Product Solutions Resources Open Source Enterprise Pricing Sign in Sign up

rootm0s / **WinPwnage** Public

Notifications Fork 381 Star 2.6k

Code Issues 4 Pull requests 1 Actions Projects Security Insights

master Go to file Code

714 Commits

winpwnage
README.md
main.py

README

WINPWNAGE
Windows penetration techniques

build_status Python 3

- Build into single executable
- Scan for compatible methods
- Importing and usage as module
- UAC-bypass techniques
- Persistence techniques
- Elevation techniques

Disclaimer

This tool is provided for educational and research purposes only. The authors of this project are no way responsible for any misuse of this tool.

Building

This build works on Python >= 3.6 and puts the .exe file into the **dist** directory. Install pyinstaller using pip command:

```
pip install pyinstaller
```

And run the following command:

```
pyinstaller --onefile main.py
```

Scanning

About

UAC bypass, Elevate, Persistence methods

Readme
Activity
2.6k stars
107 watching
381 forks
Report repository

Releases

No releases published

Packages

No packages published

Contributors 5

Languages

Python 100.0%

Compares build number against 'Fixed In' build numbers and displays the results.

```
main.py --scan uac
main.py --scan persist
main.py --scan elevate
```

Example results when scanning for possible UAC methods

```
 Id:    Type:           Compatible:     Description:
 ----   ------          -----------     -------------
 1      UAC bypass      No              UAC bypass using runas
 2      UAC bypass      Yes             UAC bypass using fodhelper.e
 3      UAC bypass      Yes             UAC bypass using slui.exe
 4      UAC bypass      Yes             UAC bypass using silentclean
 5      UAC bypass      No              UAC bypass using sdclt.exe (
 6      UAC bypass      No              UAC bypass using sdclt.exe (
 7      UAC bypass      No              UAC bypass using perfmon.exe
```

Importing

Bypass UAC using uacMethod2

```
from winpwnage.functions.uac.uacMethod2 import uacMethod2
uacMethod2(["c:\\windows\\system32\\cmd.exe", "/k", "whoami"])
```

Persist on system using persistMethod4

```
from winpwnage.functions.persist.persistMethod4 import persistMethod
persistMethod4(["c:\\windows\\system32\\cmd.exe", "/k", "whoami"], a

# Removal
persistMethod4(["c:\\windows\\system32\\cmd.exe", "/k", "whoami"], a
```

Elevate from administrator to SYSTEM using elevateMethod1

```
from winpwnage.functions.elevate.elevateMethod1 import elevateMethod
elevateMethod1(["c:\\windows\\system32\\cmd.exe", "/k", "whoami"])
```

UAC bypass techniques

▶ Functions (Expand/Collapse)

Persistence techniques

▶ Functions (Expand/Collapse)

Elevation techniques

▶ Functions (Expand/Collapse)

Read

- https://wikileaks.org/ciav7p1/cms/page_2621770.html
- https://wikileaks.org/ciav7p1/cms/page_2621767.html
- https://wikileaks.org/ciav7p1/cms/page_2621760.html
- https://msdn.microsoft.com/en-us/library/windows/desktop/bb736357(v=vs.85).aspx
- https://winscripting.blog/2017/05/12/first-entry-welcome-and-uac-bypass/
- https://github.com/winscripting/UAC-bypass/

- https://www.greyhathacker.net/?p=796
- https://github.com/hfiref0x/UACME
- https://bytecode77.com/hacking/exploits/uac-bypass/performance-monitor-privilege-escalation
- https://bytecode77.com/hacking/exploits/uac-bypass/slui-file-handler-hijack-privilege-escalation
- https://media.defcon.org/DEF%20CON%2025/DEF%20CON%2025%20workshops/DEFCON-25-Workshop-Ruben-Boobeb-UAC-0day-All-Day.pdf
- https://lolbas-project.github.io

© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information