]>]
[-LocalFilePath <System.String>] -Name <System.String> -NetworkWatcherName <System.String> -ResourceGroupName <System.String> [-Scope
<Microsoft.Azure.Commands.Network.Models.PSPacketCaptureMachineScope>] [-StorageAccountId <System.String>] [-StoragePath <System.String>] -TargetId <System.String>
[-TargetType <System.String>] [-TimeLimitInSecond <System.Nullable`1[System.Int32]>] [-TotalBytesPerSession <System.Nullable`1[System.UInt32]>] [-Confirm] [-WhatIf]
[<CommonParameters>]

DESCRIPTION

The New-AzNetworkWatcherPacketCaptureV2 cmdlet creates a new packet capture resource and starts a packet capture session on a VM, VMSS or few instances of VMSS. The
length of the Packet Capture sessions can be configured via a time constraint or a size constraint. The amount of data

captured for each packet can also be

configured. Filters can be applied to a given packet capture session, allowing you to customize the type of packets captured. Filters can restrict packets on local

and remote IP addresses & address ranges, local and remote ports & port ranges, and the session level protocol to be captured. Filters are composable, and multiple

filters can be applied to provide you with granularity of capture.

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

-BytesToCapturePerPacket <System.Nullable`1[System.Int32]>

Bytes to capture per packet.

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

-Filter <Microsoft.Azure.Commands.Network.Models.PSPacketCaptureFilter[]>

Filters for packet capture session.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-LocalFilePath <System.String>

Local file path.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Location <System.String>

Location of the network watcher.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Name <System.String>

The packet capture name.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-NetworkWatcher <Microsoft.Azure.Commands.Network.Models.PSNetworkWatcher>

The network watcher resource.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-NetworkWatcherName <System.String>

The name of network watcher.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-ResourceGroupName <System.String>

The name of the network watcher resource group.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Scope <Microsoft.Azure.Commands.Network.Models.PSPacketCaptureMachineScope>

Scope of VMSS Instances to be Included or Excluded.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-StorageAccountId <System.String>

Storage account Id.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-StoragePath <System.String>

Storage path.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-TargetId <System.String>

The target virtual machine ID or virtual machine scale set ID

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-TargetType <System.String>

Target Type of the Resource.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-TimeLimitInSecond <System.Nullable`1[System.Int32]>

Time limit in seconds.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-TotalBytesPerSession <System.Nullable`1[System.UInt32]>

Total bytes per session.

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

Microsoft.Azure.Commands.Network.Models.PSNetworkWatcher

System.String

System.Nullable`1[[System.Int32, System.Private.CoreLib, Version=6.0.0.0, Culture=neutral, PublicKeyToken=7cec85d7bea7798e]]

OUTPUTS

Microsoft.Azure.Commands.Network.Models.PSPacketCaptureResult

NOTES

---------- Example 1: Create a Packet Capture on a VM ----------

$nw = Get-AzResource | Where {$_.ResourceType -eq "Microsoft.Network/networkWatchers" -and $_.Location -eq "WestCentralUS" }

$networkWatcher = Get-AzNetworkWatcher -Name $nw.Name -ResourceGroupName $nw.ResourceGroupName

$storageAccount = Get-AzStorageAccount -ResourceGroupName contosoResourceGroup -Name contosostorage123

$filter1 = New-AzPacketCaptureFilterConfig -Protocol TCP -RemoteIPAddress "1.1.1.1-255.255.255" -LocalIPAddress "10.0.0.3" -LocalPort "1-65535" -RemotePort "20;80;443"

$filter2 = New-AzPacketCaptureFilterConfig -Protocol UDP

New-AzNetworkWatcherPacketCaptureV2 -NetworkWatcher $networkWatcher -TargetId $vm.Id -TargetType "azurevm" -Name "PacketCaptureTest" -StorageAccountId

$storageAccount.id -TimeLimitInSecond 60 -Filter $filter1, $filter2

Name : PacketCaptureTest

Id :

/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/NetworkWatcherRG/providers/Microsoft.Network/networkWatchers/NetworkWatche

r_eastus/packetCaptures/PacketCaptureTest

Etag : W/"0b3c52cb-aa63-4647-93d3-3221c13ccdd2"

ProvisioningState : Succeeded

Target :

/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/contosoResourceGroup/providers/Microsoft.Compute/virtualMachines/SampleVM

TargetType : AzureVM

BytesToCapturePerPacket : 0

TotalBytesPerSession : 1073741824

TimeLimitInSeconds : 18000

StorageLocation : {

"StorageId":

"/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/contosoResourceGroup/providers/Microsoft.Storage/storageAccounts/contosostorage123",

"StoragePath":

"https://contosostorage123.blob.core.windows.net/network-watcher-logs/subscriptions/00000000-0000-0000-0000-000000000000/resourcegroups/contosoResourceGroup/providers/microsoft.compute/virtualmachines/SampleVM/2022/07/21/packetcapture_09_20_07_166.cap"

}

Filters : [

{

"Protocol": "TCP",

"RemoteIPAddress": "1.1.1.1-255.255.255",

"LocalIPAddress": "10.0.0.3",

"LocalPort": "1-65535",

"RemotePort": "20;80;443"

},

{

"Protocol": "UDP",

"RemoteIPAddress": "",

"LocalIPAddress": "",

"LocalPort": "",

"RemotePort": ""

}

]

Scope : {

"Include": [],

"Exclude": []

}

In this example we create a packet capture named "PacketCaptureTest" with multiple filters and a time limit. Once the session is complete, it will be saved to the

specified storage account. Note: The Azure Network Watcher extension must be installed on the target virtual machine to create packet captures.

--------- Example 2: Create a Packet Capture on a VMSS ---------

$nw = Get-AzResource | Where {$_.ResourceType -eq "Microsoft.Network/networkWatchers" -and $_.Location -eq "WestCentralUS" }

$networkWatcher = Get-AzNetworkWatcher -Name $nw.Name -ResourceGroupName $nw.ResourceGroupName

$storageAccount = Get-AzStorageAccount -ResourceGroupName contosoResourceGroup -Name contosostorage123

$filter1 = New-AzPacketCaptureFilterConfig -Protocol TCP -RemoteIPAddress "1.1.1.1-255.255.255" -LocalIPAddress "10.0.0.3" -LocalPort "1-65535" -RemotePort "20;80;443"

$filter2 = New-AzPacketCaptureFilterConfig -Protocol UDP

New-AzNetworkWatcherPacketCaptureV2 -NetworkWatcher $networkWatcher -TargetId $vmss.Id -TargetType "azurevmss" -Name "PacketCaptureTest" -StorageAccountId

$storageAccount.id -TimeLimitInSecond 60 -Filter $filter1, $filter2

Name : PacketCaptureTest

Id :

/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/NetworkWatcherRG/providers/Microsoft.Network/networkWatchers/NetworkWatche

r_eastus/packetCaptures/PacketCaptureTest

Etag : W/"0b3c52cb-aa63-4647-93d3-3221c13ccdd2"

```
    ProvisioningState       : Succeeded
    Target                  :

/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/contosoResourceGroup/providers/Microsoft.Com
pute/virtualMachineScaleSets/SampleVMSS
    TargetType              : AzureVMSS
    BytesToCapturePerPacket : 0
    TotalBytesPerSession    : 1073741824
    TimeLimitInSeconds      : 60
    StorageLocation         : {
                              "StorageId":

"/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/contosoResourceGroup/providers/Microsoft.Sto
rage/storageAccounts/contosostorage123",
                                                                                                "StoragePath":
"https://contosostorage123.blob.core.windows.net/network-watcher-logs/subscriptions/00000000-0000-0000-0000-0000000
00000/res

ourcegroups/contosoResourceGroup/providers/microsoft.compute/virtualmachinescalesets/SampleVMSS/2022/07/21/packe
tcapture_09_20_07_166.cap"
                            }
    Filters                 : [
                              {
                                "Protocol": "TCP",
                                "RemoteIPAddress": "1.1.1.1-255.255.255",
                                "LocalIPAddress": "10.0.0.3",
                                "LocalPort": "1-65535",
                                "RemotePort": "20;80;443"
                              },
                              {
                                "Protocol": "UDP",
                                "RemoteIPAddress": "",
                                "LocalIPAddress": "",
```

```
                    "LocalPort": "",
                    "RemotePort": ""
                  }
                ]
  Scope           : {
                  "Include": [],
                  "Exclude": []
                }
```

In this example we create a packet capture named "PacketCaptureTest" with multiple filters and a time limit. Once the session is complete, it will be saved to the

specified storage account. Note: The Azure Network Watcher extension must be installed on the target virtual machine scale set and all the respective instances

adhering to the latest vmss model, to create packet captures.

- Example 3: Create a Packet Capture on few Instances of VMSS -

```
$nw = Get-AzResource | Where {$_.ResourceType -eq "Microsoft.Network/networkWatchers" -and $_.Location -eq "WestCentralUS" }
$networkWatcher = Get-AzNetworkWatcher -Name $nw.Name -ResourceGroupName $nw.ResourceGroupName

$storageAccount = Get-AzStorageAccount -ResourceGroupName contosoResourceGroup -Name contosostorage123

$filter1 = New-AzPacketCaptureFilterConfig -Protocol TCP -RemoteIPAddress "1.1.1.1-255.255.255" -LocalIPAddress "10.0.0.3" -LocalPort "1-65535" -RemotePort "20;80;443"
$filter2 = New-AzPacketCaptureFilterConfig -Protocol UDP

$instance1 = $vmssInstance1.Name
$instance2 = $vmssInstance2.Name
$scope = New-AzPacketCaptureScopeConfig -Include $instance1, $instance2

New-AzNetworkWatcherPacketCaptureV2 -NetworkWatcher $networkWatcher -TargetId $vmss.Id -TargetType
```

"azurevmss" -Scope $scope -Name "PacketCaptureTest"

-StorageAccountId $storageAccount.id -TimeLimitInSecond 60 -Filter $filter1, $filter2

Name : PacketCaptureTest

Id :

/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/NetworkWatcherRG/providers/Microsoft.Network

/networkWatchers/NetworkWatche

r_eastus/packetCaptures/PacketCaptureTest

Etag : W/"0b3c52cb-aa63-4647-93d3-3221c13ccdd2"

ProvisioningState : Succeeded

Target :

/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/contosoResourceGroup/providers/Microsoft.Com

pute/virtualMachineScaleSets/SampleVMSS

TargetType : AzureVMSS

BytesToCapturePerPacket : 0

TotalBytesPerSession : 1073741824

TimeLimitInSeconds : 18000

StorageLocation : {

"StorageId":

"/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/contosoResourceGroup/providers/Microsoft.Sto

rage/storageAccounts/contosostorage123",

"StoragePath":

"https://contosostorage123.blob.core.windows.net/network-watcher-logs/subscriptions/00000000-0000-0000-0000-0000000

00000/res

ourcegroups/contosoResourceGroup/providers/microsoft.compute/virtualmachinescalesets/SampleVMSS/2022/07/21/packe

tcapture_09_20_07_166.cap"

}

Filters : [

{

"Protocol": "TCP",

```
                    "RemoteIPAddress": "1.1.1.1-255.255.255",
                    "LocalIPAddress": "10.0.0.3",
                    "LocalPort": "1-65535",
                    "RemotePort": "20;80;443"
                  },
                  {
                    "Protocol": "UDP",
                    "RemoteIPAddress": "",
                    "LocalIPAddress": "",
                    "LocalPort": "",
                    "RemotePort": ""
                  }
                ]
  Scope           : {
                  "Include": [
"/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/contosoResourceGroup/providers/Microsoft.Compute/virtualMachineScaleS
  ets/SampleVMSS/virtualMachines/0",
"/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/contosoResourceGroup/providers/Microsoft.Compute/virtualMachineScaleS
  ets/SampleVMSS/virtualMachines/1"
                  ],
                  "Exclude": []
                }
```

In this example we create a packet capture named "PacketCaptureTest" with multiple filters and a time limit. Once the session is complete, it will be saved to the specified storage account. Note: The Azure Network Watcher extension must be installed on the target virtual machine scale set and on the respective instances in include scope adhering to the latest vmss model, to create packet captures.

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/az.network/new-aznetworkwatcherpacketcapturev2

New-AzNetworkWatcher

Get-AzNetworkWatcher

Remove-AzNetworkWatcher

Get-AzNetworkWatcherNextHop

Get-AzNetworkWatcherSecurityGroupView

Get-AzNetworkWatcherTopology

Start-AzNetworkWatcherResourceTroubleshooting

New-AzNetworkWatcherPacketCapture

New-AzPacketCaptureFilterConfig

New-AzNetworkWatcherPacketCapture

Get-AzNetworkWatcherPacketCapture

Remove-AzNetworkWatcherPacketCapture

Stop-AzNetworkWatcherPacketCapture

New-AzNetworkWatcherProtocolConfiguration

Test-AzNetworkWatcherIPFlow

Test-AzNetworkWatcherConnectivity

Stop-AzNetworkWatcherConnectionMonitor

Start-AzNetworkWatcherConnectionMonitor

Set-AzNetworkWatcherConnectionMonitor

Set-AzNetworkWatcherConfigFlowLog

Remove-AzNetworkWatcherConnectionMonitor

New-AzNetworkWatcherConnectionMonitor

Get-AzNetworkWatcherTroubleshootingResult

Get-AzNetworkWatcherReachabilityReport

Get-AzNetworkWatcherReachabilityProvidersList

Get-AzNetworkWatcherFlowLogStatus

Get-AzNetworkWatcherConnectionMonitorReport

Get-AzNetworkWatcherConnectionMonitor