December 8, 2021

GHSL-2021-125: Path traversal in SharpZipLib - CVE-2021-32840, CVE-2021-32841, CVE-2021-32842

Jaroslav Lobacevski

Coordinated Disclosure Timeline

- 2021-09-03: Report sent to sharpziplib@containrrr.dev according to security policy.
- 2021-09-08: No response. The email repeated.
- 2021-09-17: No response. Public issue created asking for the contact.
- 2021-09-17: Contact established.
- 2021-09-19: v1.3.3 with a fix was released.

Summary

SharpZipLib allows full or partial (depending on the version) traversal of the extraction path.

Product

SharpZipLib

Tested Version

0.86.0-1.3.2, however according to the SharpZipLib security policy version 0.86.0 is no longer supported.

Details

Issue 1: 0.86.0 <= SharpZipLib <= 1.2.0 TAR extraction doesn’t validate if the destination path is under the expected extraction directory

The following code example from SharpZipLib wiki:

```
using System;
using System.IO;
using ICSharpCode.SharpZipLib.Tar;

public void ExtractTar(String tarFileName, String destFolder)
{
    Stream inStream = File.OpenRead(tarFileName);

    TarArchive tarArchive = TarArchive.CreateInputTarArchive(inStream);
    tarArchive.ExtractContents(destFolder);
    tarArchive.Close();

    inStream.Close();
}
```

A TAR file entry `../evil.txt` will be extracted in the parent directory of `destFolder`.

Impact

It leads to arbitrary file write that may lead to code execution.

CVE

- CVE-2021-32840

Issue 2: 1.3.0 <= SharpZipLib <= 1.3.2 TAR extraction directory path is not enforced to be slash terminated

Starting versioin 1.3.0 a check was added if the destination file is under destination directory:

```
if (!allowParentTraversal && !Path.GetFullPath(destFile).StartsWith(destDir, StringComparison.InvariantCultureIgnoreCase))
{
        throw new InvalidNameException("Parent traversal in paths is not allowed");
}
```

However it is not enforced that `destDir` ends with slash. If the `destDir` is not slash terminated like `/home/user/dir` it is possible to create a file with a name thats begins with the destination directory, i.e. `/home/user/dir.sh`.

Impact

Because of the file name and destination directory constraints the arbitrary file creation impact is limited and depends on the use case.

CVE

- CVE-2021-32841

Issue 3: 0.86.0 <= SharpZipLib <= 1.3.1 ZIP extraction directory path is not enforced to be slash terminated

The following code example from SharpZipLib wiki:

```
using System;
using ICSharpCode.SharpZipLib.Zip;

public void TestFastZipUnpack(string zipFileName, string targetDir) {

    FastZip fastZip = new FastZip();
    string fileFilter = null;

    // Will always overwrite if target filenames already exist
    fastZip.ExtractZip(zipFileName, targetDir, fileFilter);
}
```

Starting version 1.0.0 a check was added if the destination file is under destination directory:

```
if (_baseDirectory != null) {
        name = Path.Combine(_baseDirectory, name);

        if(!_allowParentTraversal && !Path.GetFullPath(name).StartsWith(_baseDirectory, StringComparison.InvariantCultureIgnoreCase))
        {
                throw new InvalidNameException("Parent traversal in paths is not allowed");
        }
```

However it is not enforced that `_baseDirectory` ends with slash. If the `_baseDirectory` is not slash terminated like `/home/user/dir` it is possible to create a file with a name thats begins as the destination directory one level up from the directory, i.e. `/home/user/dir.sh`.

Impact

Because of the file name and destination directory constraints the arbitrary file creation impact is limited and depends on the use case.

CVE

- CVE-2021-32842

Credit

This issue was discovered and reported by GHSL team member @JarLob (Jaroslav Lobačevski).

Contact

You can contact the GHSL team at `securitylab@github.com`, please include a reference to `GHSL-2021-125` in any communication regarding this issue.

GitHub

Product

- Features
- Security
- Enterprise
- Customer stories
- Pricing
- Resources

Platform

- Developer API
- Partners
- Atom
- Electron
- GitHub Desktop

Support

- Docs
- Community Forum
- Professional Services
- Status
- Contact GitHub

Company

- About
- Blog
- Careers
- Press
- Shop

- Terms
- Privacy
- Cookie settings