

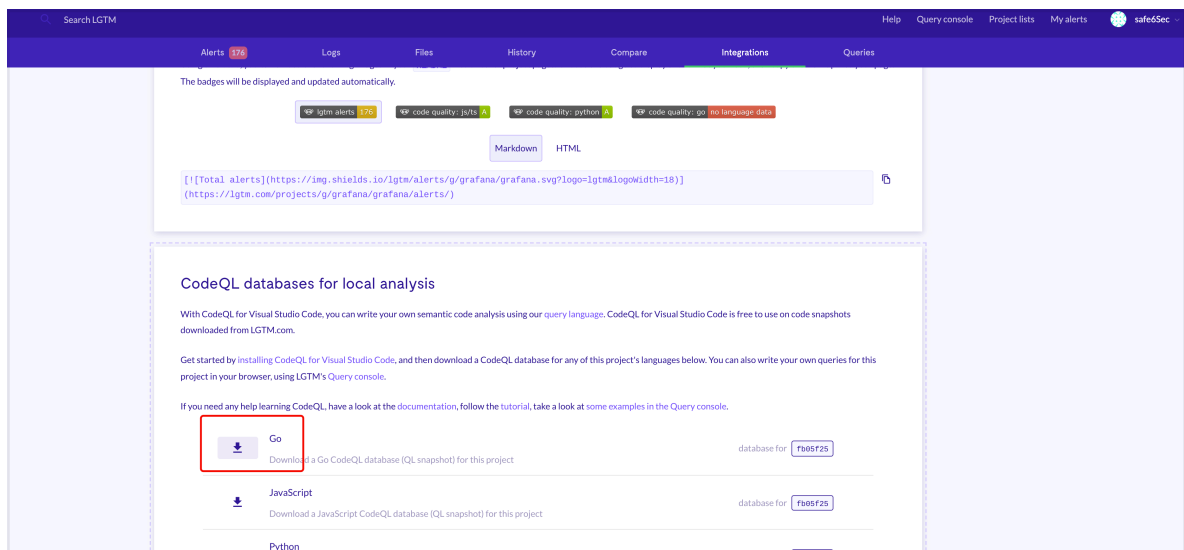
# 用 codeql 分析 grafana 最新任意文件读取

## 生成数据库

最近学了一下 codeql，刚好拿这个来练一下手。简单记录一下，有疑问的师傅可以一起探讨。

先从 lgtm 把数据库下下来，发现洞已经被修。

<https://lgtm.com/projects/g/grafana/grafana/ci/#ql>



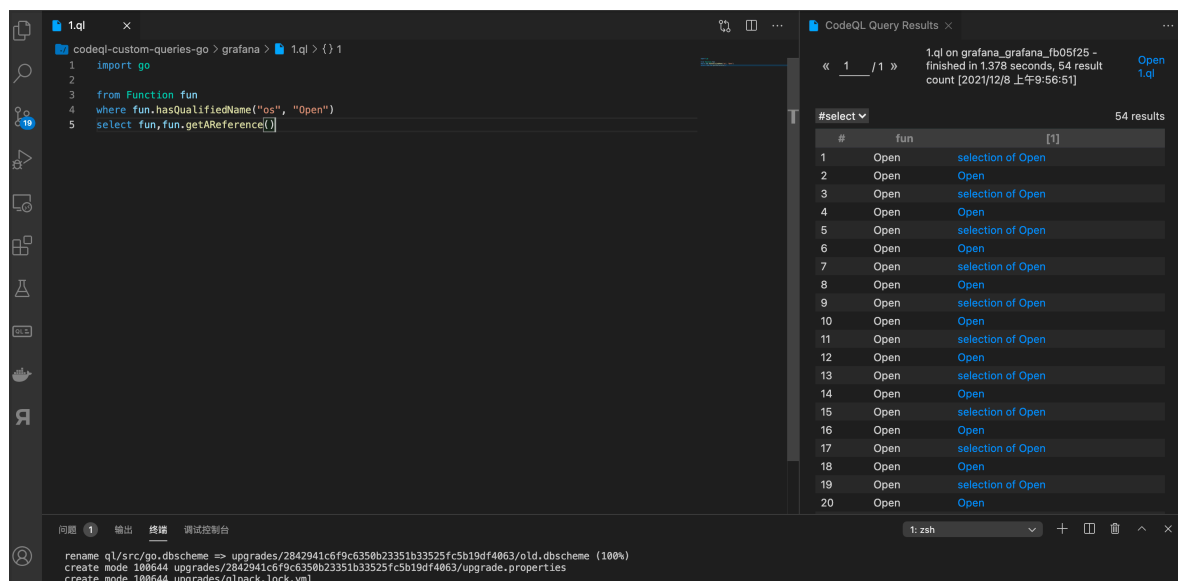
只能自己编译了

codeql database create /Users/safe6/codeql/database/gf --language="go" --source-root=/Users/safe6/Desktop/grafana-8.2.6 --overwrite

编译好的库，有需要的师傅可以找我要。

## 开始分析

各位大佬都把 sink 分析好了，我们直接来找 os.open 的全部引用。



The screenshot shows the Grafana CodeQL interface. On the left, a query is written in the editor:

```
1 import go
2
3 from Function fun
4 where fun.hasQualifiedName("os", "Open")
5 select fun, fun.getReference()
```

On the right, the 'CodeQL Query Results' panel shows the query execution details and a table of results. The query is '1.q1 on grafana\_grafana\_fb05f25 - finished in 1.378 seconds, 54 result count [2021/12/8 上午9:56:51]'. The table has 54 results, with the first 20 shown:

#	fun	[1]
1	Open	selection of Open
2	Open	Open
3	Open	selection of Open
4	Open	Open
5	Open	selection of Open
6	Open	Open
7	Open	selection of Open
8	Open	Open
9	Open	selection of Open
10	Open	Open
11	Open	selection of Open
12	Open	Open
13	Open	selection of Open
14	Open	Open
15	Open	selection of Open
16	Open	Open
17	Open	selection of Open
18	Open	Open
19	Open	selection of Open
20	Open	Open

At the bottom, a terminal window shows the following commands and output:

```
rename q1/src/go.dbscheme => upgrades/2842941c6f9c6350b23351b33525fc5b19df4063/old.dbscheme (100%)
create mode 100644 upgrades/2842941c6f9c6350b23351b33525fc5b19df4063/upgrade.properties
create mode 100644 upgrades/q1pack.lock.yml
```

居然有 50 多个,我们先不管。也不知道能不能挖出新的洞。

接下来开始找 source。

进到关键类，可以看到有很多种接收方式。

```

// RouterRegister allows you to add routes and webhandlers
// that the web server should serve.
type RouteRegister interface {}

// Get adds a list of handlers to a given route with a GET HTTP verb
Get(string, ...web.Handler)

// Post adds a list of handlers to a given route with a POST HTTP verb
Post(string, ...web.Handler)

// Delete adds a list of handlers to a given route with a DELETE HTTP verb
Delete(string, ...web.Handler)

// Put adds a list of handlers to a given route with a PUT HTTP verb
Put(string, ...web.Handler)

// Patch adds a list of handlers to a given route with a PATCH HTTP verb
Patch(string, ...web.Handler)

// Any adds a list of handlers to a given route with any HTTP verb
Any(string, ...web.Handler)

// Group allows you to pass a function that can add multiple routes
// with a shared prefix route.
Group(string, func(RouteRegister), ...web.Handler)

// Insert adds more routes to an existing Group.
Insert(string, func(RouteRegister), ...web.Handler)

// Register iterates over all routes added to the RouteRegister
// and add them to the `Router` pass as an parameter

```

查出来 300 多个 api 接口

codeql-custom-queries-go > grafana > 2.q1 > {} 2

```

1 import go
2
3 from Function fun
4 where fun.hasQualifiedName("github.com/grafana/grafana/pkg/api/routing.RouteRegister")
5 ["Get", "Post", "Delete", "Put", "Patch", "Any"]
6 select fun.getReference()

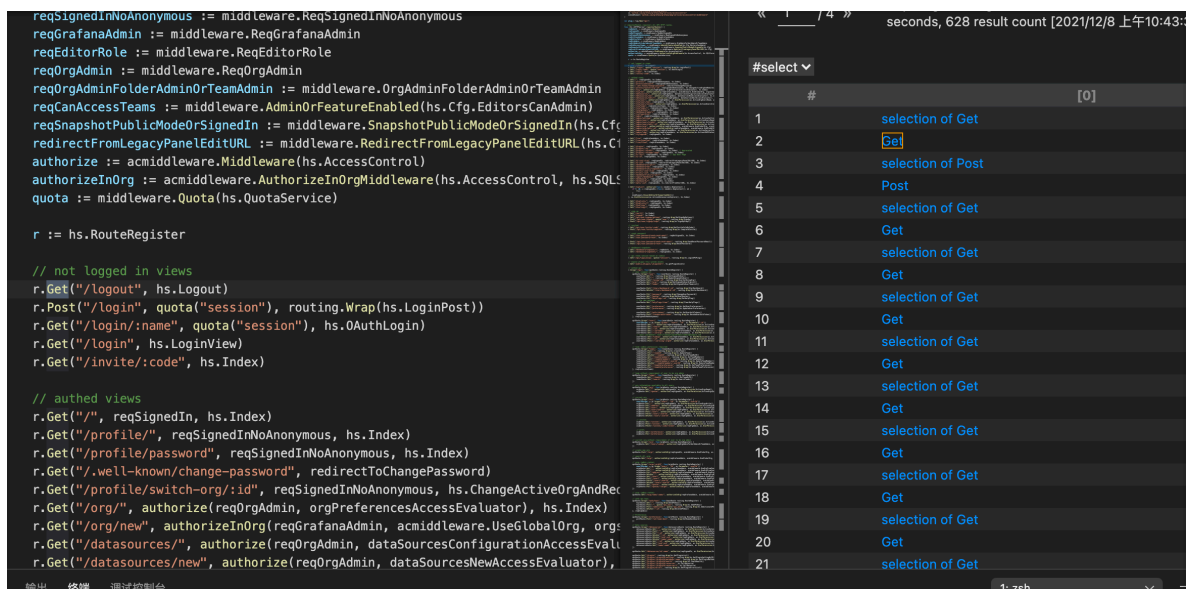
```

« 1 / 4 » 2.q1 on grafana\_grafana\_fb05f25 - finished in 0.387 seconds, 628 result count [2021/12/8 上午10:43:31] Open 2.q1

#select [0] 628 results

#	selection of Get
1	selection of Get
2	Get
3	selection of Post
4	Post
5	selection of Get
6	Get
7	selection of Get
8	Get
9	selection of Get
10	Get
11	selection of Get
12	Get
13	selection of Get
14	Get
15	selection of Get
16	Get
17	selection of Get
18	Get
19	selection of Get
20	Get
21	selection of Get

问题 1 输出 终端 调试控制台 1: zsh



下面开始污点跟踪

定义 source

```
class GfSource extends DataFlow::Node {
  GfSource(){
    exists( Function fun |
      fun.hasQualifiedName("github.com/grafana/grafana/pkg/api/routing.RouteRegister",
        ["Get","Post","Delete","Put","Patch","Any"]) and
      //["Get","Post"]) and
      fun.getAResource()=this.asExpr()
    )
  }
}
```

Sink

```
override predicate isSink(DataFlow::Node sink) {
  exists(Function fun ,CallExpr call|
    fun.hasQualifiedName("os", "Open") and
    call.getTarget() = fun and
    call.getAnArgument()= sink.asExpr()
  )
}
```

isAdditionalTaintStep

```

/**
 * sink参数只能是两个，第二个参数才是真正的sink
 */
override predicate isAdditionalTaintStep(DataFlow::Node expSrc, DataFlow::Node expDest) {
    exists(CallExpr call|
        call=expSrc.asExpr() and
        call.getArgument(0).getType().toString()=="string" and
        call.getArgument(1).(CallExpr).getTarget().getAParameter()= expDest.asParameter()
    )
}

```

尝试跑了一下并没有结果

The screenshot shows the CodeQL IDE interface. On the left, the query file '3.q' is open, displaying the following code:

```

21 class Gfconfig extends TaintTracking::Configuration{
22
23     Gfconfig() { this = "Gfconfig" }
24
25     override predicate isSource(DataFlow::Node source) {
26         source instanceof GfSource
27     }
28
29     override predicate isSink(DataFlow::Node sink) {
30         exists(Function fun ,CallExpr call|
31             fun.hasQualifiedName("os", "Open") and
32             call.getTarget() = fun and
33             call.getAnArgument()= sink.asExpr()
34         )
35     }
36
37     /**
38      * sink参数只能是两个，第二个参数才是真正的sink
39      */
40     override predicate isAdditionalTaintStep(DataFlow::Node expSrc, DataFlow::Node expDest) {
41         exists(CallExpr call|
42             call=expSrc.asExpr() and
43             call.getArgument(0).getType().toString()=="string" and
44             call.getArgument(1).(CallExpr).getTarget().getAParameter()= expDest.asParameter()
45         )
46     }
47 }
48
49 from Gfconfig gf,DataFlow::PathNode source,DataFlow::PathNode sink
50 where gf.hasFlowPath(source, sink)
51 select source.getNode(), source, sink, "test"
52
53

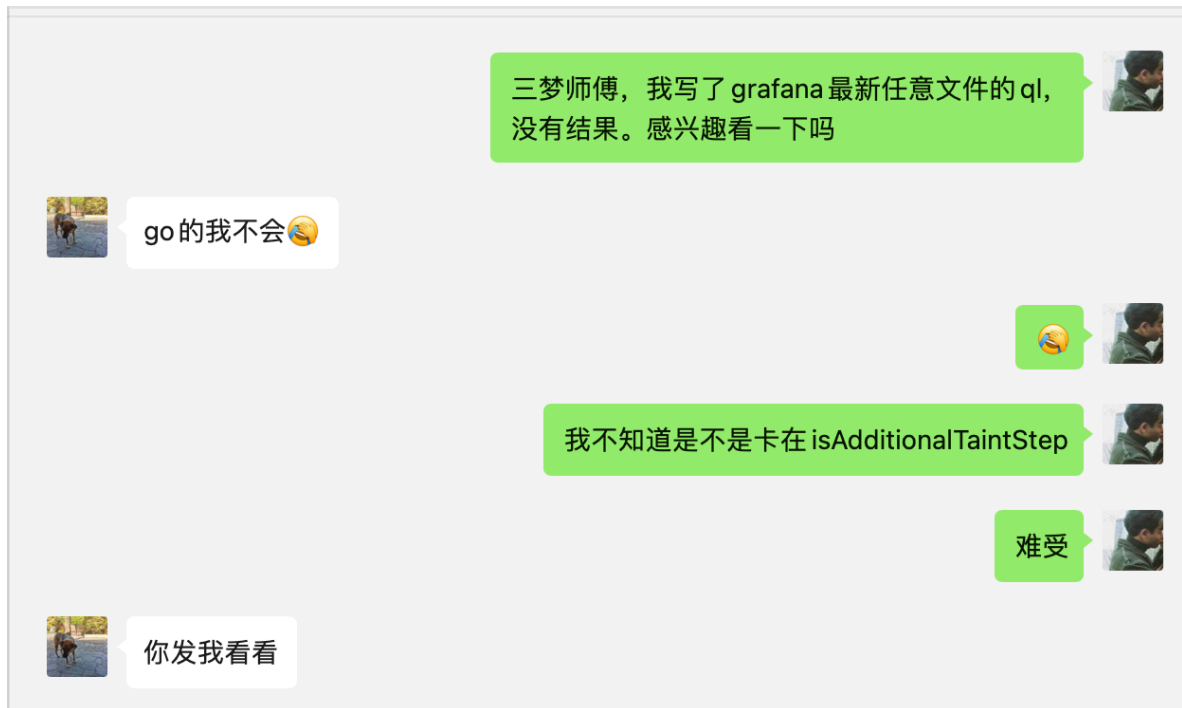
```

On the right, the 'CodeQL Query Results' panel shows the following information:

- Query: 3.q on gf - finished in 0.068 seconds, 0 result count [2021/12/9 下午1:06:05]
- Alerts: 0 results
- Show results in Problem List: ☐
- No Alerts

于是找了三梦师傅，请教一下。

热心的三梦师傅，直接开撸。



经过三梦师傅，指点后，改造了一下 source。

```
class GfSource1 extends DataFlow::Node {
  GfSource1(){
    exists( Function fun, CallExpr call, SelectorExpr se |
      call.getTarget().hasQualifiedName("github.com/grafana/grafana/pkg/api/routing.RouteR
[\"Get\", \"Post\", \"Delete\", \"Put\", \"Patch\", \"Any\"] ) and
      // [\"Get\", \"Post\"] ) and
      (call.getAnArgument() = se or call.getAnArgument().getAChildExpr() = se) and
      fun.getAReference() = se.getSelector() and
      fun.getAParameter() = this.asParameter()
    )
  }
}
```

再次查询，这次有了结果。可是我们想要的并没有在里面

CodeQL Query Results

Message

- test dashboard.go:424:7
  - Path
    - 1 definition of hs : pointer type dashboard.go:424:7
    - 2 filePath dashboard.go:456:23
  - Path
    - 1 definition of hs : pointer type dashboard.go:424:7
    - 2 implicit dereference : HTTPServer dashboard.go:448:14
    - 3 filePath dashboard.go:456:23
  - Path
    - 1 definition of hs : pointer type dashboard.go:424:7
    - 2 selection of Cfg : pointer type dashboard.go:448:14
    - 3 filePath dashboard.go:456:23
- test plugins.go:207:61
  - Path
    - 1 definition of apiCmd : ImportDashboardCommand plugins.go:207:61
    - 2 selection of Path : string plugins.go:221:75
    - 3 definition of path : string dashboard\_import.go:24:52
    - 4 path : string dashboard\_import.go:30:56
    - 5 definition of path : string dashboards.go:73:56
    - 6 dashboardFilePath dashboards.go:84:25
- test plugins.go:371:59
  - Path
    - 1 definition of dto : InstallPluginCommand plugins.go:371:59
    - 2 selection of Version : string plugins.go:374:61
    - 3 definition of version : string manager.go:725:65
    - 4 version : string manager.go:756:51
    - 5 definition of version : string installer.go:94:60
    - 6 pluginZipURL : string installer.go:148:42

回过头看看，发现这个api的路由用到了\*，然后在具体方法里面用 Params 进行获取

```
// expose plugin file system assets
r.Get("/public/plugins/:pluginId/*", hs.getPluginAssets)
```

```
requestedFile := filepath.Clean(macaron.Params(c.Req)["*"])
pluginFilePath := filepath.Join(plugin.PluginDir, requestedFile)

if !plugin.IncludedInSignature(requestedFile) {
    hs.log.Warn("Access to requested plugin file will be forbidden in upcoming version", "file", requestedFile)
}
```

那么我们需要继续加个isAdditionalTaintStep，把断掉的接上。

经过各种查资料发现赋值语句可以满足需求，赋值语句具体的 Examples 如下

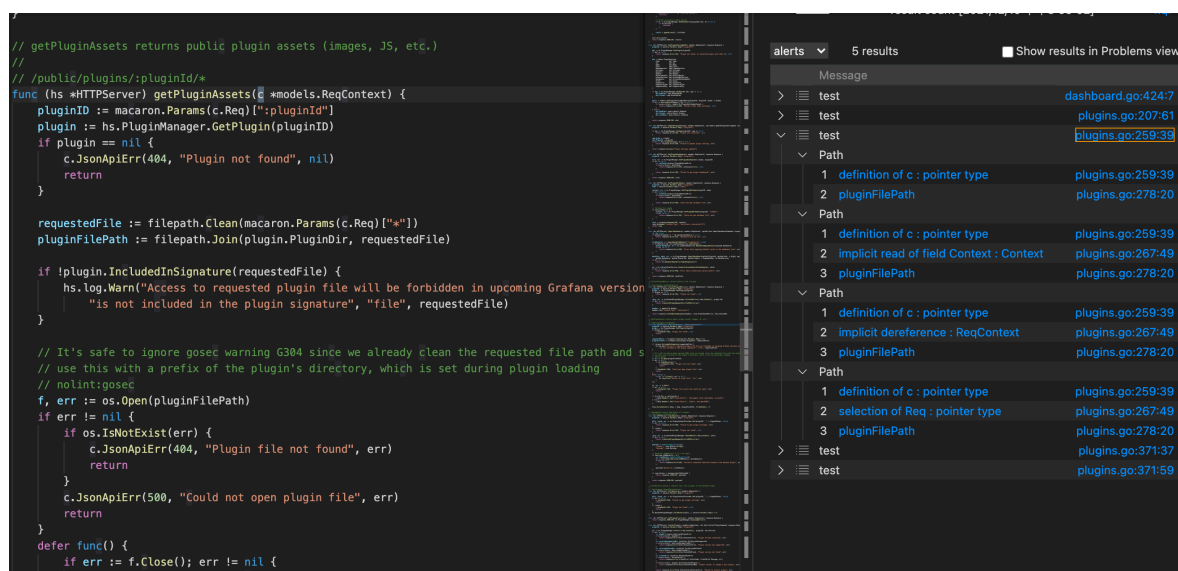
```
/**
 * A simple assignment statement, that is, an assignment without a compound operator.
 *
 * Examples:
 *
 * ```go
 * x := 1
 * *p = f()
 * a[i] = 23
 * (k) = <-ch // same as: k = <-ch
 * ```
 */
```

最后写出来的isAdditionalTaintStep如下

```
predicate isOther(DataFlow::Node expSrc, DataFlow::Node expDest) {
    exists(CallExpr call, SimpleAssignStmt sas |
        call.getTarget().getName().toString()=="Params" and
        call.getArgument(0)=expSrc.asExpr() and
        sas.getRhs().getAChild()==call.getParent*().getAChild() and
        sas.getRhs()==expDest.asExpr()
    )
}
```

再来看看结果，成功了！！！！！！





## 最后

Codeql 资料真挺少的，全靠官方文档续命。

最后还是要感谢三梦师傅，在我学习 codeql 给到的帮助。

代码放在：<https://github.com/safe6Sec/codeql-grafana>

正在整理的一点笔记 <https://github.com/safe6Sec/CodeqlNote>