GitHub - EmpireProject/Empire: Empire is a PowerShell and Python post-exploitation agent.

By xorrior

Archived: 2026-04-05 20:57:05 UTC

This project is no longer supported

Empire is a post-exploitation framework that includes a pure-PowerShell2.0 Windows agent, and a pure Python 2.6/2.7 Linux/OS X agent. It is the merge of the previous PowerShell Empire and Python EmPyre projects. The framework offers cryptologically-secure communications and a flexible architecture. On the PowerShell side, Empire implements the ability to run PowerShell agents without needing powershell.exe, rapidly deployable post-exploitation modules ranging from key loggers to Mimikatz, and adaptable communications to evade network detection, all wrapped up in a usability-focused framework. PowerShell Empire premiered at BSidesLV in 2015 and Python EmPyre premeiered at HackMiami 2016.

Empire relies heavily on the work from several other projects for its underlying functionality. We have tried to call out a few of those people we've interacted with heavily here and have included author/reference link information in the source of each Empire module as appropriate. If we have failed to improperly cite existing or prior work, please let us know.

Empire is developed by @harmj0y, @sixdub, @enigma0x3, rvrsh3ll, @killswitch_gui, and @xorrior.

Feel free to join us on Slack! https://bloodhoundgang.herokuapp.com

Install

To install, run `sudo ./setup/install.sh` script or use the corresponding docker image `docker pull empireproject/empire`.

There's also a quickstart here and full documentation here.

Quickstart

Check out the Empire wiki for instructions on getting started with Empire.

Contribution Rules

Contributions are more than welcome! The more people who contribute to the project the better Empire will be for everyone. Below are a few guidelines for submitting contributions.

- Beginning with version 2.4, we will only troubleshoot issues for Kali, Debian, or Ubuntu. All other operating systems will not be supported. We understand that this is frustrating but hopefully the new

docker build can provide an alternative.

- Submit pull requests to the dev branch. After testing, changes will be merged to master.
- Depending on what you're working on, base your module on ./lib/modules/powershell_template.py or ./lib/modules/python_template.py. **Note** that for some modules you may need to massage the output to get it into a nicely displayable text format with Out-String.
- Cite previous work in the '**Comments**' module section.
- If your script.ps1 logic is large, may be reused by multiple modules, or is updated often, consider implementing the logic in the appropriate **data/module_source/*** directory and pulling the script contents into the module on tasking.
- Use approved PowerShell verbs for any functions.
- PowerShell Version 2 compatibility is **STRONGLY** preferred.
- TEST YOUR MODULE! Be sure to run it from an Empire agent before submitting a pull to ensure everything is working correctly.
- For additional guidelines for your PowerShell code itself, check out the PowerSploit style guide.
