

Product ▾

Solutions ▾

Resources ▾

Open Source ▾

Enterprise ▾

Pricing

🔍

Sign in

Sign up

📄

Neo23x0 / DLLRunner

Public

🔔

Notifications

🍴

Fork

44

★

Star

141

<>

Code

🔗

Issues

🔗

Pull requests

🎬

Actions

📁

Projects

📖

Wiki

🛡️

Security

📈

Insights

🔗 master ▾

🔗

📁


🔍

Go to file

<>

Code

▾



Neo23x0

Compiled EXE

49cbf83 · 10 years ago

🕒 12 Commits

📄 README.md	Update README.md	10 years ago
📄 dllrunner.exe	Compiled EXE	10 years ago
📄 dllrunner.py	Added fuzzing params and debug outp...	10 years ago

📖 README

⋮

DLLRunner

DLLRunner is a smart DLL execution script for malware analysis in sandbox systems.

Instead of executing a DLL file via "rundll32.exe file.dll" it analyzes the PE and executes all exported functions by name or ordinal in order to determine if one of the functions causes malicious activity.

```
rundll32.exe path/to/file.dll,exportedfunc1
rundll32.exe path/to/file.dll,exportedfunc2
rundll32.exe path/to/file.dll,exportedfunc3
```

Furthermore it tries to fuzz parameters in order to trigger acitivity in functions that require parameters to work.

```
rundll32.exe path/to/file.dll,exportedfunc1 "0"
rundll32.exe path/to/file.dll,exportedfunc1 "1"
rundll32.exe path/to/file.dll,exportedfunc1 "http://evil.local"
rundll32.exe path/to/file.dll,exportedfunc1 "Install"
...
```

Usage

```
usage: dllrunner.py [-h] [-f dllfile] [-l limit] [--fuzz] [--demo] [-
```

DLLRunner

optional arguments:

-h, --help

show this help message and exit

-f dllfile

DLL file to execute exported functions

-l limit

Only perform extended calls if export function count is less than limit

--fuzz

Add fuzzing parameters to the functions calls (currently no params are defined)

--demo

Run a demo using \system32\url.dll

--debug

Debug output

About

Smart DLL execution for malware analysis in sandbox systems

📖

Readme

📈

Activity

★

141 stars

👁

14 watching

🍴

44 forks

Report repository

Releases

No releases published

Packages

No packages published

Languages

Python

100.0%

Page 1 of 2

