MyWebUniversity

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-EventLog'

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

```
NAME
    Get-EventLog

SYNOPSIS
    Gets the events in an event log, or a list of the event logs, on the local
    computer or remote computers.


SYNTAX
    Get-EventLog [-LogName] <System.String> [[-InstanceId] <System.Int64[]>]
    [-After <System.DateTime>] [-AsBaseObject] [-Before <System.DateTime>]
    [-ComputerName <System.String[]>] [-EntryType {Error | Information |
    FailureAudit | SuccessAudit | Warning}] [-Index <System.Int32[]>] [-Message
    <System.String>] [-Newest <System.Int32>] [-Source <System.String[]>]
    [-UserName <System.String[]>] [<CommonParameters>]

    Get-EventLog [-AsString] [-ComputerName <System.String[]>] [-List]
    [<CommonParameters>]


DESCRIPTION
```

The `Get-EventLog` cmdlet gets events and event logs from local and remote computers. By default, `Get-EventLog` gets logs from the local computer. To get logs from remote computers, use the ComputerName parameter.

You can use the `Get-EventLog` parameters and property values to search for events. The cmdlet gets events that match the specified property values.

PowerShell cmdlets that contain the `EventLog` noun work only on Windows classic event logs such as Application, System, or Security. To get logs that use the Windows Event Log technology in Windows Vista and later Windows versions, use `Get-WinEvent`.

> [!NOTE] > `Get-EventLog` uses a Win32 API that is deprecated. The results may not be accurate. Use the > `Get-WinEvent` cmdlet instead.

PARAMETERS

-After <System.DateTime>

Gets events that occurred after a specified date and time. The After parameter date and time are excluded from the output. Enter a DateTime object, such as the value returned by the `Get-Date` cmdlet.

-AsBaseObject <System.Management.Automation.SwitchParameter>

Indicates that this cmdlet returns a standard System.Diagnostics.EventLogEntry object for each event. Without this parameter, `Get-EventLog` returns an extended PSObject object with additional EventLogName , Source , and InstanceId properties.

To see the effect of this parameter, pipe the events to the `Get-Member` cmdlet and examine the TypeName value in the result.

-AsString <System.Management.Automation.SwitchParameter>

Indicates that this cmdlet returns the output as strings, instead of objects.

-Before <System.DateTime>

Gets events that occurred before a specified date and time. The Before parameter date and time are excluded from the output. Enter a DateTime object, such as the value returned by the `Get-Date` cmdlet.

-ComputerName <System.String[]>

This parameter specifies a remote computer's NetBIOS name, Internet Protocol (IP) address, or a fully qualified domain name (FQDN).

If the ComputerName parameter isn't specified, `Get-EventLog` defaults to the local computer. The parameter also accepts a dot (`.`) to specify the local computer.

The ComputerName parameter doesn't rely on Windows PowerShell remoting. You can use `Get-EventLog` with the ComputerName parameter even if your computer is not configured to run remote commands.

-EntryType <System.String[]>

Specifies, as a string array, the entry type of the events that this cmdlet gets.

The acceptable values for this parameter are:

- Error

- Information

- FailureAudit

- SuccessAudit

- Warning

-Index <System.Int32[]>

Specifies the index values to get from the event log. The parameter accepts a comma-separated string of values.

-InstanceId <System.Int64[]>

Specifies the Instance IDs to get from the event log. The parameter accepts a comma-separated string of values.

-List <System.Management.Automation.SwitchParameter>

Displays the list of event logs on the computer.

-LogName <System.String>

Specifies the name of one event log. To find the log names use `Get-EventLog -List`. Wildcard characters are permitted. This parameter is required.

-Message <System.String>

Specifies a string in the event message. You can use this parameter to search for messages that contain certain words or phrases. Wildcards are permitted.

-Newest <System.Int32>

Begins with the newest events and gets the specified number of events. The number of events is required, for example `-Newest 100`. Specifies the maximum number of events that are returned.

-Source <System.String[]>

Specifies, as a string array, sources that were written to the log that this cmdlet gets. Wildcards are permitted.

-UserName <System.String[]>

Specifies, as a string array, user names that are associated with events.
Enter names or name patterns, such as `User01`, `User `, or `Domain01\User
`. Wildcards are permitted.

<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 event logs on the local computer -------

```
Get-EventLog -List
```

```
Max(K)   Retain   OverflowAction      Entries  Log
------   ------   --------------      -------  ---
15,168        0   OverwriteAsNeeded   20,792   Application
15,168        0   OverwriteAsNeeded   12,559   System
15,360        0   OverwriteAsNeeded   11,173   Windows PowerShell
```

The `Get-EventLog` cmdlet uses the List parameter to display the available
logs.

Example 2: Get recent entries from an event log on the local computer

```
Get-EventLog -LogName System -Newest 5
```

```
Index   Time          EntryType    Source              InstanceID   Message
-----   ----          ---------    ------              ----------   -------
13820   Jan 17 19:16  Error        DCOM                     10016   The
description for Event...
13819   Jan 17 19:08  Error        DCOM                     10016   The
description for Event...
13818   Jan 17 19:06  Information  Service Control...  1073748864   The start
```

```
type of the Back...
13817   Jan 17 19:05  Error        DCOM                    10016   The
description for Event...
13815   Jan 17 19:03  Information  Microsoft-Windows...        35   The time
service is now sync...
```

The `Get-EventLog` cmdlet uses the LogName parameter to specify the System event log. The Newest parameter returns the five most recent events.

Example 3: Find all sources for a specific number of entries in an event log

```
$Events = Get-EventLog -LogName System -Newest 1000
$Events | Group-Object -Property Source -NoElement | Sort-Object -Property
Count -Descending
```

```
Count   Name
-----   ----
  110   DCOM
   65   Service Control Manager
   51   Microsoft-Windows-Kern...
   14   EventLog
   14   BTHUSB
   13   Win32k
```

The `Get-EventLog` cmdlet uses the LogName parameter to specify the System log. The Newest parameter selects the 1000 most recent events. The event objects are stored in the `$Events` variable. The `$Events` objects are sent down the pipeline to the `Group-Object` cmdlet. `Group-Object` uses the Property parameter to group the objects by source and counts the number of objects for each source. The NoElement parameter removes the group members from the output. The `Sort-Object` cmdlet uses the Property parameter to sort by the count of each source name. The Descending parameter sorts the list in order by count from highest to lowest.

---- Example 4: Get error events from a specific event log ----

```
Get-EventLog -LogName System -EntryType Error
```

```
Index Time          EntryType   Source  InstanceID Message
----- ----          ---------   ------  ---------- -------
13296 Jan 16 13:53  Error       DCOM    10016 The description for Event ID
'10016' in Source...
13291 Jan 16 13:51  Error       DCOM    10016 The description for Event ID
'10016' in Source...
13245 Jan 16 11:45  Error       DCOM    10016 The description for Event ID
'10016' in Source...
13230 Jan 16 11:07  Error       DCOM    10016 The description for Event ID
'10016' in Source...
```

The `Get-EventLog` cmdlet uses the LogName parameter to specify the System log. The EntryType parameter filters the events to show only Error events.

Example 5: Get events from an event log with an InstanceId and Source value

```
Get-EventLog -LogName System -InstanceId 10016 -Source DCOM
```

```
Index Time          EntryType  Source  InstanceID  Message
----- ----          ---------  ------  ----------  -------
13245 Jan 16 11:45  Error      DCOM         10016  The description for Event
ID '10016' in Source...
13230 Jan 16 11:07  Error      DCOM         10016  The description for Event
ID '10016' in Source...
13219 Jan 16 10:00  Error      DCOM         10016  The description for Event
ID '10016' in Source...
```

The `Get-EventLog` cmdlet uses the LogName parameter to specify the System log. The InstanceID parameter selects the events with the specified Instance ID. The Source parameter specifies the event property.

-------- Example 6: Get events from multiple computers --------

```
Get-EventLog -LogName System -ComputerName Server01, Server02, Server03
```

The `Get-EventLog` cmdlet uses the LogName parameter to specify the System log. The ComputerName parameter uses a comma-separated string to list the computers from which you want to get the event logs.

Example 7: Get all events that include a specific word in the message

```
Get-EventLog -LogName System -Message *description*
```

```
Index Time          EntryType   Source      InstanceID   Message
----- ----          ---------   ------      ----------   -------
13821 Jan 17 19:17  Error       DCOM             10016   The description for
Event ID '10016'...
13820 Jan 17 19:16  Error       DCOM             10016   The description for
Event ID '10016'...
13819 Jan 17 19:08  Error       DCOM             10016   The description for
Event ID '10016'...
```

The `Get-EventLog` cmdlet uses the LogName parameter to specify the System event log. The Message parameter specifies a word to search for in the message field of each event.

------ Example 8: Display the property values of an event ------

```
$A = Get-EventLog -LogName System -Newest 1
$A | Select-Object -Property *
```

```
EventID            : 10016
MachineName        : localhost
Data               : {}
Index              : 13821
Category           : (0)
CategoryNumber     : 0
```

```
EntryType          : Error
Message            : The description for Event ID '10016' in Source 'DCOM'...
Source             : DCOM
ReplacementStrings : {Local,...}
InstanceId         : 10016
TimeGenerated      : 1/17/2019 19:17:23
TimeWritten        : 1/17/2019 19:17:23
UserName           : username
Site               :
Container          :
```

The `Get-EventLog` cmdlet uses the LogName parameter to specify the System event log. The Newest parameter selects the most recent event object. The object is stored in the `$A` variable. The object in the `$A` variable is sent down the pipeline to the `Select-Object` cmdlet. `Select-Object` uses the Property parameter with an asterisk (`*`) to select all of the object's properties.

Example 9: Get events from an event log using a source and event ID

```
Get-EventLog -LogName Application -Source Outlook | Where-Object {$_.EventID
-eq 63} |
            Select-Object -Property Source, EventID, InstanceId, Message
```

```
Source   EventID   InstanceId   Message
------   -------   ----------   -------
Outlook       63   1073741887   The Exchange web service request succeeded.
Outlook       63   1073741887   Outlook detected a change notification.
Outlook       63   1073741887   The Exchange web service request succeeded.
```

The `Get-EventLog` cmdlet uses the LogName parameter to specify the Application event log. The Source parameter specifies the application name, Outlook. The objects are sent down the pipeline to the `Where-Object` cmdlet. For each object in the pipeline, the `Where-Object` cmdlet uses the variable

`$_.EventID` to compare the Event ID property to the specified value. The objects are sent down the pipeline to the `Select-Object` cmdlet. `Select-Object` uses the Property parameter to select the properties to display in the PowerShell console.

-------- Example 10: Get events and group by a property --------

```
Get-EventLog -LogName System -UserName NT* | Group-Object -Property UserName
-NoElement |
             Select-Object -Property Count, Name

Count  Name
-----  ----
6031   NT AUTHORITY\SYSTEM
  42   NT AUTHORITY\LOCAL SERVICE
   4   NT AUTHORITY\NETWORK SERVICE
```

The `Get-EventLog` cmdlet uses the LogName parameter to specify the System log. The UserName parameter includes the asterisk (`*`) wildcard to specify a portion of the user name. The event objects are sent down the pipeline to the `Group-Object` cmdlet. `Group-Object` uses the Property parameter to specify that the UserName property is used to group the objects and count the number of objects for each user name. The NoElement parameter removes the group members from the output. The objects are sent down the pipeline to the `Select-Object` cmdlet. `Select-Object` uses the Property parameter to select the properties to display in the PowerShell console.

Example 11: Get events that occurred during a specific date and time range

```
$Begin = Get-Date -Date '1/17/2019 08:00:00'
$End = Get-Date -Date '1/17/2019 17:00:00'
Get-EventLog -LogName System -EntryType Error -After $Begin -Before $End

Index Time          EntryType   Source   InstanceID  Message
----- ----          ---------   ------   ----------  -------
```

```
13821 Jan 17 13:40  Error       DCOM          10016  The description for Event
ID...
13820 Jan 17 13:11  Error       DCOM          10016  The description for Event
ID...
...
12372 Jan 17 10:08  Error       DCOM          10016  The description for Event
ID...
12371 Jan 17 09:04  Error       DCOM          10016  The description for Event
ID...
```

The `Get-Date` cmdlet uses the Date parameter to specify a date and time. The DateTime objects are stored in the `$Begin` and `$End` variables. The `Get-EventLog` cmdlet uses the LogName parameter to specify the System log. The EntryType parameter specifies the Error event type. The date and time range is set by the After parameter and `$Begin` variable and the Before parameter and `$End` variable.

REMARKS

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

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

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

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