Product Solutions Resources Open Source Enterprise Pricing Sign in Sign up

audibleblink / **xordump** Public

Notifications | Fork 2 | Star 18

Code | Pull requests | Actions | Security | Insights

main | Go to file | Code

audibleblink adds pid flag 8857597 · 3 years ago 6 Commits

.github/workflows	initial commit	3 years ago
.gitignore	initial commit	3 years ago
Makefile	initial commit	3 years ago
go.mod	initial commit	3 years ago
go.sum	initial commit	3 years ago
main.go	adds cmdline flag for pid	3 years ago
readme.md	adds pid flag	3 years ago

README

xordump

Made for use with Atomic Red Team.

```
Usage of xordump.exe:
  -in string
        Input file to Xor
  -m string
        [ dbghelp | dbgcore | comsvcs ] (default "dbghelp")
  -out string
        minidump outfile (default "minidump.dmp")
  -process string
        Process to dump (default "lsass.exe")
  -pid int
        PID of process. Takes precedence over --process
  -x int
        Single Byte Xor Key
```

In some cases, lsass.exe minidump files are signatured by AV and deleted. It's not unusual for the binary that initiated the lsass dump to be left on disk and not treated as malicious.

The dll loaded into this bin for minidumping (dgbhelp) *ALWAYS* writes the minidump to disk, but before this binary closes the file handle, it re-reads the contents into memory, closes the handle and immediately deletes the file.

There may exist a race between Go deleting the minidump file after a `close(handle)` and with AV detecting and deleting the file. In either case, the output is safe in memory and passed to a Xor function which then re-writes the xor'd data to disk, where it can be safely exfilled.

OPSEC consideration If you lose the race, AV may see the dumpfile and say something.

Part of the miniDump and seDebug code written by @C-Sto

About

No description, website, or topics provided.

Readme
Activity
18 stars
5 watching
2 forks
Report repository

Releases 2

Release v0.0.2 Latest
on May 25, 2021

+ 1 release

Packages

No packages published

Languages

Go 92.3% Makefile 7.7%

Building / Usage

Running `make` should build the windows exe and the shellcode that can be injected. Go 1.16+ is required because of the additional PIE (position independence) flag introduced in 1.15 and `ioutil` deprecation in 1.16.