golang / go Public

Code Issues 5k+ Pull requests 406 Discussions Actions Projects

New issue

x/image/tiff: OOM from malicious IFD offset #78267

Closed

Labels BugReport FixPending Security

Milestone Unreleased

ZephrFish opened 2 weeks ago · edited by ZephrFish Edits

A crafted 8-byte TIFF file with IFD offset 0xFFFFFFFF causes
buffer.fill() to allocate ~4GB of memory when decoding via
io.Reader (non-ReaderAt path), leading to an out-of-memory
crash in any Go application that calls Decode or DecodeConfig
on untrusted input.

Read the data, and allocate the buffer, in chunks,
to limit memory allocation to the size of the input file.

Fix: golang/image#25
References: https://issuetracker.google.com/issues/494365189

👍 1

ZephrFish mentioned this 2 weeks ago

tiff: cap buffer growth to prevent OOM from malicious IFD offset image#25

ZephrFish added a commit that references this issue 2 weeks ago

`tiff: cap buffer growth to prevent OOM from malicious IFD offset ···`

Verified `81bdee5`

gopherbot added this to the Unreleased milestone 2 weeks ago

gopherbot 2 weeks ago Contributor

Change https://go.dev/cl/757660 mentions this issue: `tiff: cap buffer growth to prevent OOM from malicious IFD offset`

gabyhelp 2 weeks ago

Related Issues

- x/image/tiff: over allocation in DecodeConfig (CVE-2022-41727) #58003 (closed)
- x/image/tiff: lack of limits on compressed tile data [CVE-2023-29408] #61582 (closed)
- x/image/tiff: excessive CPU consumption from no-op loop iterations [CVE-2023-29407] #61581 (closed)
- x/image/tiff: corrupt or malicious paletted images parse successfully and later panic in (*Palleted).At #67624 (closed)
- x/image/tiff: slice bounds out of range #10395 (closed)
- x/image/tiff: index out of range #10597 (closed)
- x/image/tiff: slice bounds out of range #10596 (closed)
- x/image/tiff: integer divide by zero #10393 (closed)

Related Discussions

- [security] Vulnerability in golang.org/x/image/tiff
- [security] golang.org/x/image/tiff fix pre-announcement

(Emoji vote if this was helpful or unhelpful; more detailed feedback welcome in this discussion.)

gabyhelp added **BugReport** 2 weeks ago

jub0bs 2 weeks ago Contributor

@ZephrFish Shouldn't this have been filed as a private report to the security team?

ZephrFish added 2 commits that reference this issue 2 weeks ago

`tiff: read in chunks in buffer.fill to prevent OOM` ··· Verified `b89e8ff`

`tiff: prevent OOM from large IFD offset in buffer.fill` ··· Verified `8e6d978`

dmitshur added **FixPending** 2 weeks ago

gopherbot closed this as completed in 23ae9ed 2 weeks ago

ZephrFish added a commit that references this issue 2 weeks ago

`tiff: cap buffer growth to prevent OOM from malicious IFD offset` ··· 23ae9ed

neild 2 weeks ago Contributor

We've assigned this CVE-2026-33809. This is de-facto PUBLIC track, since it's public now.

👍 3

neild mentioned this 2 weeks ago

x/vulndb: potential Go vuln in golang.org/x/image/tiff: CVE-2026-33809 vulndb#4815

dmitshur added Security 2 weeks ago

gabyhelp mentioned this 2 weeks ago

x/image/font/sfnt: crafted GPOS table causes OOM via unchecked class count product #78382

github-actions mentioned this last week

fix(deps): update module golang.org/x/image to v0.38.0 [security] koki-develop/gat#259

ZephrFish 4 days ago Author

@**neild** would it be possible to get the acknowledgement for that CVE updated to me please?

neild now Contributor

> @**neild** would it be possible to get the acknowledgement for that CVE updated to me please?

Sorry about that, I missed filling out one of the fields. Updated now.

Metadata

Assignees

No one assigned

Labels

BugReport FixPending Security

Type

No type

Projects

No projects

Milestone

Unreleased

Due by December 31, 2099, 83% complete

Relationships

None yet

Development

Code with agent mode

tiff: cap buffer growth to prevent OOM from malicious IFD offset

golang/image

Participants

+1