Infinite recursion on malformed input (parseTypeSystemDefinition) #637

○ Open

WGH- opened this issue on Jul 19 · 6 comments · Fixed by solidwall/graphql-go#1 or tailor-inc/graphql#3 · May be fixed by #642

WGH- commented on Jul 19

Simple input `String r` crashes the parser with infinite recursion.

```go
func TestInfiniteRecursion(t *testing.T) {
        body := `String r`
        source := source.NewSource(&source.Source{
                Body: []byte(body),
        })
        _, err := Parse(
                ParseParams{
                        Source: source,
                        Options: ParseOptions{
                                NoSource: true,
                        },
                },
        )
        if err != nil {
                t.Fatalf("unexpected error: %v", err)
        }
}
```

```
untime: goroutine stack exceeds 1000000000-byte limit
runtime: sp=0xc020178350 stack=[0xc020178000, 0xc040178000]
fatal error: stack overflow

runtime stack:
runtime.throw({0x581369?, 0x67d3c0?})
        /usr/lib/go/src/runtime/panic.go:992 +0x71
runtime.newstack()
        /usr/lib/go/src/runtime/stack.go:1101 +0x5cc
runtime.morestack()
```

```
        /usr/lib/go/src/runtime/asm_amd64.s:547 +0x8b

goroutine 50 [running]:
runtime.mapaccess2_faststr(0x55ce00?, 0xc0000aa510?, {0xc0001630b8, 0x6})
        /usr/lib/go/src/runtime/map_faststr.go:108 +0x3ee fp=0xc020178360 sp=0xc020178358
pc=0x41254e
github.com/graphql-go/graphql/language/parser.parseTypeSystemDefinition(0xc00015a8c0)
        /tmp/graphql/language/parser/parser.go:867 +0x146 fp=0xc02017ca88 sp=0xc02017c9d0
pc=0x533006
github.com/graphql-go/graphql/language/parser.parseTypeSystemDefinition(0xc00015a8c0)
        /tmp/graphql/language/parser/parser.go:867 +0x146 fp=0xc02017ca88 sp=0xc02017c9d0
pc=0x533006
...additional frames elided...
created by testing.(*T).Run
        /usr/lib/go/src/testing/testing.go:1486 +0x35f
```

👍 3

WGH- commented on Jul 19 Author

Just FYI, this test case was found with the new Go 1.18 fuzzer (`*testing.F`).

👍 3 🚀 3

WGH- changed the title ~~Infinite recursion on malformed input~~ **Infinite recursion on malformed input (parseTypeSystemDefinition)** on Jul 19

Invizory added a commit to Invizory/graphql-go that referenced this issue on Jul 29

`Fix infinite recursion in parser` … `edac3f0`

Invizory added a commit to Invizory/graphql-go that referenced this issue on Jul 29

`Fix infinite recursion in parser` … `6d83653`

Invizory linked a pull request on Jul 29 that will close this issue

Fix infinite recursion in type definition parser #642

Open

Invizory added a commit to Invizory/graphql-go that referenced this issue on Jul 29

`Fix infinite recursion in type definition parser` ... `4188bd5`

Invizory commented on Aug 2

This was assigned CVE-2022-37315.

JohnStarich commented on Aug 7

@alex-lange @chris-ramon Would you mind taking a look when you have a chance?

If needed, let me know where I can help. Looks like the above PR may be the needed fix. (Due to the assigned CVE, my team is getting alerts to patch.)

👍 3

jamesdphillips pushed a commit to jamesdphillips/graphql-go that referenced this issue on Aug 10 ⓘ

`Fix infinite recursion in type definition parser` ... `8a04f2f`

jamesdphillips pushed a commit to jamesdphillips/graphql-go that referenced this issue on Aug 10 ⓘ

`Fix infinite recursion in type definition parser` ... `efd2a06`

romko11l mentioned this issue on Sep 7

Fix infinite recursion in type definition parser solidwall/graphql-go#1

Merged

miseyu mentioned this issue on Oct 12

fix: graphql-go#637 tailor-inc/graphql#3

Merged

JohnStarich commented on Oct 13

@chris-ramon @sogko Are any maintainers available to take a look at this? This CVE is now 2 months old.

We can't afford to continue using dependencies with active CVEs. I'd much prefer to avoid dropping this dependency. If there's anything the community can do to help, please shout.

WGH- commented on Oct 13 Author

I apologize for dropping zero-day DoS without consideration...

❤️ 3 👀 1

Pashugan commented 5 days ago • edited ▾

`nancy` fails on this vulnerability now. Any chance to merge the fix?

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

Fix infinite recursion in type definition parser
solidwall/graphql-go

fix: graphql-go#637
tailor-inc/graphql

Fix infinite recursion in type definition parser
Invizory/graphql-go

4 participants