Arbitrary File Write via Archive Extraction (Zip Slip)

Affecting github.com/unknwon/cae/tz package, versions <1.0.1

INTRODUCED: 26 MAY 2020 CVE-2020-7668 CWE-22 FIRST ADDED BY SNYK

Share

How to fix?

Upgrade `github.com/unknwon/cae/tz` to version 1.0.1 or higher.

Overview

github.com/unknwon/cae/tz is a package that provides archiving functionality for `.tar.gz` archives.

Affected versions of this package are vulnerable to Arbitrary File Write via Archive Extraction (Zip Slip). The `ExtractTo` function doesn't securely escape file paths in zip archives which include leading or non-leading "..". This allows an attacker to add or replace files system-wide.

PoC by Georgios Gkitsas

```
package main
import ( cae_tz "github.com/unknwon/cae/tz" )

func main() { file := "relative.tar.gz" z, _ := cae_tz.Open(file) z.ExtractTo(".") }
```

where "relative.tar.gz" is a .tar.gz archive that includes a leading/non-leading ".." in at least one file path.

Details

It is exploited using a specially crafted zip archive, that holds path traversal filenames. When exploited, a filename in a malicious archive is concatenated to the target extraction directory, which results in the final path ending up outside of the target folder. For instance, a zip may hold a file with a "../../file.exe" location and thus break out of the target folder. If an executable or a configuration file is overwritten with a file containing malicious code, the problem can turn into an arbitrary code execution issue quite easily.

The following is an example of a zip archive with one benign file and one malicious file. Extracting the malicous file will result in traversing out of the target folder, ending up in `/root/.ssh/` overwriting the `authorized_keys` file:

```
+2018-04-15 22:04:29 ..... 19 19 good.txt

+2018-04-15 22:04:42 ..... 20 20 ../../../../../../root/.ssh/authorized_keys
```

References

- GitHub Commit

HIGH

Snyk CVSS

Exploit Maturity	Proof of concept
Attack Complexity	Low
Integrity	HIGH

See more

NVD 7.5 HIGH

Do your applications use this vulnerable package?

In a few clicks we can analyze your entire application and see what components are vulnerable in your application, and suggest you quick fixes.

Test your applications

Snyk Learn

Learn about Arbitrary File Write via Archive Extraction (Zip Slip) vulnerabilities in an interactive lesson.

Start learning

Snyk ID	SNYK-GOLANG-GITHUBCOMUNKNWONCAETZ-570384
Published	5 Jun 2020
Disclosed	26 May 2020
Credit	Georgios Gkitsas of Snyk Security Team

Report a new vulnerability | Found a mistake?