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

Windows PowerShell Get-Help on Cmdlet 'Get-ExecutionPolicy'

PS:\>Get-HELP Get-ExecutionPolicy -Full

NAME

Get-ExecutionPolicy

SYNOPSIS

Gets the execution policies for the current session.

SYNTAX

Get-ExecutionPolicy [[-Scope] {CurrentUser | LocalMachine | MachinePolicy | Process | UserPolicy}] [-List] [<CommonParameters>]

DESCRIPTION

To display the execution policies for each scope in the order of precedence, use `Get-ExecutionPolicy -List`. To see the effective execution policy for your

PowerShell session use `Get-ExecutionPolicy` with no parameters.

The effective execution policy is determined by execution policies that are set by `Set-ExecutionPolicy` and Group Policy settings.

For more information, see about_Execution_Policies (../Microsoft.PowerShell.Core/about/about_Execution_Policies.md).

PARAMETERS

-List <System.Management.Automation.SwitchParameter>

Gets all execution policy values for the session. By default, `Get-ExecutionPolicy` gets only the effective execution policy.

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

-Scope <Microsoft.PowerShell.ExecutionPolicyScope>

Specifies the scope that is affected by an execution policy.

The effective execution policy is determined by the order of precedence as follows:

- `MachinePolicy`. Set by a Group Policy for all users of the computer.

- `UserPolicy`. Set by a Group Policy for the current user of the computer.

- `Process`. Affects only the current PowerShell session.

- `LocalMachine`. Default scope that affects all users of the computer.

- `CurrentUser`. Affects only the current user.

Required?	false
Position?	0
Default value	Effective execution policy
Accept pipeline input?	True (ByPropertyName)

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

None

You can't pipe objects to this cmdlet.

OUTPUTS

Microsoft.PowerShell.ExecutionPolicy

This cmdlet returns the current execution policy.

NOTES

An execution policy is part of the PowerShell security strategy. Execution policies determine whether you can load configuration files, such as your PowerShell
profile, or run scripts. And, whether scripts must be digitally signed before they are run.

------------ Example 1: Get all execution policies ------------

```
Get-ExecutionPolicy -List
```

```
Scope          ExecutionPolicy
-----          ---------------
MachinePolicy  Undefined
UserPolicy     Undefined
```

```
Process        Undefined
CurrentUser    AllSigned
LocalMachine   Undefined
```

The `Get-ExecutionPolicy` cmdlet uses the List parameter to display each scope's execution policy.

-------------- Example 2: Set an execution policy --------------

```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope LocalMachine
Get-ExecutionPolicy -List

Scope ExecutionPolicy
     ----- ---------------
MachinePolicy       Undefined
  UserPolicy       Undefined
    Process       Undefined
 CurrentUser       AllSigned
 LocalMachine    RemoteSigned
```

The `Set-ExecutionPolicy` cmdlet uses the ExecutionPolicy parameter to specify the `RemoteSigned` policy. The Scope parameter specifies the default scope value,

`LocalMachine`. To view the execution policy settings, use the `Get-ExecutionPolicy` cmdlet with the List parameter.

-------- Example 3: Get the effective execution policy --------

```
PS> Get-ExecutionPolicy -List


     Scope ExecutionPolicy
     ----- ---------------
MachinePolicy       Undefined
  UserPolicy       Undefined
    Process       Undefined
 CurrentUser       AllSigned
 LocalMachine    RemoteSigned
```

```
PS> Get-ExecutionPolicy

AllSigned
```

The `Get-ExecutionPolicy` cmdlet uses the List parameter to display each scope's execution policy. The `Get-ExecutionPolicy` cmdlet is run without a parameter to display the effective execution policy, `AllSigned`.

Example 4: Unblock a script to run it without changing the execution policy

```
PS> Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope LocalMachine

PS> Get-ExecutionPolicy

RemoteSigned

PS> .\Start-ActivityTracker.ps1

.\Start-ActivityTracker.ps1 : File .\Start-ActivityTracker.ps1 cannot be loaded.
The file .\Start-ActivityTracker.ps1 is not digitally signed.
The script will not execute on the system.
For more information, see about_Execution_Policies at https://go.microsoft.com/fwlink/?LinkID=135170.
At line:1 char:1
+ .\Start-ActivityTracker.ps1
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
+ CategoryInfo          : NotSpecified: (:) [], PSSecurityException
+ FullyQualifiedErrorId : UnauthorizedAccess

PS> Unblock-File -Path .\Start-ActivityTracker.ps1

PS> Get-ExecutionPolicy

RemoteSigned
```

```
PS> .\Start-ActivityTracker.ps1
```

Task 1:

The `Set-ExecutionPolicy` uses the ExecutionPolicy parameter to specify the `RemoteSigned` policy. The policy is set for the default scope, `LocalMachine`.

The `Get-ExecutionPolicy` cmdlet shows that `RemoteSigned` is the effective execution policy for the current PowerShell session.

The `Start-ActivityTracker.ps1` script is executed from the current directory. The script is blocked by `RemoteSigned` because the script isn't digitally signed.

For this example, the script's code was reviewed and verified as safe to run. The `Unblock-File` cmdlet uses the Path parameter to unblock the script.

To verify that `Unblock-File` didn't change the execution policy, `Get-ExecutionPolicy` displays the effective execution policy, `RemoteSigned`.

The script, `Start-ActivityTracker.ps1` is executed from the current directory. The script begins to run because it was unblocked by the `Unblock-File` cmdlet.

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/microsoft.powershell.security/get-executionpolicy?view=powershell-5.1&WT.mc_id=ps-gethelp

about_Execution_Policies

about_Group_Policy_Settings

Get-AuthenticodeSignature

Set-AuthenticodeSignature

Set-ExecutionPolicy