









 last-byte	Update README.md [skip ci]	ac4751a · 4 months ago	 256 Commits
 .github/workflows	Update main.yml		2 years ago
 PersistenceSniper	update 1.16.1		5 months ago
 .gitignore	updated to v1.15.0		10 months ago
 CHANGELOG.md	update 1.16.1		5 months ago
 LICENSE	updated to 1.9.3		last year
 README.md	Update README.md [skip ci]		4 months ago
 SECURITY.md	Create SECURITY.md [skip ci]		2 years ago
 persistencesnipernew4.png	Add files via upload		5 months ago

READMELicenseSecurity

PersistenceSniper



Language Powershell

Module Version v1.16.1

Persistence Techniques 56

Digital Signature Valid

Gallery Downloads 38k

Follow @PersistSniper

Follow @last0x00

Follow @dottor_morte

buy me a coffee

PersistenceSniper is a Powershell module that can be used by Blue Teams, Incident Responders and System Administrators to hunt persistences implanted in Windows machines. It is also available on [Powershell Gallery](#) and it is digitally signed with a valid code signing certificate. The tool is under active development with new releases coming out by the week, so make sure to use the up-to-date version.

Official Twitter/X account [@PersistSniper](#).


About

Powershell module that can be used by Blue Teams, Incident Responders and System Administrators to hunt persistences implanted in Windows machines. Official Twitter/X account @PersistSniper. Made with ❤️ by @last0x00 and @dottor_morte

- windows
- registry
- powershell
- persistence
- incident-response
- powershell-script
- techniques
- powershell-module
- malware-detection

- Readme
- View license
- Security policy
- Activity
- 1.9k stars
- 42 watching
- 184 forks
- Report repository





Releases 22

 PersistenceSniper v1.16.1 Latest

on Jun 30

+ 21 releases

Contributors 4

-  last-byte last - @last0x00
-  d3f0x0 Vadim
-  fkadibs fka dibs
-  suinswofi Consty

Languages



The Why

Why writing such a tool, you might ask. Well, for starters, I tried looking around and I did not find a tool which suited my particular use case, which was looking for known

persistence techniques, automatically, across multiple machines, while also being able to quickly and easily parse and compare results. Sure, [Sysinternals' Autoruns](#) is an amazing tool and it's definitely worth using, but, given it outputs results in non-standard formats and can't be run remotely unless you do some shenanigans with its command line equivalent, I did not find it a good fit for me. Plus, some of the techniques I implemented so far in PersistenceSniper have not been implemented into Autoruns yet, as far as I know. Anyway, if what you need is an easy to use, GUI based tool with lots of already implemented features, Autoruns is the way to go, otherwise let PersistenceSniper have a shot, it won't miss it 😊

The How

To learn how to use PersistenceSniper properly, head to the [Project's Wiki](#).

TL;DR If you are too lazy to read the [Wiki](#) (which I highly recommend you do) you can install, import, and fire PersistenceSniper with the following three commands.

```
PS> Install-Module PersistenceSniper
PS> Import-Module PersistenceSniper
PS> Find-AllPersistence
```

Persistence techniques implemented so far

The persistence techniques implemented so far are detailed in the [Detections Page](#) of PersistenceSniper's Wiki.

Credits

Most of this tool is based on the work of other skilled researchers, so it's right to give credit where credit's due. This project wouldn't be around if it weren't for:

- [Hexacorn](#) and his never-ending [Beyond good ol' Run key series](#);
- [Grzegorz Tworek](#) and his amazing [persistence-info.github.io website](#);
- All the other researchers who disclosed cool and unknown persistence techniques.

Furthermore, these people contributed to the project:

- [Riccardo Ancarani](#)
- [Cecio](#)
- [Vadim](#)
- [fkadibs](#)
- [suinswofi](#)
- [Antonio Blescia](#)
- [Strassi](#)

I'd also like to give credits to my fellow mates at [@APTortellini](#) for the flood of ideas that helped it grow from a puny text-oriented script to a full-fledged Powershell module.

License

This project is under the [Commons Clause version of the MIT License](#) license. TL;DR: you can copy, modify, distribute and perform the work for whatever reason, **excluding** commercial purposes, all without asking permission.