[BUG]A carefully crafted RAR archive can trigger an infinite loop while parsing. #73

Closed **Han0nly** opened this issue on Jan 26 · 10 comments

Labels bug released

Han0nly commented on Jan 26 • edited

Describe the bug
A carefully crafted RAR archive can trigger an infinite loop while parsing the file. This could be used to mount a denial of service attack against services that use junrar.

To Reproduce

```
String encodedString =
"UmFyIRoHAM+Qcw4AAAAAKgAAAAAXF3QggE4ASwEAACICAAADZXUl9710qkIdNC4ApIEAAF9fTUlDT1NYXC5fYXBwbGUpdG91Y2gt

byte[] decodedBytes = Base64.getDecoder().decode(encodedString);
InputStream inputStream = new ByteArrayInputStream(decodedBytes);

final Archive archive = new Archive(inputStream);
while (true) {
        FileHeader fileHeader = archive.nextFileHeader();
        if (fileHeader == null) {
                break;
        }
        archive.extractFile(fileHeader, OutputStream.nullOutputStream());
}
```

Expected behavior
Infinite loop.

File
loop-913d3158487310b1b4b74086ab888f5ed56a8493.zip

Environment (please complete the following information):

- OS: Mac OS 12.1 & Ubuntu Linux 16.04 (4.15.0-163-generic)
- Junrar version: 7.4.0

Additional context

It seems this PoC can reach [this while loop] (

junrar/src/main/java/com/github/junrar/unpack/vm/RarVM.java
Lines 227 to 629 in dc3d299

```
227     while (true) {
228         VMPreparedCommand cmd = preparedCode.get(IP);
229         int op1 = getOperand(cmd.getOp1());
230         int op2 = getOperand(cmd.getOp2());
231         switch (cmd.getOpCode()) {
232             case VM_MOV:
233                 setValue(cmd.isByteMode(), mem, op1, getValue(cmd.isByteMode(),
234                         mem, op2)); // SET_VALUE(Cmd->ByteMode,Op1,GET_VALUE(Cmd->ByteMode
235                 break;
236             case VM_MOVB:
```

)
but never break.

gotson commented on Jan 26 Member

How did you find out, if I may ask?

Han0nly commented on Jan 26 • edited ▾ Author

> How did you find out, if I may ask?

Hi **@gotson**, We found this sample using a testing technique called fuzzing.

👍 1

gotson commented on Jan 26 Member

It doesn't seem the provided file is even a rar file, no ?

```
unrar t loop-913d3158487310b1b4b74086ab888f5ed56a8493
Thu Jan 27 11:36:52 2022
```

```
UNRAR 6.10 freeware      Copyright (c) 1993-2022 Alexander Roshal

Corrupt header is found
Main archive header is corrupt

Testing archive loop-913d3158487310b1b4b74086ab888f5ed56a8493

Unexpected end of archive
No files to extract
```

Han0nly commented on Jan 26 Author

Hi **@gotson** , this infinite loop PoC file we provided here is indeed a broken RAR file. We use fuzzing to iteratively mutate some valid RAR files to test the junrar.

gotson commented on Jan 26 Member

Thanks, i manage to reproduce in the tests, will have a look.

👍 1

gotson closed this as completed in `7b16b3d` on Jan 26

github-actions bot pushed a commit that referenced this issue on Jan 26

`chore(release): 7.4.1 [skip ci]` … `d058d84`

github-actions bot pushed a commit that referenced this issue on Jan 26

`chore(release): 7.4.1 [skip ci]` … `1c59d9d`

github-actions bot commented on Jan 26

🎉 This issue has been resolved in version 7.4.1 🎉

The release is available on:

- `v7.4.1`
- GitHub release

Your semantic-release bot 📦🚀

github-actions bot added the released label on Jan 26

gotson added the bug label on Jan 26

Han0nly commented on Jan 27 Author

Hi **@gotson** , are you willing to help us to request a CVE ID through GitHub Security Advisories for this bug, which can cause Denial of Service. You can follow this tutorial to manage your bug fixing and alert any downstream dependencies of the issue so they can patch immediately if using the broken release. Thanks for your help!

gotson commented on Jan 27 Member

> Hi **@gotson** , are you willing to help us to request a CVE ID through GitHub Security Advisories for this bug, which can cause Denial of Service. You can follow this tutorial to manage your bug fixing and alert any downstream dependencies of the issue so they can patch immediately if using the broken release. Thanks for your help!

Thanks, it's a new process to me, but that's actually a good idea.

👍 1

github-actions bot pushed a commit to andrebrait/junrar that referenced this issue on Mar 2

`chore(release): 7.4.1 [skip ci]` … `1de639b`

Han0nly commented on Mar 24 • edited Author

Hi **@gotson** , I found some files which can also trigger this infinite loop. I have tested these on the latest version (7.5.0)
loops.zip.

gotson commented on Mar 24 Member

> Hi **@gotson** , I found some files which can also trigger this infinite loop. I have tested these on the latest version (7.5.0) loops.zip.

Please open a new issue.

junrar locked as **resolved** and limited conversation to collaborators on Mar 24

Assignees

No one assigned

Labels

bug released

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants