【cve-2018-11759】针对Apache mod_jk 访问控制的绕过方法

0aspir1ng0 / 2019-01-30 07:48:00 / 浏览数 1881 技术文章 翻译文章 顶(0) 踩(0)

本文为《CVE-2018-11759JApache mod_jk access control bypass》的翻译文章。

嘿，各位大佬们：

我非常激动向你们介绍一个我同事（ID为Xel）和我在一次渗透测试中发现的新漏洞：CVE-2018-11795

一年前，我在Wordpress上面才拥有我的第一个CVE，但是现在我们又搞到了第二个！

漏洞概要

1. 由于httpd和Tomcat在路径处理规范上存在差异，因此可以绕过Apache mod_jk Connector 1.2.0版本到1.2.44版本上由JkMount httpd指令所定义端点的访问控制限制。
2. 值得注意的是，如果一个只有只读权限的jkstatus的接口可以访问的话，那么就有可能能够公开由mod_jk模块给AJP提供服务的内部路由。
3. 不仅如此，如果一个具有读写权限的jkstatus接口可供访问，我们就能通过修改AJP的配置文件中相关配置来劫持或者截断所有经过mod_jk的流量，又或者进行内部的端

漏洞详情

比对CVE-2018-1323的漏洞补丁

在我们一位客户的一次基于Apache mod_jk 模块的JBoss / Tomcat 网络服务器进行渗透测试时候，我们发现服务器上存在CVE-2018-1323漏洞（是由Biznet Bilisim A.S.公司的Alphan YAVAS发现并提交的），这会影响到mod_jk模块，同时所影响版本的范围到1.2.42。

为理解其漏洞原理，我们分析了1.2.43版本的mod_jk 补丁。在这个过程中，我们意识到这个漏洞是一个路径穿越问题，是基于Apache httpd、Tomcat或者其他Java服务器的，而且关于他们在处理当用分号进行跨目录路径遍历时各自路径解析的差异点。

Apache httpd将url中的分号视作路径解析中的普通字符，然而Tomcat却将他们当作查询分隔符（和"?"类似）。因此，攻击者可以获取到例如这样的路径：`http://server/java`。*

译者注："..."是两个点加一个空格

这就允许攻击者可以获取到本不应该在Tomcat上可以访问的资源。

```
710  -    /* d) remove trailing xx/.. segment. */
711  -    if (l == 2 && name[0] == '.' && name[1] == '.')
712  -        name[0] = '\0';
713  -    else if (l > 2 && name[l - 1] == '.' && name[l - 2] == '.'
714  -             && name[l - 3] == '/') {
715  -        l = l - 4;
716  -        if (l >= 0) {
717  -            while (l >= 0 && name[l] != '/')
718  -                l--;
719  -            l++;
709  +    /* Third pass.
710  +     * Remove /xx/../ segments including those with path parameters such as
711  +     * /xxx/..;foo=bar/
712  +     * Trailing segments will be removed but leading /../ segments are an error
713  +     * condition.
714  +     */
715  +    for (l = 1, w = 1; name[l] != '\0';) {
716  +        if (name[l] == '.' && name[l + 1] == '.' &&
717  +                (name[l + 2] == '/' || name[l + 2] == ';' || name[l + 2] == '\0') &&
718  +                        (l == 0 || name[l - 1] == '/')) {
719  +
720  +            // Wind w back to remove the previous segment
721  +            if (w == 1) {
722  +                return BAD_NORMALIZATION;
723  +            }
724  +            do {
725  +                w--;
726  +            } while (w != 0 && name[w - 1] != '/');
727  +
728  +            // Move l forward to the next segment
729  +            l += 2;
730  +
731  +            while (name[l] != '/' && name [l] != '\0') {
732  +                l++;
733  +            }
734  +            if (name[l] != '\0') {
735  +                l++;
736  +            }
```

图1：比对CVE-2018-1323

我们现在已经确定了这个漏洞其实并未完全挖掘出来，因为mod_jk的补丁虽然确实修复了针对mod_jk的特定路径遍历攻击，但是还是没有解决mod_jk对分号路径解析方式

探测JK状态管理器

探测jkstatus

jkstatus是mod_jk模块的管理界面。当设为读写权限的时候，它允许通过配置AJP连接Java Web服务器来代理HTTP请求。

通常，我们可以限制对jkstatus的访问，比如使用如下的httpd指令：

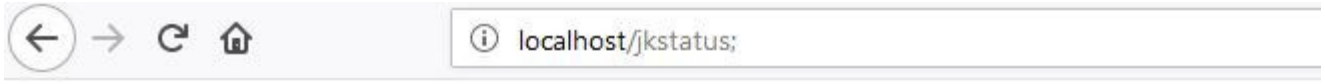
```
<Location /jkstatus>
JKMount jk-status
Require ip 127.0.0.1
</Location>
```

这条指令会阻拦任何外部资源对jkstatus的访问。

Forbidden

You don't have permission to access /jkstatus on this server.

我们发现通过在/jkstatus后面注入分号，就可以绕过这样的限制。



JK Status Manager for localhost:80

Server Version: Apache/2.4.6 (CentOS) mod_jk/1.2.44 Server Time: 2018-11-01 11:10:56 +0000
JK Version: mod_jk/1.2.44 Unix Seconds: 1541070656

Start auto refresh (every 10 seconds) | Change format XML

[Read Only] [Dump] [S=Show only this worker, E=Edit worker, R=Reset worker state, T=Try wor

Listing Load Balancing Worker (1 Worker) [Hide]

[S|E|R] Worker Status for loadbalancer

Type	Sticky Sessions	Force Sticky Sessions	Retries	LB Method	Locking	Recover Wait Time	Error Escalation Time	Max Reply Timeouts	[
lb	True	False	2	Request	Optimistic	60	30	0	

Good	Degraded	Bad/Stopped	Busy	Max Busy	Next Maintenance	Last Reset [Hide]
2	0	0	0	0	13/73	7027

Balancer Members [Hide]

Name	Type	Hostname	Address:Port	Source	Connection Pool Timeout	Connect Timeout	Prep Time
node1	ajp13	cve-2018-11759_client1_1	192.168.160.2:8009	undefined	0	10000	1000
node2	ajp13	cve-2018-11759_client2_1	192.168.160.4:8009	undefined	0	10000	1000

	Name	Act	State	D	F	M	V	Acc	Sess	Err	CE	RE	Wr	Rd	Busy	M
[S|E|R]	node1	ACT	OK/IDLE	0	1	1	0	0 (0/sec)	0 (0/sec)	0	0	0	0 (0 /sec)	0 (0 /sec)	0	0
[S|E|R]	node2	ACT	OK/IDLE	0	1	1	0	0 (0/sec)	0 (0/sec)	0	0	0	0 (0 /sec)	0 (0 /sec)	0	0

Edit this attribute for all members: Activation Go

图3：jkstatus访问控制绕过（分号注入）

上图可以看到，在url分号之后提交的get参数，就可以成功向jkstatus请求修改其访问权限的配置。

图4：url中分号后面get参数部分能被解析

如果给jkstatus设定成具有读写访问权限的配置，要绕过jkstatus访问控制的话，其产生的影响就等同于通过更改工作人员使用的端口来实现对所有由mod_jk供应的应用程序

理论上讲，通过将AJP的目标和端口修改为内部主机和其对应的端口，我们也可以进行内部TCP端口扫描，这是因为Tomcat和httpd的jkstatus返回的错误信息不一致的缘故

译者注：错误网关：Bad gateway；服务不可用：Service Unavaliable

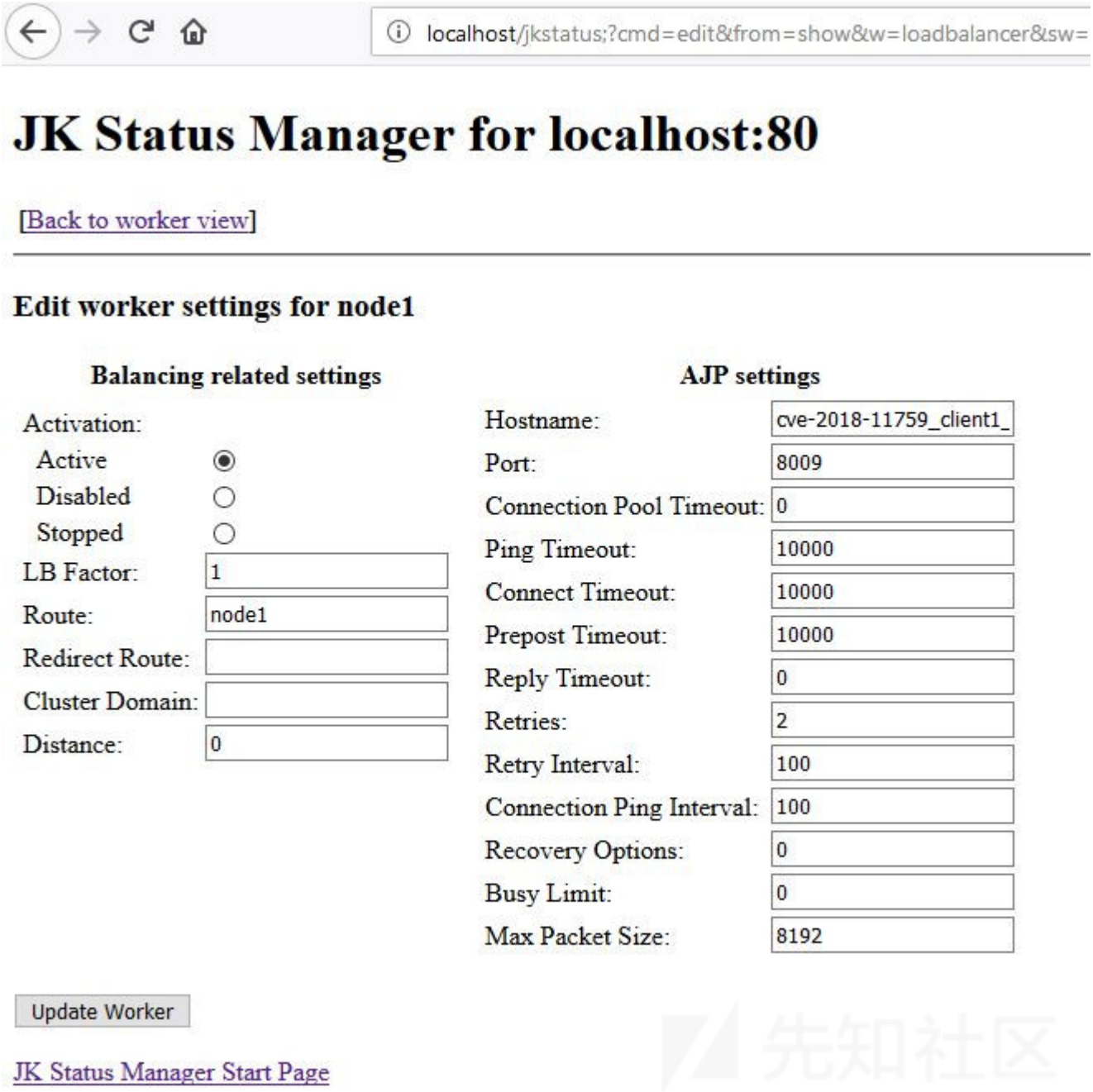


图5：AJP：可以提交任意主机名和端口

jkstatus（如果是只读权限的配置）还会公开内部服务器主机名、ip、端口、mod_jk模块服务的服务器和路由、以及文件系统上http服务器的绝对路径。

这种对访问控制权限绕过的手法具有很大的破坏效果，但是必须要注意一点的就是，通过JkMount指令定义任何端点的访问控制都有可能被分号注入绕过。

漏洞索引

mod_jk（1.2.46版本）已经提供了相应补丁，其他的修复措施（注意，并一定能完全防御，只是起缓解作用）包括有：使用例如`/jkstatus*`这样的位置值设定。

Github POC

github上我们的库里面有一个docker环境可以拿来测试复现这个漏洞。

漏洞挖掘时间线

- 2018/09/06：第一次和Apache Tomcat安全团队上报此漏洞
- 2018/09/06：第一次收到Apache Tomcat安全团队对此漏洞的回应
- 2018/10/13：mod_jk 1.2.46版本补丁发行
- 2018/10/31：发布CVE-2018-11759公告
- 2018/11/01：漏洞揭露公示完毕

漏洞挖掘人员

两位来自immunIT公司的Raphaël Arrouas（ID为Xel）和Jean Lejeune（ID为Nitrax）同志。

上一篇：安全工具——SET工具包 下一篇：渗透测试-从打印机到拿下主域控制器权限

1. 0 条回复
   - 动动手指，沙发就是你的了！