Authorization Bypass Through User-Controlled Key in emicklei/go-restful

2



✓ Valid) Reported on Mar 7th 2022

Description

Hello go restful maintainer team, I would like to report a security concerning your CORS Filter feature.

Go restful allows user to specify a CORS Filter with a configurable AllowedDomains param which is an array of domains allowed in CORS policy.

However, although there's is already another param called allowedOriginPatterns used for matching origin using regular expression, all domains in AllowedDomains is also used as regular expression to check for matching origin in this code in file cors_filter.go:

```
if len(c.allowedOriginPatterns) == 0 {
            // compile allowed domains to allowed origin patterns
            allowedOriginRegexps, err := compileRegexps(c.AllowedDomains)
            if err != nil {
                return false
            c.allowedOriginPatterns = allowedOriginRegexps
        for , pattern := range c.allowedOriginPatterns {
            if allowed = pattern.MatchString(origin); allowed {
                break
```

So by this, if the user input example.com to be one of domain in AllowedDoma* starting with example.com would be acceptable.

Chat with us

Proof of Concept

Install go restful and create a file main.go with this content:

```
package main
import (
    restful "github.com/emicklei/go-restful/v3"
    "net/http"
func main() {
    container := restful.NewContainer()
    ws := new(restful.WebService)
    ws.Route(ws.GET("hello").To(hello))
    container.Add(ws)
    server := &http.Server{Addr: ":8000", Handler: container}
    //container.Filter(logHeaders)
    cors := restful.CrossOriginResourceSharing{
        ExposeHeaders: []string{"X-My-Header"},
        AllowedDomains: []string{"example.com"},
        CookiesAllowed: true,
        Container: container,
    }
    container.Filter(cors.Filter)
    server.ListenAndServe()
}
func hello(req *restful.Request, resp *restful.Response) {
    io.WriteString(resp, "world")
}
```

In the above code, example.com is configured as an allowed domain.

Run the above code and access link /hello with Origin Header = example.cr and see that the request gets through CORS policy and response looks like Chat with us

HTTP/1.1 200 OK

Access-Control-Allow-Credentials: true

Access-Control-Allow-Origin: example.com.hacker.domain

Access-Control-Expose-Headers: X-My-Header

Date: Mon, 07 Mar 2022 13:31:08 GMT

Content-Length: 5

Content-Type: text/plain; charset=utf-8

Connection: close

world

Impact

This vulnerability is capable of breaking CORS policy and thus allowing any page to make requests, retrieve data on behalf of other users.

Occurrences



cors_filter.go L135

CVE

Vulnerability Type

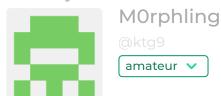
CWE-639: Authorization Bypass Through User-Controlled Key

Severity

Visibility

Status

Found by



Chat with us

This report was seen 2.018 times.

We are processing your report and will contact the **emicklei/go-restful** team within 24 hours. 9 months ago

We created a **GitHub Issue** asking the maintainers to create a **SECURITY.md** 9 months ago

Jamie Slome 8 months ago

Admin

As requested by the maintainer, a GitHub Issue has been created with the details of the report:

https://github.com/emicklei/go-restful/issues/489

Morphling submitted a patch 8 months ago

Jamie Slome 6 months ago

Admin

Current status of the report can be tracked via this PR.

Jamie Slome validated this vulnerability 6 months ago

M0rphling has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

Jamie Slome marked this as fixed in v3.8.0 with commit fd3c32 6 months ago

The fix bounty has been dropped 🗶

This vulnerability will not receive a CVE x

cors_filter.go#L135 has been validated 🗸

Jamie Slome 6 months ago

Chat with us

In accordance with the full disclosure of the issue and validation from the maintainer nere - this report has been marked as valid and fixed in v3.8.0 as stated by the maintainer here

Great work to all involved!



Sign in to join this conversation

huntr

part of 418sec