scapy 2.6.0

```
pip install scapy
```

Latest version

Released: Sep 28, 2024

Scapy: interactive packet manipulation tool

Navigation

Project description

Release history

Download files

Verified details

These details have been verified by PyPI

Maintainers

gpotter2

guedou

phil

Project description

Scapy

pypi v2.6.0 License GPL v2

Scapy is a powerful Python-based interactive packet manipulation program and library.

It is able to forge or decode packets of a wide number of protocols, send them on the wire, capture them, store or read them using pcap files, match requests and replies, and much more. It is designed to allow fast packet prototyping by using default values that work.

It can easily handle most classical tasks like scanning, tracerouting, probing, unit tests, attacks or network discovery (it can replace `hping`, 85% of `nmap`, `arpspoof`, `arp-sk`, `arping`, `tcpdump`, `wireshark`, `p0f`, etc.). It also performs very well at a lot of other specific tasks that most other tools can't handle, like sending invalid frames, injecting your own 802.11 frames, combining

p-l

techniques (VLAN hopping+ARP cache poisoning, VoIP decoding on WEP protected channel, ...), etc.

Scapy supports Python 3.7+. It's intended to be cross platform, and runs on many different platforms (Linux, OSX, *BSD, and Windows).

Getting started

Scapy is usable either as a **shell** or as a **library**. For further details, please head over to Getting started with Scapy, which is part of the documentation.

Shell demo

Scapy can easily be used as an interactive shell to interact with the network. The following example shows how to send an ICMP Echo Request message to `github.com`, then display the reply source IP address:

```
sudo ./run_scapy
Welcome to Scapy
>>> p = IP(dst="github.com")/ICMP()
>>> r = sr1(p)
Begin emission:
```

Unverified details

*These details have **not** been verified by PyPI*

Project links

- Changelog
- Documentation
- Download
- Homepage
- Source Code

Meta

- **License:** GNU General Public License v2 (GPLv2) (GPL-2.0-only)
- **Author:** Philippe BIONDI
- **Maintainer:** Pierre LALET, Gabriel POTTER, Guillaume VALADON, Nils WEISS
- network
- **Requires:** Python <4, >=3.7
- **Provides-Extra:** `all`, `cli`, `doc`

Classifiers

Development Status

- 5 - Production/Stable

Environment

- Console

Intended Audience

- Developers
- Information Technology
- Science/Research
- System Administrators
- Telecommunications Industry

License

- OSI Approved :: GNU General Public License v2 (GPLv2)

Programming Language

- Python :: 3
- Python :: 3 :: Only
- Python :: 3.7
- Python :: 3.8
- Python :: 3.9
- Python :: 3.10
- Python :: 3.11
- Python :: 3.12
- Python :: 3.13

Topic

- Security
- System :: Networking
- System :: Networking :: Monitoring

```
.Finished to send 1 packets.
*
Received 2 packets, got 1 answers, remaining 0 packets
>>> r[IP].src
'192.30.253.113'
```

Resources

The documentation contains more advanced use cases, and examples.

Other useful resources:

- Scapy in 20 minutes
- Interactive tutorial (part of the documentation)
- The quick demo: an interactive session (some examples may be outdated)
- HTTP/2 notebook
- TLS notebooks

Installation

Scapy works without any external Python modules on Linux and BSD like operating systems. On Windows, you need to install some mandatory dependencies as described in the documentation.

On most systems, using Scapy is as simple as running the following commands:

```
git clone https://github.com/secdev/scapy
cd scapy
./run_scapy
```

To benefit from all Scapy features, such as plotting, you might want to install Python modules, such as `matplotlib` or `cryptography`. See the documentation and follow the instructions to install them.

Help

Installing packages
Uploading packages
User guide
Project name retention
FAQs

About PyPI

PyPI Blog
Infrastructure dashboard
Statistics
Logos & trademarks
Our sponsors

Contributing to PyPI

Bugs and feedback
Contribute on GitHub
Translate PyPI
Sponsor PyPI
Development credits

Using PyPI

Code of conduct
Report security issue
Privacy Notice
Terms of Use
Acceptable Use Policy

Status: all systems operational

Developed and maintained by the Python community, for the Python community.
Donate today!

Site map

Switch to desktop version

English español français 日本語 português (Brasil) українська Ελληνικά Deutsch 中文 (简体) 中文 (繁體) русский עברית Esperanto