ID：T1189

战术：初始访问

平台：Windows、Linux、MacOS

所需权限：用户

数据源：数据包捕获、网络设备日志、网络进程使用、Web 代理、网络入侵检测系统、SSL/TLS 检查

drive-by compromise

drive-by compromise 是，攻击者通过用户正常浏览过程中访问一个网站而获得对系统的访问权。使用这种技术，用户的 Web 浏览器将成为攻击的目标。这可以通过几种方式实现，但有几个主要组成部分：

多种向浏览器提交漏洞代码的方法，包括：

一个合法的网站被攻陷，被攻击者注入了某种形式的恶意代码（如 javascript、iframes、跨站点脚本）

恶意广告，通过合法的广告提供商支付和提供服务。

内置的 Web 应用程序接口可用于插入任何其他类型的对象，这些对象可用于显示 Web 内容或包含在访问客户端上执行的脚本（例如论坛帖子、评论和其他用户可控制的 Web 内容）。

通常，攻击者使用的网站是特定的人群会访问的网站，例如政府、特定行业或地区，其目标是基于共享的兴趣来危害某一个特定用户或一组用户。这种有针对性的攻击是指一种战略性的网络攻击或水坑攻击。有几个已知的例子说明了这一点。〔1〕

典型 drive-by compromise 过程：

用户访问攻击者控制的网站。

脚本会自动执行，通常根据浏览器和插件的版本搜索易受攻击的版本。

有时候需要用户通过启用脚本或激活网站组件并忽略警告对话框。

一旦发现一个易受攻击的版本，漏洞代码就会被发送到浏览器。

如果攻击成功，那么攻击者的代码将在用户系统上，除非有其他保护措施。

在某些情况下，在执行漏洞代码之前，需要用户在初始扫描后再次访问网站。

与利用面向公共的应用程序不同，此技术的重点是在目标访问网站时利用客户端端点上的软件漏洞。这通常会让攻击者获取访问内部网络系统的权限，而不是 DMZ 中的外部系统。

APT19　2014 年，APT19 在福布斯网站上进行了一次水坑攻击，以破坏目标。

APT32　APT32 通过诱骗受害者访问泄密的水坑网站来感染他们。

缓解

drive-by compromise 依赖于客户端系统上存在易受攻击的软件。使用打开安全功能的现代浏览器。确保所有浏览器和插件都保持更新，可以防止这种技术的漏洞利用阶段。对于通过广告进行攻击的恶意代码，adblocker 可以帮助防止代码执行。脚本阻塞扩展工具可以帮助防护在利用过程中可能常用的 javascript 的执行。
浏览器沙箱可以用来减少一部分漏洞的影响，但沙盒可以逃逸。
其他类型的虚拟化和应用程序微细分也可以减轻客户端开发的影响。额外利用的风险和实施中的弱点可能仍然存在。〔20〕
安全应用程序可以查找在攻击过程中使用的行为，例如 WDEG 和增 EMET,可以缓解某些攻击行为。[21]控制流完整性检查是识别和阻止软件漏洞发生的另一种方法。[22]其中许多防护技术依赖于体系结构和目标应用程序二进制的兼容性。

检测

防火墙和代理可以检查 URL 中潜在的已知坏 domain 或参数。他们还可以对网站和他们请求的资源进行基于声誉的分析，例如域名的历史、注册对象、黑名单中，或者有多少其他用户以前连接过它。

网络入侵检测系统，有时用 SSL/TLS MITM 检查，可以用来查找已知的恶意脚本（Reon，heap spray，和浏览器标识脚本已经被频繁重用），通用脚本混淆，漏洞利用代码。

从合法的网站上检测基于漏洞攻击的失陷可能是困难的。还可以在端点系统上查找可能表明失陷的行为，例如浏览器进程的异常行为。这可能包括写入可疑文件、试图隐藏执行的进程注入证据、可以证明向系统传送了其他工具的异常网络流量

Aomic test:

No

规则：

No