Arbitrary code execution from pip's "—extra-index-url"

Blake Griffith — 2020-05-08 05:28 — 2 Comments

Once upon a time I made a python library, and gave it a very common name. Let's say the name was "`foo`". I also created a package that depended on `foo`, lets call it "`bar`". Both packages were in in my own private PyPI server, but something was wrong. When I installed `bar` the wrong dependency was being installed for `foo`. The package with the name `foo` from the public PyPI was being installed, not the one from my private PyPI.

I had inadvertently installed a package that could have run arbitrary code on my computer.

The problem occurs when `--extra-index-url` is used to point to a private PyPI that has packages with names shadowed on the public PyPI. When then this happens pip is needs to choose which PyPI to take the package from. It simply chooses the one with the higher version number.

This is a problem when: someone is using a private PyPI, with the `--extra-index-url`, and they are using a package on the private PyPI with a name they have not claimed on the public PyPI.

Going back to our original scenario, if an attacker controlled the name `foo` on the public PyPI, they could replace it with a malicious payload. The attacker could get control of the public `foo` name if it wasn't taken, or even if it was taken but unused (see PEP 541 (https://www.python.org/dev/peps/pep-0541/)).

I think **many people** who have private python packages are vulnerable to having their packages hijacked like this.

I went looking around the internet. Here is one vulnerable PyPI instance I found.

CERN's vulnerable PyPI

CERN hosts many of it's own python packages here (https://www.python.org/dev/peps/pep-0541/). Many of these packages names were not taken on the public PyPI, so I took them. Note that they also provide instructions (https://twiki.cern.ch/twiki/bin/view/LHCb/PythonPyPIServer) to their users that they should use `--extra-index-url`.

I notified CERN that I had taken the names for their packages on the public PyPI and transferred ownership to them.

They offered to give me a tour if I was ever in Geneva :)

Upstream disclosure

I disclosed this to the `security@python.org` list. Unfortunately they said there is currently no path to fix this.

They recommended using "version-pinning and hash-pinning for deployments" to avoid this.

They also discussed shadowing the names of your own private packages on the public PyPI. This has problems because it reveals the names of packages you use, and forces you to effectively squat those names.

CVE-2018-20225

I reported this as a CVE. The link is here (https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-20225).