



python



PowerShell

FPDF Library
PDF generator

Full credit is given to the above companies including the OS that this PDF file was generated!

PowerShell Get-Help on command 'Get-NetIPAddress'

PS C:\Users\wahid> Get-Help Get-NetIPAddress

NAME

Get-NetIPAddress

SYNOPSIS

Gets the IP address configuration.

SYNTAX

```
Get-NetIPAddress [[-IPAddress] <String[]>] [-AddressFamily {IPv4 | IPv6}]  
[-AddressState {Invalid | Tentative | Duplicate | Deprecated | Preferred}]  
[-AsJob] [-AssociatedIPInterface <CimInstance>] [-CimSession <CimSession[]>]  
[-IncludeAllCompartments] [-InterfaceAlias <String[]>] [-InterfaceIndex  
<UInt32[]>] [-PolicyStore <String>] [-PreferredLifetime <TimeSpan[]>]  
[-PrefixLength <Byte[]>] [-PrefixOrigin {Other | Manual | WellKnown | Dhcp |  
RouterAdvertisement}] [-SkipAsSource <Boolean[]>] [-SuffixOrigin {Other |  
Manual | WellKnown | Dhcp | Link | Random}] [-ThrottleLimit <Int32>] [-Type  
{Unicast | Anycast}] [-ValidLifetime <TimeSpan[]>] [<CommonParameters>]
```

DESCRIPTION

The Get-NetIPAddress cmdlet gets the IP address configuration, such as IPv4

addresses, IPv6 addresses and the IP interfaces with which addresses are associated. Without parameters, this cmdlet gets the entire IP address configuration for the computer.

PARAMETERS

`-AddressFamily <AddressFamily[]>`

Specifies an array of IP address families. The cmdlet gets the IP address configuration that matches the address families. The acceptable values for this parameter are:

-- IPv4 -- IPv6

`-AddressState <AddressState[]>`

Specifies an array of duplicate address detection (DAD) state values for the IP address. Use this parameter to filter the output based on addresses that are valid and available for use. The acceptable values for this parameter are:

-- Invalid. IP address configuration information for addresses that are not valid and will not be used.

-- Tentative. IP address configuration information for addresses that are not used for communication, as the uniqueness of those IP addresses is being verified.

-- Duplicate. IP address configuration information for addresses for which a duplicate IP address has been detected and the current IP address will not be used.

-- Deprecated. IP address configuration information for addresses that will no longer be used to establish new connections, but will continue to be used with existing connections.

-- Preferred. IP address configuration information for addresses that are valid and available for use.

-AsJob [<SwitchParameter>]

Runs the cmdlet as a background job. Use this parameter to run commands that take a long time to complete.

-AssociatedIPInterface <CimInstance>

Specifies an IP interface as a CIM object. To obtain an IP interface, use the Get-NetIPInterface cmdlet.

-CimSession <CimSession[]>

Runs the cmdlet in a remote session or on a remote computer. Enter a computer name or a session object, such as the output of a New-CimSession (<https://go.microsoft.com/fwlink/p/?LinkId=227967>) or [Get-CimSession](<https://go.microsoft.com/fwlink/p/?LinkId=227966>)cmdlet. The default is the current session on the local computer.

-IPAddress <String[]>

Specifies an array of IPv4 or IPv6 addresses.

-IncludeAllCompartments [<SwitchParameter>]

Indicates that the cmdlet includes addresses from all configured network compartments. If you do not specify this parameter, the cmdlet gets only addresses in the default network compartment.

-InterfaceAlias <String[]>

Specifies an array of aliases of network interfaces. The cmdlet gets IP addresses that match the aliases.

-InterfaceIndex <UInt32[]>

Specifies an array of indexes of network interfaces. The cmdlet gets IP addresses that match the indexes.

-PolicyStore <String>

Specifies a PolicyStore value. The acceptable values for this parameter are:

-- ActiveStore. The IP address information is valid.

-- PersistentStore. The computer saves IP address information across restarts. When the computer restarts, it copies the saved settings to the ActiveStore.

The default value is ActiveStore.

-PreferredLifetime <TimeSpan[]>

Specifies an array of values of preferred lifetimes, as TimeSpan objects, for an IP address. To obtain a TimeSpan object, use the New-TimeSpan cmdlet.

-PrefixLength <Byte[]>

Specifies an array of prefix lengths. This parameter defines the local subnet size, and is also known as a subnet mask.

-PrefixOrigin <PrefixOrigin[]>

Specifies an array of origins for address prefixes. IP addresses are divided into two parts, the prefix and the suffix. The address prefix identifies the network portion of an IP address, and the address suffix identifies the host portion. The acceptable values for this parameter are:

-- Manual. The IP address prefix was manually specified.

-- WellKnown. The IP address prefix is from a well-known source.

-- DHCP. The IP address prefix was provided by DHCP settings.

-- RouterAdvertisement. The IP address prefix was obtained through a router advertisement (RA).

-- Other. The IP address prefix was obtained from another source, such as a VPN.

-SkipAsSource <Boolean[]>

Indicates whether an address is a primary IP address. This parameter identifies the primary IP address for outgoing traffic in a multiple IP address scenario. If this parameter is set to True, the addresses are not used for outgoing traffic and are not registered in DNS.

-SuffixOrigin <SuffixOrigin[]>

Specifies an array of origins for address suffixes. IP addresses are divided into two parts, the prefix and the suffix. The address prefix identifies the network portion of an IP address, and the address suffix identifies the host portion. The acceptable values for this parameter are:

-- Manual. The IP address suffix was manually specified.

-- WellKnown. The IP address suffix is from a well-known source.

-- DHCP. The IP address suffix was provided by DHCP settings.

-- Link. The IP address suffix was obtained from the link-layer address.

-- Random. The IP address suffix was obtained from a random source.

-- Other. The IP address suffix was obtained from another source, such as a VPN.

-ThrottleLimit <Int32>

Specifies the maximum number of concurrent operations that can be established to run the cmdlet. If this parameter is omitted or a value of `0` is entered, then Windows PowerShell® calculates an optimum throttle limit for the cmdlet based on the number of CIM cmdlets that are running on the computer. The throttle limit applies only to the current cmdlet, not to the session or to the computer.

-Type <Type[]>

Specifies an array of IP address types. The acceptable values for this parameter are:

-- Unicast

-- Anycast

The default value is Unicast.

-ValidLifetime <TimeSpan[]>

Specifies an array of values for valid lifetimes, as TimeSpan objects, for an IP address. To obtain a TimeSpan object, use the New-TimeSpan cmdlet.

<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>).

----- Example 1: Get IPv6 address configuration -----

```
PS C:\>Get-NetIPAddress -AddressFamily IPv6
```

This command gets information about IP address configuration for all IPv6 addresses on the computer.

- Example 2: Get IP address information and format the output -

```
PS C:\>Get-NetIPAddress | Format-Table
```

This command gets information about IP address configuration, and displays that information in a table. The table format provides a convenient overview for computers with many IP addresses.

- Example 3: Get IP address information by the interface index -

```
PS C:\>Get-NetIPAddress -InterfaceIndex 12
```

This command gets information about IP address configuration for a specific interface index.

-- Example 4: Get IP address information and sort the output --

```
PS C:\>Get-NetIPAddress | Sort-Object -Property InterfaceIndex | Format-Table
```

This command gets information about IP address configuration, sorts them numerically by the interface index in the cmdlet name, and then displays them in a table format. This display can help you find IP address information by interface index.

- Example 5: Get IP address information and filter the output -

```
PS C:\>Get-NetIPAddress | Where-Object -FilterScript { $_.ValidLifetime -Lt  
([TimeSpan]::FromDays(1)) }
```

Similarly, this cmdlet can be used to get information about IP address configuration for IP addresses that have an Infinite ValidLifetime.

```
PS C:\>Get-NetIPAddress | Where-Object -FilterScript { $_.ValidLifetime -Eq  
([TimeSpan]::MaxValue) }
```

This command gets IP address configuration for IP addresses that have a ValidLifetime of less than one day.

REMARKS

To see the examples, type: "get-help Get-NetIPAddress -examples".

For more information, type: "get-help Get-NetIPAddress -detailed".

For technical information, type: "get-help Get-NetIPAddress -full".

For online help, type: "get-help Get-NetIPAddress -online"