登录

CVE-2018-15685：Electron WebPreferences远程代码执行漏洞

mss**** / 2018-08-26 11:34:43 / 浏览数 3717 技术文章 技术文章 顶(0) 踩(0)

原文：https://www.contrastsecurity.com/security-influencers/cve-2018-15685

最近，Contrast
Labs安全人员发现了一个影响应用程序的远程代码执行（RCE）漏洞，具体来说，该漏洞能够在不同版本的Electron（3.0.0-beta.6、2.0.7、1.8.7和1.7.15）中打开嵌套子窗

对于该漏洞的POC代码，可以从这里找到。

关于Electron

简单来说，Electron是一个可以为日常使用的各种应用程序提供支持的框架。举例来说，Slack、Atom、Visual Studio Code、WordPress Desktop、Github Desktop、Skype和Google
Chat都是运行在Electron框架之上的应用程序，当然，这些只是其中一小部分应用程序而已。该框架允许开发人员将传统Web应用程序快速移植为本机跨平台桌面应用程序。

漏洞详情

最近，@SecurityMB在一篇介绍Google
Chat安全漏洞的文章中指出，攻击者可以创建一个链接，当受害者点击该链接时，就会从谷歌站点重定向到攻击者控制的内容，但是，他仍然位于该Electron应用中。在Go

虽然谷歌很快就修复了这个与重定向相关的安全问题，但我仍然好奇代码执行的确切路径到底是什么。我们知道，Google
Chat是建立在Electron框架之上的。Electron有一个详细的安全指南（可从这里访问），并且，Google也遵循了大部分的安全建议。更具体地说，他们将"nodeIntegration
Javascript引擎的。同时，Node也提供了许多核心库，用来协助访问文件系统并执行代码。在构建应用程序时，这些库虽然能够带来很大的便利性，但是，从安全角度来说

但是，经过一些测试后，我发现了一个比较简单的有效载荷，竟然可以用来访问Node绑定功能：

```
open('about:blank').open('data:text/html,<script>document.write(process.cwd())</script>')
```

刚开始，我认为这一定是Google Chat开发小组的一个疏忽，但经过一番研究后，我发现事实并非如此，因为，创建的每个窗口都设置了以下属性：

```
win.webPreferences = {
 allowRunningInsecureContent: false,
 contextIsolation: true,
 nodeIntegration: false,
   nativeWindowOpen: true
}
```

经过一段时间的探索，并与Luca
Carettoni（过去曾在Electron方面做过一些非常好的研究）进行多次交流之后，我们更加确信，这其实是Electron框架本身的一个漏洞。我们在一个未合并的PR中也遇到了

看起来，问题在于窗口的属性无法通过嵌套窗口和iframe正确地继承。这意味着任何应用程序，在下列情况下都会生成易受攻击的窗口：

- 用户代码在iframe内运行，或
- 可以创建一个iframe，或
- 如果您使用"nativeWindowOpen：true"或"sandbox：true"选项打开任何窗口

（导致该问题的两个情形原本是用于安全控制的，这的确有点讽刺）

读者可以在这里观看相应的视频。

之后，我们通过适当的渠道通知了各个团队。

Electron团队迅速对这个问题给出了回应，并承诺在几天内提供安全补丁。他们真的很棒！

归根结底，问题的"根本原因"在于不安全的默认设置。在这种情况下，如果使用默认设置，那么窗口就可以访问Node绑定功能，并且不会被隔离。因此，当一个窗口没有按

关于Electron的安全建议

- 请遵循https://github.com/electron/electron/blob/master/docs/tutorial/security.md中的安全准则
- 这些不是默认设置，因此您需要确保这些设置就位。
- 最重要的是，将contextIsolation设置为true，并将nodeIntegration设置为false。
- 如果可以避免的话，请不要在Electron窗口中使用允许用户控制的HTML/Javascript（或XSS）。
- 处理好新建窗口的方式（如果不需要，则阻止新建窗口）
- mainWindow.webContents.on('new-window', e => e.preventDefault())

如何保护企业免受类似的CVE漏洞的威胁

在采用第三方框架时，请务必验证默认的安全设置，因为这些默认值通常不是为安全性而生，而是为性能优化而设置的。

在此，我们要再次感谢Electron团队快速回应，并且及时提供了相应的安全补丁。

上一篇：Windows进程注入技术之PRO... 下一篇：使用 Semmle QL 进行漏洞...

1. 0 条回复
   - 动动手指，沙发就是你的了！