



News & Insights

Community

Research

IT Tools

Newsletters

Events

SpiceWorld

Log In | Join



- Daily Challenge
- Unanswered Questions
- Community Guidelines
- Events
- More

Categories

- Cloud Computing & ...
- Collaboration
- Data Storage, Backu...
- Hardware
- Networking
- Programming & Dev...
- Security
- Software
- Vendors
- Virtualization
- Windows

All categories

Tags

- active-directory-gpo
- dns
- firewalls
- microsoft-office-365
- powershell

Skip to main content



Spiceworks Community > Security

Batch Script to uninstall Symantec Endpoint Protection

Security

general-windows

powershell

question

symantec



aceelsheik Poblano

Feb 2019

Hi All,

I am trying to create a batch script to uninstall Symantec. I've created a rough draft, but was wondering on how I could make it better. The uninstall script that I currently have is as follows:

```
@echo OFF
call MsiExec.exe /x {4C89867B-2E80-4B0D-87DB-1BD643D5EF5D}
taskkill /IM PAUI.exe
timeout 10
taskkill /IM ccSvcHst.exe /F
timeout 10
call "C:\Program Files\Symantec.cloud\PlatformAgent\Uninstall\Uninstall.bat"
exit /B 3010
```

Is there a way to check if the product code is installed first before initiating the command

For example:

```
REM IF (PRODUCT CODE Exists)
{
  REM uninstall product code
  Continue to uninstall (call "C:\Program Files\Symantec.cloud\PlatformAgent\Uninstall\Uninstall.bat")
}
else
```

1 / 6

```
{  
Continue to uninstall (call "C:\Program Files\Symantec.cloud
```

Or to check if the service is running. And the return the exit code, and not a static one (3010)

Any help would be greatly appreciated

Regards,

Ace.

2  

1.2k 2
views links



DoctorDNS  Pure Capsaicin

Feb 2019

Thanks for posting this to the PowerShell forum - except this is not PowerShell?!

Another way to do it is;

```
(Get-WmiObject -Class Win32_Product -Filter "Name='Symantec
```

 



Evan7191 Ghost Chili

Feb 2019

You can use an If statement to check if the product code exists in the registry and then run the msixec command to uninstall it.

 

[Skip to main content](#)



Evan7191 Ghost Chili

1 / 6

Yes, the Win32_Product class has methods for removing software, but it is inefficient.

 Scripting Blog [archived] – 13 Nov 11 [↗](#)



Use PowerShell to Quickly Find Installed Software - Scripting Blog [archived] 4 [↗](#)

Summary: Learn how to use Windows PowerShell to quickly find installed software on local and remote computers. Microsoft Scripting Guy Ed Wilson here. Guest Blogger Weekend concludes with Marc Carter.

The Scripting Wife and I were lucky enough...

 Scripting Blog [archived] – 14 Dec 11 [↗](#)



Use PowerShell to Find and Uninstall Software - Scripting Blog [archived] 4 [↗](#)

Summary: Learn how to use Windows PowerShell to get software installation locations, and to uninstall software from remote computers. Hey, Scripting Guy! We have a dumb application that we have to use at

work. The company has released a new version...



DoctorDNS  Pure Capsaicin

Feb 2019

Sure - it is inefficient. But it works, it's simple to implement and if perf is an issue, run it overnight!



[Skip to main content](#)

1 / 6



aceelsheik Poblano

Author

Mar 2019

Hi All,

Thanks for the replies. So I've run into another hiccup. After uninstalling the product, it for some reason is downloading an upgrade package. Upon further troubleshooting, the files gets downloaded to "C:\Users\USER\AppData\Local\Temp\{GUID}" folder and amongst those files, lies FSDUI.exe which when launched, comes up with a norton download manager prompt

I have then created a powershell script and have deployed it via SCCM to all users.

The script is as follows:

```
cls
# if process is not running
if((Get-Process "FSDUI" -ErrorAction SilentlyContinue) -eq $null)
{
    $Folder = (Get-ChildItem -Path "$env:userprofile\AppData\Local\Temp\{GUID}")
    if ($Folder -eq $null)
    {
        #Write-Host "Folder was empty and process (FSDUI) was not running"
        #exit
    }
    else
    {
        #Write-Host "Folder exists and process (FSDUI) was running"
        Remove-Item $Folder -Force -Recurse -ErrorAction SilentlyContinue
    }
}
```

[Skip to main content](#)

1 / 6

```
    }  
  
    else  
    #if process is running  
    ,
```

I have then set the detection method to detect based on another powershell script as well:

```
[String]$LogfileName = "FSDUI_Detection"  
[String]$Logfile = "$env:SystemRoot\Temp\$LogfileName.log"  
Function Write-Log  
{  
    Param ([string]$logstring)  
    If (Test-Path $Logfile)  
    {  
        If ((Get-Item $Logfile).Length -gt 2MB)  
        {  
            Rename-Item $Logfile $Logfile".bak" -Force  
        }  
    }  
    $WriteLine = (Get-Date).ToString() + " " + $logstring  
    Add-content $Logfile -value $WriteLine  
}  
  
$User = gwmi win32_computersystem -Property Username  
$UserName = $User.UserName  
$Folder = (Get-ChildItem -Path "$env:userprofile\AppData\Loc  
# Parameter to Log  
Write-Log "Start Script Execution"
```

But the script runs indefinitely everytime the sccm action cycles run, is there a way to stop the script after running once

Regards,

Ace.

 Reply

Do you know the answer to today's question? 🏆

What is the decimal number 25 in Octal?

☐ 31

☐ 19

☐ 00011001

☐ 10103072

☐ I'm not sure, but I'm excited to learn more!

Submit

Skip to main content

Related Topics

1 / 6

1 / 6

- FAQs
- Support Forum
- Community Guidelines

Resources

- All Categories
- Tech How-Tos
- Scripts
- IT Research
- Product Reviews

For Tech Marketers

- Marketing Services
- Demand Generation
- Account Based Marketing
- Data Capabilities
- Contact Sales

Spiceworks

- About Us
- Contact Us
- News & Insights
- Community
- Tools & Apps
- Aberdeen
- Careers

Skip to main content



[Sitemap](#) • [Privacy Policy](#) • [Terms of Use](#) • [Cookie Policy](#) • [Accessibility Statement](#) • [Do Not Sell My Personal Information](#)

Copyright © 2006 – 2024 Spiceworks Inc.