Multiple CVEs (1 CRITICAL, 3 HIGH, 1 MODERATE) affecting the tarfile module

Seth Larson

June 3, 2025 1:01 p.m.

There are multiple advisories (1 CRITICAL, 3 HIGH, 1 MODERATE) affecting the CPython tarfile module.

Bypasses in tarfile extraction filtering

These three vulnerabilities are all different methods of bypassing tar extraction filtering which is a feature in Python 3.12 and later.

You are affected by this vulnerability if using the tarfile module to extract untrusted tar archives using TarFile.extractall() or TarFile.extract() using the filter= parameter with a value of "data" or "tar". See the tarfile extraction filters documentation for more information. Only Python versions 3.12 or later are affected by these vulnerabilities, earlier versions don't include the extraction filter feature.

Note that for Python 3.14 or later the default value of filter= changed from "no filtering" to `"data", so if you are relying on this new default behavior then your usage is also affected.

Note that none of these vulnerabilities significantly affect the installation of source distributions which are tar archives as source distributions already allow arbitrary code execution during the build process. However when evaluating source distributions it's important to avoid installing source distributions with suspicious links.

Note that when extracting an untrusted tar archive without extraction filtering enabled already allows for arbitrary write access outside of the extraction directory.

- CVE-2025-4517 <https://www.cve.org/CVERecord?id=CVE-2025-4517> (CRITICAL) allows arbitrary filesystem writes outside the extraction directory during extraction with filter="data".
- CVE-2025-4330 <https://www.cve.org/CVERecord?id=CVE-2025-4330> (HIGH) allows the extraction filter to be ignored, allowing symlink targets to point outside the destination directory, and the modification of some file metadata.

- CVE-2025-4138 <https://www.cve.org/CVERecord?id=CVE-2025-4138> (HIGH) allows creating arbitrary symlinks outside the extraction directory during extraction with filter="data".

- CVE-2024-12718 <https://www.cve.org/CVERecord?id=CVE-2024-12718> (MODERATE) allows modifying some file metadata (e.g. last modified) with `filter="data"` or file permissions (chmod) with `filter="tar"` of files outside the extraction directory.

Filtered members not skipped with TarFile.errorlevel = 0

When using a `TarFile.errorlevel = 0` and extracting with a filter the documented behavior is that any filtered members would be skipped and not extracted. However the actual behavior of `TarFile.errorlevel = 0` in affected versions is that the member would still be extracted and not skipped.

- CVE-2025-4435 <https://www.cve.org/CVERecord?id=CVE-2025-4435> (HIGH)

Mitigation

The recommended mitigation is to upgrade your Python version to a fixed version or apply the patch(es) linked within the CVEs. If you cannot patch or upgrade, rejecting all links with the parent directory segment (`".."`) prior to calling extract will mitigate the vulnerabilities below:

```
# Avoid insecure segments in link names.
for member in tar.getmembers():
    if not member.islnk():
        continue
    if os.pardir in os.path.split(member.linkname):
        raise OSError("Tarfile with insecure segment ('..') in linkname")

# Now safe to extract members with the data filter.
tar.extractall(filter="data")
```

Please see the linked CVE IDs for the latest information on affected versions.

Attachments:
attachment.html (text/html — 6.7 KB)

Download

2 comments
1 participants

Add to favorites

TAGS

PARTICIPANTS (1)

Seth Larson