

中级命令注入

高级命令注入

防范措施

中级命令注入

Command Execution Source

```
1 <?php
2
3 if( isset( $_POST[ 'submit' ] ) ) {
4
5     $target = $_REQUEST[ 'ip' ];
6
7     // Remove any of the characters in the array (blacklist).
8     $substitutions = array(
9 #         '&&' => '', // '&&'是变量的key, 后面是变量的值(这里的值为空)!
10         ';' => '', // 这个变量是分号;
11     ); // 这是php里面的数组!
12
13     $target = str_replace( array_keys( $substitutions ), $substitutions, $target ); // $
14
15     // Determine OS and execute the ping command.
16     if (strcasecmp(php_uname('s'), 'Windows NT')) {
17
18         $cmd = shell_exec( 'ping ' . $target );
19         echo '<pre>'.$cmd.'</pre>';
20
21     } else {
22
23         $cmd = shell_exec( 'ping -c 3 ' . $target );
24         echo '<pre>'.$cmd.'</pre>';
25
26     }
27 }
28
29 ?>
```

高级命令注入

```
1 <?php
2
3 if( isset( $_POST[ 'submit' ] ) ) {
4
5     $target = $_REQUEST["ip"];
6
7     $target = stripslashes( $target );//去掉魔法引号!
8
9
10    // Split the IP into 4 octects
11    #     $octet = explode(".", $target);//看下面的一个图 ! 就是把target里面的ip地址用.分割,而后行
12
13    // Check IF each octet is an integer
14    if ((is_numeric($octet[0])) && (is_numeric($octet[1])) && (is_numeric($octet[2]))
15        //(is_numeric($octet[0])判断第一个数值是不是数字; (sizeof($octet) == 4) 表示数值的个数为4
16
17    // If all 4 octets are int's put the IP back together.
18    $target = $octet[0].'.'.$octet[1].'.'.$octet[2].'.'.$octet[3];
19
20
21    // Determine OS and execute the ping command.
22    if (stristr(PHP_UNAME('s'), 'Windows NT')) {
23
24        $cmd = shell_exec( 'ping ' . $target );
25        echo '<pre>'.$cmd.'</pre>';
26
27    } else {
28
29        $cmd = shell_exec( 'ping -c 3 ' . $target );
30        echo '<pre>'.$cmd.'</pre>';
31
32    }
33
34 }
35
36 else {
37     echo '<pre>ERROR: You have entered an invalid IP</pre>';
38 }
39
40
41 }
42
43 ?>
44
```

代码分析

- `$octet = explode(".", $target);`

通过`explode`函数以 "." 为分隔符将`$target`变量中的IP地址进行分割，分割后会得到一个数组，并赋值给变量`$octet`。

- `if ((is_numeric($octet[0])) && (is_numeric($octet[1])) && (is_numeric($octet[2])) && (is_numeric($octet[3])) && (sizeof($octet) == 4))`

用`is_numeric`函数依次判断`$octet`数组中的每个值是否是数字型数据，并且还使用`sizeof`函数判断`$octet`数组中元素的个数是否是4个。

edu.51cto.com

防范措施

常见防范措施

- 对传入的命令进行严格过滤；
- 在后台对应用的权限进行控制（比如可以给PHP建立独立的账号，控制PHP的启动权限）。

- 1.对传入的命令进行严格过滤
- 2.在后台对应用的权限进行控制;

漏洞防御

- `EscapeShellCmd()`函数

把字符串中所有可能瞒过Shell而去执行另外一个命令的字符转义，如管道符（|）、分号（;）、重定向（>）、从文件读入（<）等。

- `EscapeShellArg()`函数

在给定的字符串两边加上单引号，并把字符串中的单引号转义，这样这个字符串就可以安全地作为命令的参数。

- 修改low级别网页文件`dwa\vulnerabilities\exec\source\low.php`进行测试。

edu.51cto.com

1.EscapeShellCmd()函数; (最好记忆!!!)

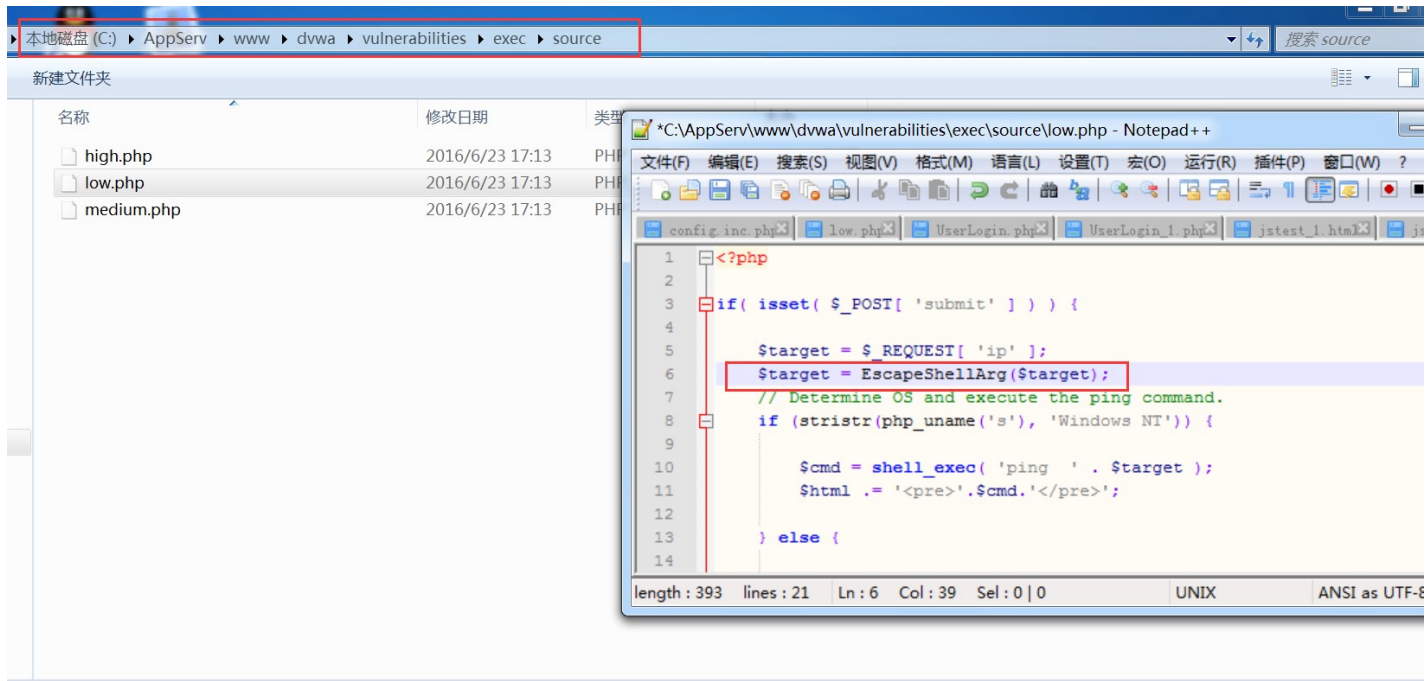
把字符串里面所有可能瞒过shell而去执行另外一个命令的字符转义,如管道符(),分号(;),重定向(>);从文件读入(<)等;

2.EscapeShellArg()函数; (这个函数也比较重要,最好记忆!!!)

在给定的字符串两边加上单引号,并把字符串中的单引号转义,这样这个字符串就可以安全地作为命令的参数;

3.修改low级别网页文件 dvwa\vulnerabilities\exec\source\low.php 进行测试!

这里举个例子:(修改网页源码!)



这里没有效果了!

● strstr(str1,str2)函数

用于判断字符串str2是否是str1的子串。如果是,则该函数返回str2在str1中首次出现的地址;否则,返回NULL。。

2016任务三

● strstr(str1,str2)函数

用于判断字符串str2是否是str1的子串。如果是，则该函数返回str2在str1中首次出现的地址；否则，返回NULL。

□ strstr("abcd1234","cd1")=3 I

strstr()函数的用法如上;