计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2016.20160686
53(10):2163-2172,2016

# HTML5 新特性安全研究综述

张玉清[1,2] 贾 岩[1] 雷柯楠[1] 吕少卿[3] 乐洪舟[1]

[1](综合业务网理论及关键技术国家重点实验室(西安电子科技大学) 西安 710071)

[2](中国科学院大学国家计算机网络入侵防范中心 北京 101408)

[3](陕西省信息通信网络及安全重点实验室(西安邮电大学) 西安 710121)

(zhangyq@nipc.org.cn)

# Survey of HTML5 New Features Security

Zhang Yuqing[1,2], Jia Yan[1], Lei Kenan[1], Lü Shaoqing[3], and Yue Hongzhou[1]

[1](*State Key Laboratory of Integrated Services Networks (Xidian University), Xi'an* 710071)

[2](*National Computer Network Intrusion Protection Center, University of Chinese Academy of Sciences, Beijing* 101408)

[3](*Shaanxi Key Laboratory of Information Communication Network and Security (Xi'an University of Posts and Telecommunications), Xi'an* 710121)

**Abstract** HTML5 is the latest standard of building Web applications. It introduces many new features to browsers, but also brings new security issues. The security of new features is the essence of HTML5 security. According to the differences in function, we analyze and summarize the security of new features including new label and form, communication, offline and storage, multimedia, performance, device access. The security problems and possible prevention methods are pointed out. Then we summarize existing researches, and classify HTML5 security problems into three categories: extending traditional threats, malicious use and improper use, to provide a new thought for the further study of HTML5 security. At last, four directions of the future work are pointed out: the security of new features, detection of malicious use, cross platform security and new security applications.

**Key words** Web security; HTML5; literature review; postMessage; WebSocket; AppCache; WebStorage

**摘 要** HTML5 是构建 Web 应用的最新标准,它引入许多新特性来赋予浏览器丰富的功能,但因此也引入了新的安全问题. HTML5 安全问题实际由各个新特性的安全问题组成. 根据功能差异,对 HTML5 中的标签与表单、通信功能、离线应用与存储、多媒体、性能与表现、设备访问等新特性的安全性进行了详尽的分析、总结和讨论,指出其蕴含的安全问题及可能的防范方法. 然后对现有国内外研究工作进行了总结,进一步将 HTML5 安全问题归纳为 3 类:传统威胁延伸、恶意利用、使用不当,为进一步研究提供了思路. 最后,指出了 HTML5 安全研究未来有价值的 4 个方向:新特性安全性、恶意利用检测、跨平台安全性和新安全应用形式.

收稿日期:2016-08-18;修回日期:2016-09-20

基金项目:国家自然科学基金项目(61272481,61572460);国家发改委信息安全专项[(2012)1424];国家重点研发计划项目(2016YFB0800703);国家 111 项目(B16037);教育部-中国移动科研基金项目(MCM20130431)

This work was supported by the National Natural Science Foundation of China (61272481,61572460), the National Information Security Special Projects of National Development and Reform Commission of China [(2012)1424], the National Key Research and Development Project (2016YFB0800703), China 111 Project (B16037), and the Research Fund of Ministry of Education-China Mobile (MCM20130431).

关键词 Web 安全; HTML5; 综述; postMessage; WebSocket; AppCache; WebStorage

中图法分类号 TP393.08

随着互联网的飞速发展,Web 已经融入生活的方方面面,我们几乎每天都会使用 Web 处理各种事务,如银行业务、网上购物、浏览新闻、电子邮件、社交网络等等.为推动 Web 的标准化及满足应用日益丰富的要求,2007 年万维网联盟 W3C(World Wide Web Consortium)立项 HTML5,并于 2014 年 10 月完成标准化工作[1].它包含了一系列的 HTML 语义标签、JavaScript API、CSS3 等新特性.目前,无论是电脑还是智能终端,互联网的主要浏览器都对新标准提供了良好的支持[2].

HTML5 标准由众多的新特性组成,根据功能的不同,可以分为标签与表单、通信 API、离线应用与存储、多媒体、性能与表现、设备访问 6 类,如表 1 所示.其中,新标签与表单引入了语义信息和丰富的交互事件与属性,使客户端更加灵活;通信 API 提供了浏览器与服务器、浏览器与浏览器、浏览器标签之间新的通信方式,满足多种应用场景下的开发需求;多媒体特性使得浏览器原生支持多种图形、视频、音频,无需额外安装插件;性能与表现特性则使性能显著优化,并提供给 UI 多种友好的表现形式;设备访问特性使得浏览器能够充分的利用多种平台的不同硬件,构建出丰富的应用.

**Table 1 Summary of HTML5 New Features**

**表 1 HTML5 新特性分类一览**

| Type | Features |
|---|---|
| Tag and Form | article, video, email, formaction, pattern, sandbox, autofocus, etc. |
| Communication | WebSocket, postMessage, XHR2 |
| Offline Application | AppCache, IndexedDB, WebStorage |
| Multimedia | WebRTC, SVG, canvas |
| Performance | WebWorkers, SSE, Drag/Drop, history, Notification |
| Device Access | Geolocation, Camera, getUserMedia, Battery, etc. |

虽然 HTML5 提供了丰富的应用形式,但同时也带来了许多新的安全问题,传统的 Web 安全问题如 XSS、CSRF、设备指纹识别、UI 欺骗等在 HTML5 环境下有了新的发展,同时,XMLHttpRequest2、WebStorage、postMessage 等新 API 还引入了新的跨源、本地存储、标签通信风险.主要原因归纳有如下 3 点:1)传统的安全问题会以新的形式在使用了 HTML5 技术的应用中出现,如 HTML5 新标签带来了更多的 XSS 攻击向量;2)HTML5 许多新 API 赋予了 Web 强大的功能,这同时提供给攻击者许多新的攻击方式;3)由于开发者对 HTML5 新的特性不够熟悉,使用时难免会考虑不周,导致应用存在被攻击者利用的漏洞.总的来说,正是由于 HTML5 引入了许多新的特性,所以才产生了诸多新的安全问题,

现今 Web 的广泛应用使得其安全性至关重要,而 HTML5 作为新的 Web 标准,其安全问题直接关系着整个 Web 平台的安全.所以,安全界已经对 HTML5 安全开展了广泛的研究.首先,2010 年 Kuppan[3]在黑帽大会上公布了通过 HTML5 技术进行攻击的系列方法,包括新标签跨站点脚本(cross-site scripting, XSS)、COR 反向代理、点击劫持、应用程序缓存中毒、客户端 RFI、网络扫描、构建僵尸网络等,并同时给出了概念验证程序.这是安全界最早分析 HTML5 安全问题的系统性报告.在 2012 年欧洲黑帽大会上,Shah[4]再次补充了通过 HTML5 技术进行攻击的 10 种攻击手段,包括 CORS、UI 欺骗以及 HTML5 带来的新 XSS 攻击载荷等.国际著名 Web 安全组织 OWASP 针对 HTML5 安全,专门成立工作组并维护了在线 HTML5 安全手册[5],对开发者具有非常高的参考价值.在国内,吴翰清[6]的著作《白帽子讲 Web 安全》中较早地对 HTML5 安全进行了探讨;2013 年,孙松柏等人[7]对 HTML5 安全做了概括性论述,并对国内许多知名站点进行了安全测试,发现 HTML5 安全问题严重影响当前国内 Web 安全.

除对 HTML5 的综合安全研究外,学术界自 2010 年后对 HTML5 的各个特性也进行了深入的研究,发现了 XSS、信息泄露、用户追踪等多方面的安全隐患,并提出相应方案来加强 HTML5 的安全性.

HTML5 安全问题实际由各个新特性的安全问题组成,故本文试图围绕着引入的各类新特性,对 HTML5 安全展开讨论.

## 1 HTML5 新特性安全分析

本节按照新特性功能的不同,分类讨论各个新特性的安全研究现状.由于跨站点脚本的普遍性,并

且其经常能够作为其他攻击的先决条件,故单独列出讨论.另外为保证研究的完整性,在最后补充了HTML5 其他的相关安全问题.

### 1.1 标签与表单

HTML5 规范了 HTML 解析器,定义了许多新的内容,如新的结构标签、音视频标签、MathML 等,并丰富了原有表单、内嵌窗口的属性.

这些新内容除带来跨站点脚本威胁外,还会引入其他的风险. iframe 的 sandbox 属性可以设置是否允许内嵌窗口执行脚本,从而使攻击者进行 UI 欺骗时绕过 FrameBusting 检查. HTML5 对表单功能的加强也引入了许多新的风险,如 autocomplete 属性提供自动完成的功能,使浏览器根据表单 ID 将输入信息保存在本地,方便用户的同时违背了同源策略,产生泄露表单中隐私的风险[8];pattern 属性提供客户端的正则表达式过滤,但开发者仍需在服务端对输入进行验证,疏忽会导致严重的安全风险;formaction 属性可以覆盖 form 的 action 属性,攻击者若能对表单注入该属性,即可操纵表单提交的地址,并且由于不是脚本注入,所以无法受到 CSP 的保护,甚至不受注入位置的限制[9]. Preibusch 等人[10]针对 Web 表单中的许多特性进行了安全性研究.另外,一些标签属性的使用还能够对客户端造成拒绝服务,如〈input *onblur*=*focus*() *autofocus*〉,*focus*()是获取焦点的函数[11].

HTML5 新标签的丰富性导致其带来的安全问题也是多种多样的,它既提供给攻击者手段绕过现有防御措施,也使开发者在使用时易带来新的漏洞,如与原有防御机制相冲突、信息泄露、表单注入、拒绝服务等问题.目前,对新标签的安全性问题通常难以有系统性的分析,研究以打补丁的形式进行,未来仍需研究各个新标签潜在的安全风险,如在不同终端不同浏览器上的不同实现形式.

### 1.2 通信功能

#### 1.2.1 postMessage API

出于安全方面的考虑,运行在同一浏览器中的框架、标签页、窗口间的通信一直都受到了严格的限制,然而,现实中存在一些合理的需求让不同站点的内容能在浏览器内进行交互. HTML5 为跨文档消息机制提供了 postMessage API,用来实现跨框架、标签页、窗口通信.其通过响应事件来接收消息,通过检查消息的来源来决定是否对这条消息进行处理.

2010 年,Hanna 等人[12]发现了新型浏览器在部署 postMessage 技术时客户端通信协议层面出现的安全问题. 2013 年,Son 等人[13]发现了许多实际中 origin 源验证存在的漏洞,由此攻击者可以注入脚本、任意修改本地存储,如 www.ieee.org 等知名的站点均存在该问题,表明开发者在使用新的 API 时非常容易出现问题.因此,李潇宇等人[14]同年提出了一种跨文档消息传递方案 SafePM 来帮助开发者安全使用 postMessage.

#### 1.2.2 WebSocket API

HTML5 引入了 WebSocket 来增加异步通信和跨源通信的支持,使得开发者能更便捷地构建实时应用.但同时,该功能的引入也为攻击者提供了构造各种恶意代码的可能. 2010 年 Kuppan[3]详细讲解了使用 Web-Socket 扫描的技术细节,并发布了一个概念验证工具"JS-Recon",该工具可以通过受害主机浏览器对内网进行扫描. 2012 年,Schema 等人[15]分析了 WebSocket 在 MITM、DoS、IDS、Fingerprinting、Fuzzing、使用不当等多方面的安全问题. 2013 年,Kulshrestha[16]回顾了 WebSocket 的协议和 API,并讨论了不同浏览器面对混合内容和不可靠证书时存在的安全问题.

#### 1.2.3 XMLHttpRequest2(XHR2)

XMLHttpRequest API 使得 Ajax 技术的实现成为了可能,作为其改进版本,XHR2 主要增加了跨源和进度事件 2 方面的功能.其跨源支持使得 CSRF 成为可能,通过将 XHR2 对象的 setRequestHeader 设置为 Content-type,multi-part/form-data,将属性 WithCredentials 设置为 true,就能够对 cookie 进行重放,从而实现攻击[4,7,17]. 同 WebSocket 类似,利用状态响应时间的不同,XHR2 也能够用来进行网络扫描[3].

目前,各界对 postMessage API 和 WebSocket 的安全问题已经进行了较为丰富的研究,并且实际中也已经有了较为广泛的应用.其中,对 postMessage 的研究主要包含了浏览器实现漏洞和开发者安全应用 2 个方面.对 WebSocket 安全的研究既包括了主要的通信功能,也包括了在其他安全问题下的表现和新恶意利用方式. XHR2 是对原有 XHR 的升级改动,研究集中于其跨源新特性所带来的新风险.通信 API 常常涉及到客户端与服务端 2 方面的通信,研究服务端实现方面的安全问题可作为未来方向之一.

### 1.3 离线应用与存储

#### 1.3.1 本地存储 API

WebStorage 是 HTML 新增的本地存储解决方案之一,意图在于解决本来不应该 cookie 做,却不得不用 cookie 的本地存储. 相比于 cookie, WebStorage 减少了通信的流量,也降低了被监听的风险. WebStorage 使用简单字符串键值对在本地存储数据,方便灵活,但是对于大量结构化数据存储力不从心. IndexedDB 能够在客户端存储大量的结构化数据,并且使用索引高效检索,在 2015 年 1 月 8 日正式被 W3C 推荐,而 Web SQL Database 实际上已经被废弃.

本地存储在方便开发者在客户端存储数据的同时,也会遭到同 cookie 类似的攻击,如 XSS 和 CSRF 等. 除此之外,其还有一些其他的安全风险. 如浏览器在早期实现 WebStorage 时存在缺陷,不限制站点的存储容量,攻击者可以借此耗尽浏览器的存储空间[18]. Kimak 等人[19]研究了 IndexedDB 数据如何存储在客户端本地文件系统,并叙述了如何在应用程序删除了客户端数据库后仍然可以获取到数据及解决方案. Matsumoto 等人[20]发现,从浏览器的主内存映像(main memory image)中可以取得 WebStorage 的值信息. Acar 等人[21]在研究中发现,IndexedDB 可以被用来作为用户追踪的新手段.

研究者还提出了对本地存储安全性的改进方案与安全应用场景. 2014 年,Jemel 等人[22]针对本地存储 API 提出了一种安全加强方案,即浏览器为每个用户分别提供安全的存储空间. 然后又从云计算应用的角度,提出了加强本地存储数据安全的方法和不同设备同步本地存储数据的方案[23]. Kimak 等人[24]对 IndexedDB 的发展历程进行回顾,针对已有的安全问题提出了新的安全模型.

#### 1.3.2 应用程序缓存

在全球互联的时代,人们越来越依赖网络连接,但事实上,网络连接中断时有发生,为此 HTML5 引入了应用程序缓存(application cache, AppCache). 使用应用程序缓存,避免了加载应用程序时的常规网络请求,如果缓存清单文件是最新的,浏览器就无需检查其他资源是否最新. 这可以节省带宽,加快访问速度,并减轻服务器负载.

新缓存机制的引入同时带来了新的缓存中毒攻击. Kuppan[3]首先提出利用应用程序缓存作为缓存中毒攻击的新方式,比 HTTP 缓存中毒攻击更加持久. 2013 年,Johns 等人[25]发现使用 AppCache 缓存恶意脚本至 DNS-IP 映射信息过期可以绕过反 DNS Rebinding 机制,从而破坏浏览器的同源策略. 2015 年,Jia 等人[26]研究了应用程序缓存中毒对 HTTPS 的影响,即使用 HTTPS 也难以保证安全. 除此之外,由于实现时缺少安全性的考虑,它还赋予攻击者窃取用户隐私信息的能力. 同年,Lee 等人[27]发现了利用应用程序缓存的事件机制来判断跨源资源状态的方法,从而推断出用户的 Web 访问习惯与认证状态.

目前,本地存储 API 的安全性吸引了众多研究者的目光,包括其客户端实现的缺陷、客户端存储数据的机密性、不同应用场景的隐私风险等. 另外,研究者对于其安全性改进与安全应用场景也提出了各种不同的方案. 应用程序缓存使缓存中毒攻击方式有了新的表现形式,长时间的缓存也给已有的安全机制产生冲突,其实现上的不周也为窃取敏感信息提供了新的手段. 出于安全性的考虑,正在制定的用于离线应用的 Service Workers 仅允许在 HTTPS 连接中使用[28].

### 1.4 多媒体

Web 的功能已经逐渐从浏览简单的网页发展成为可以处理复杂媒体的应用平台. 但是,以前浏览器对多媒体的支持往往是通过各种插件来实现,如著名的 Flash 插件. 现在,HTML5 推出了一系列标签和 JavaScript API,使浏览器原生支持多媒体功能.

网络实时通信(Web real time communication, WebRTC)主要用来让浏览器实时获取和交换视频、音频等数据. getUserMedia ( ) API 还允许通过 JavaScript 脚本分享用户的屏幕,Tian 等人[29]发现使用屏幕共享 API 可以在用户无察觉的情况下窃取用户屏幕上的信息和浏览历史记录,并绕过站点的防御进行 CSRF 攻击.

可缩放矢量图形(scalable vector graphics, SVG)使用 XML 格式定义图像,曾经没有得到浏览器很好的支持,但由于 HTML5 的到来,现可以用传统的标签嵌入页面中. 随着其应用的普及,研究人员发现了许多潜在的安全风险. 2011 年,Heiderich 等人[30]发现使用〈img〉标签和 CSS 嵌入的 SVG 图像可以注入并执行任意 JavaScript 代码,称之为 AII 攻击,并提出相应的防范措施. Stone[31]在 2013 年黑帽大会上提出,使用 SVG filter 来进行计时攻击(timing attack)可以获取历史浏览记录的方法. 2012 年,Heiderich 等人[32]指出,攻击者使用 SVG 的〈set〉标签和 accessKey 事件,可以记录用户在浏览器中输

入的内容,而不必使用脚本.

〈canvas〉标签定义图形,使用脚本来绘制图形,与 SVG 以及 VML 之间的一个重要不同是,〈canvas〉有一个基于 JavaScript 的绘图 API,而 SVG 和 VML 使用一个 XML 文档来描述绘图. Mowery 等人[33]提出使用〈canvas〉绘图作为系统指纹识别的手段,因为绘图功能直接关系到显卡驱动和 GPU,通过检查产生的像素,会发现不同的系统产生不同的输出. 这种用户跟踪的手段独立于其他方法,并且易于使用和对用户透明. 因为 HTML5 给予了浏览器更多访问硬件的能力,所以 Web 现在可以基于硬件来进行设备指纹识别,而这些硬件特征更难被掩饰和更改[34].

目前,多媒体类 API 除了提供 XSS 新的攻击向量外,其中一些事件特性还可被攻击者恶意利用来窃取击键记录和浏览器记录等私密信息. 由于其与硬件设备密切相关的特性,还被挖掘出许多进行设备指纹识别的手段,威胁用户的隐私. 除浏览器的事件、编程接口支持外,多媒体类 API 的实现还需要多方面的支持,如图形绘制、解码库等,其中底层支持的安全性也可作为未来研究方向之一.

### 1.5 性能与表现

HTML5 在 Web 性能方面进行了优化,Web Workers 使 JavaScript 现在可以后台多线程执行. 但同时攻击者可以使用该特性充分利用用户的浏览器隐蔽地构建僵尸网络,进行 DDoS、挖掘比特币、进行暴力破解等[3].

除了性能方面的改进,HTML5 也定义了新的 API,使开发者可以更方便高效地丰富 Web 表现形式. 比如,服务器发送事件(server-sent event, SSE)允许网页获得来自服务器的推送更新;History API 允许 Web 页面操作浏览器的历史记录;拖拽 API 支持用户在浏览器中拖拽元素;Web Notification API 可以向用户随时进行桌面消息推送,不局限于当前页面. Yoon 等人[11]提出,可以使用 SSE 作为命令通信方式的僵尸网络,并且比 Web Workers 构建的僵尸网络更难被检测. 同时,表现形式的丰富也增加了社会工程学攻击的能力. History API 使攻击者可以加入大量的 URL 历史记录,使用户使用浏览器“后退”按钮无法回退;而修改当前显示的 URL 增加了钓鱼攻击的迷惑性[35]. Web Notification 可以被攻击者用来推送欺骗消息,拖拽 API 丰富了 UI 欺骗攻击的形式.

目前对性能与表现新特性的安全研究集中于其自身功能的滥用,即攻击者可以使用这些特性的原有功能来利用客户端的资源或者进行社会工程学攻击等. 但是,对于恶意利用的检测与防范研究较少,并且对该类新特性的实现安全性与其他恶意利用方式还没有发现,这些可作为未来研究方向的重点.

### 1.6 设备访问

HTML5 支持用户通过 Geolocation API 获取地理位置信息,极大地丰富了 Web 应用形式,但不恰当的实现会导致隐私泄露的风险. 2010 年 Doty 等人[36]关注了 Geolocation API 的隐私机制,并提出了一种隐私框架. 同年,Zalewski[37]发现 Geolocation 用户提示可以通过对用户界面进行时差攻击进行 UI 欺骗,从而窃取地理位置信息. 2014 年,Kim 等人[38]深入研究了当前浏览器对 Geolocation API 的实现,发现 14 种存在缺陷的漏洞和 603 个过高权限的站点,并提出了细粒度的权限控制和位置模型,以及对 Geolocation 发展的一些建议. 这些研究均增强了对用户隐私性的保护.

HTML5 也提供了许多对移动设备访问的支持,如照相机、触摸事件等. Yoon 等人[11]提出了使用震动 API 对移动设备拒绝服务的攻击,这种攻击会对用户使用造成不良影响,迫使用户关闭该页面. Olejnik 等人[39]发现 FireFox 实现的显示电池状态 API 时精确度过高,可以被用来进行设备指纹识别.

目前对于 HTML5 设备访问的研究主要集中于用户隐私的攻防,还有移动设备的特殊硬件产生安全隐私隐患. 研究包含了新的恶意利用方式及 API 功能实现时的缺陷,但是缺少结合客户端平台,对 API 功能实现机理安全性的深入研究.

### 1.7 其他方面

#### 1.7.1 XSS 新挑战

XSS 是 Web 上最流行的攻击之一,通过该漏洞攻击者可以向客户端浏览器注入任意可执行脚本. HTML5 引入了许多能够触发 JavaScript 脚本执行的新标签和新属性,如 audio、video、onerror、autofocus 等;除标签与表单外,SVG 和 CSS3 也增加了攻击向量的种类,若网站黑名单过滤没有包含 HTML5 新特性,那么就会产生 XSS 漏洞. HTML5 安全站点[40]维护了一个较为全面的 XSS 攻击向量备忘录;Dong 等人[41]总结了 14 种由 HTML5 引入的 XSS 攻击向量,并针对电子邮件系统开发了相应漏洞检测工具. 而且,HTML5 新特性极大地增强了客户端脚本的能力,增强了攻击危害性.

HTML5 良好的跨平台特性,导致 XSS 有了新的"用武之地". 除了传统上针对浏览器的 XSS 攻击,2014 年,Jin 等人[42]发现移动终端 HTML5 App 同样也存在 XSS 漏洞,并拥有 Contact, SMS, Barcode, MP3 等独特的注入途径. 次年,Chen 等人[43]又发现了针对移动终端 HTML5 App,通过 text box input type 和 document. getElementById("TagID"). value 注入脚本的新方式. 2016 年,Mao 等人[44]提出了通过监控 App 的运行,建立行为状态机来检测 HTML5 App 注入行为的方法.

总之,传统的 XSS 防范方法足以应对新的威胁,但是需要特别提防 HTML5 新引入的注入通道及其跨平台的影响,移动端目前已经发现了较多的安全隐患.

1.7.2 补充工作

一部分研究并不局限讨论 HTML5 某个特性的安全问题,但其主体是信息安全相关问题,并与 HTML5 密切相关,所以特在此节简要说明,以保证研究的完整性.

其中有些研究侧重于 HTML5 的应用. 文献[45-46]讨论了使用 HTML5 特性来加强会话的方案. 文献[47]提出了可以使用 HTML5 的 WebRTC 来进行数据 P2P 安全传输的方案. 文献[48]使用 Web Crypto API 实现了端对端加密的社交网络.

有些研究侧重于整体安全性与安全检测. 文献[49]采用基于行为分析的方法研究了新特性的恶意使用. 文献[7,50]设计并实现了可以检测 HTML5 安全问题的漏洞扫描器. 文献[51]提出了一种针对 HTML5 应用的权限分离方案.

HTML5 应用在移动端的实现也暴露出了新的问题,文献[52]以 Android 和 PhoneGap 为例,研究了移动端操作系统支持 HTML5 应用时的访问控制问题. HTML5 新特性的出现也影响了旧有的安全问题,如文献[53]提出了 3 种基于 HTML5 新特性的混淆技术,使 Web drive-by-download 恶意代码避开现有的安全检测系统.

## 2 现有工作总结

目前安全界已经对 HTML5 安全投入了许多的研究精力,已发表的文献主要有 2 种类型. 第 1 类是针对单个特性进行深入研究,如发现了一类新的恶意利用手段、API 本身实现缺陷、实际中使用时产生的漏洞等,大部分文献属于此类. 通过统计对各新特性研究的 64 篇文献,得到图 1 所示的比例分布图.

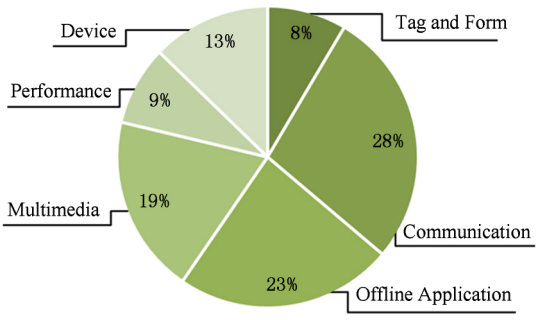



Fig. 1 Literatures distribution of new features security.

图 1 新特性安全研究文献分布比例图

其中,通信 API 和离线应用与存储得到了较多研究者的关注. 这一方面是由于其特性本身的复杂性,如通信 API 的跨源特性、协议、事件机制,本地存储客户端的实现,应用程序缓存的持久特性和事件机制均引入了新的安全风险;一方面由于其应用的广泛性,如 postMessage 的众多实际应用中对源验证存在漏洞. 多媒体类特性也较为复杂,涉及到脚本接口和与硬件密切相关的特性,同样产生了较多的安全问题. 性能表现与设备访问类特性由于其本身特点,研究主要集中于基于社会工程学的隐私泄露. 标签与表单常常引入新的注入攻击,攻击思路较为固定,防范方法较为成熟,已经得到了业界较多的关注.

第 2 类文献概括多点 HTML5 安全问题,包括总体安全分析、安全使用的建议和对以往问题的总结、安全测试等. 如对 HTML5 整体进行安全性分析的部分文献中,文献[7]通过对国内外研究工作的总结分析,将 HTML5 安全问题分为 3 类:HTML5 安全漏洞、HTML5 新型攻击机制、HTML5 新特性滥用. 文献[11]根据攻击产生的危害,将安全问题分为数据泄露、信息控制、请求伪造、拒绝服务、社会工程学 5 类,注重攻击的表现形式,将安全问题划入现有的攻击类型. 上面的分类均不是站在漏洞产生原因的角度进行分析.

经过调研分析,为提供进一步漏洞挖掘的思路,从漏洞产生原因的角度,可以将 HTML5 安全的问题归为 3 类. 1)传统威胁延伸,即传统的 Web 攻击思路延伸到 HTML5 应用之中,对原有防御机制带来了挑战或产生了新的表现形式;如新标签引入的 XSS 对黑名单过滤产生了挑战,移动端 HTML5 App 发现的 XSS 等. 该类问题重点关注新攻击与传统攻击实施思路上的相似性. 2)新功能的恶意利用,即新特性在提供开发便利的同时,也给攻击者提供了新的攻击手段,其可以是对功能的正常使用,如利

用 Web Workers 构建僵尸网络;也可以是主要功能之外的其他手段,类似密码破解中的边信道,如使用 WebSocket 进行网络扫描是利用响应时间的侧信道.该类问题重点关注新特性本身带来的风险.3)使用新特性不当产生漏洞,即开发者在使用新特性时,由于对新功能不够熟悉,疏忽大意导致编程存在安全漏洞;如使用 CORS 时缺少对源的校验和数据验证等.该类问题产生的原因在于开发者使用不当,而非新特性本身.

按照上述分类方法,各个新特性存在的安全问题汇总如表 2 所示,供安全研究人员参考.表 2 中直接给出了关键短语描述,空格表示目前没有发现此类安全问题,具体内容可参考上文对各类特性的详细讨论.注意,HTML5 使得 XSS 攻击更具威胁性,并且 XSS 是很多攻击的前提条件,故统计时不再重复考虑,通常对新标签的攻击向量使用传统方法加以完善即可防范.另外,其中有些漏洞已经随着标准化进程修复.

**Table 2 Summary of HTML5 Security Issues**

**表 2 HTML5 安全问题分类总结**

| Types | Features | Extending Traditional Threats | Malicious Use | Improper Use |
|---|---|---|---|---|
| Tag and Form | | new XSS,form injection,sandbox | DoS | pattern,autocomplete |
| Communication | postMessage | | | origin check |
| | WebSocket | | network scan,DoS | info leakage |
| | XHR2 | CSRF | network scan | |
| Offline | Storage | XSS | user track | |
| | AppCache | cache poisoning | | |
| | DNS-rebinding | status leak | | |
| Multimedia | WebRTC | | info leakage | |
| | SVG | inject XSS | timing attack,key logger | |
| | Canvas | | device fingerprinting | |
| Performance | Web Workers | | botnet | |
| | SSE | | botnet | |
| | History API | phish | DoS | |
| | Notification | | phish | |
| | Drag API | UI phish | | |
| Device Access | Geolocation API | | location leakage,timing attack | improper privilege |
| | Vibration API | | | DoS |
| | Battery API | | device fingerprinting | |

## 3 未来研究展望

HTML5 标准体系目前已经基本完成,其相关安全研究也开展了多年,产生了一大批漏洞和科研成果.未来的研究方向主要包含 4 个方面:

1) 新特性安全性分析

虽然安全研究伴随着 HTML5 整个标准化过程,但是对于其新特性本身的研究却仍然不够深入和全面,主要原因有 2 个方面:①尽管研究者已经对许多新特性进行了研究,但是随着 Web 的快速发展,不断有新的特性被提出,如 2016 年 4 月 19 日 W3C 正式推荐了 Web Storage 第 2 版,继续推出的新特性在带来易用性和修补安全缺陷的同时,难免还会带来新的安全风险,需要安全研究人员不断地跟进研究.②仍有许多特性的安全性没有被详尽地分析,已分析过的特性也无法避免会出现其他的恶意利用方式,如通信 API 的服务端实现、多媒体 API 的底层支持等.

各个新特性的加入使浏览器成为越来越复杂的应用载体,故对各个新特性潜在安全性的研究仍然是未来研究的重点方向,包括并不局限于隐私保护、与旧有安全手段冲突、新恶意利用方式、不同平台实现差异等.本文提出的分类方法给发现新的安全

问题提供了参考思路.

2) 新特性恶意利用检测

目前大部分研究局限于发现一个问题,解决一个问题,部分研究关注现有漏洞的检测,均缺少对普遍 HTML5 恶意利用的检测和防御方案. 具体可以考虑恶意行为建模分析、特征提取,机器学习等手段,参考入侵检测系统的常用方法.

3) 跨平台安全性

W3C 致力于统一各个平台的 Web 标准,以方便开发出功能更加丰富的 Web 应用程序. 然而不同平台、不同终端对标准的实现却不尽相同,这就留下了出现安全问题的隐患. 移动互联网和物联网的蓬勃发展使得 HTML5 应用日益广泛. Web 已经不单单是传统 PC 的应用,许多智能设备都在使用 Web 技术[54],物联万维网(Web of Things, WoT)正被提上议题[55]. 众多的智能终端无疑会存在更多的安全风险,如设备访问 API 在不同硬件特性的平台上会带来不同的隐患. 目前,安卓平台 HTML5 App 已经发现存在 XSS 攻击,其他类型的 Web 攻击也同样可能影响各个平台. 所以不同平台 HTML5 应用的安全性研究也是未来的方向之一,如 HTML5 在移动终端、Web TV 等平台的安全、隐私保护、可达性及恢复力等.

4) 新安全应用形式

HTML5 是新一代的 Web 标准,浏览器对其进行了充分的支持,对 HTML5 新的应用形式也在不断探索之中. 比如可以考虑利用 HTML5 来更加高效地解决如盗链等传统问题,结合多因素构建 Web 认证支付系统,设计浏览器加密通信系统,加强权限控制会话安全. 故利用 HTML5 技术结合具体应用场景需求来构建安全应用也是未来的研究方向之一.

## 4 结束语

Web 技术已经深入人们日常生活的方方面面,其中对 HTML5 安全性的研究具有重要的现实意义. 本文首先对国内外研究现状进行了总结,然后根据功能的差异,讨论了 HTML5 各类新特性的安全问题. 并将已有 HTML5 安全问题归纳为 3 类:传统威胁延伸、新恶意利用方式、使用和实现时的漏洞,为进一步发现新的安全问题提供了思路. 最后,讨论了值得进一步深入研究的安全问题,包括新特性安全性、跨平台应用、新安全应用形式 3 个方向.

## 参 考 文 献


[1] W3C. HTML5 [EB/OL]. [2016-03-19]. https://www.w3.org/TR/html5/

[2] Leenheer N. How well does your browser support HTML5 [EB/OL]. (2016-06) [2016-08-08]. http://html5test.com

[3] Kuppan L. Attacking with HTML5 [EB/OL]. Blackhat 2010. (2010-10-08) [2016-08-08]. https://media.blackhat.com/bh-ad-10/Kuppan/Blackhat-AD-2010-Kuppan-Attacking-with-HTML5-wp.pdf

[4] Shah S. HTML5 Top 10 Threats Stealth Attacks and Silent Exploits [EB/OL]. Blackhat 2012. [2016-08-08] http://media.blackhat.com/bh-eu-12/shah/bh-eu-12-Shah_HTML5_Top_10-WP.pdf

[5] OWASP. HTML5 Security Cheat Sheet [EB/OL]. [2016-03-20] https://www.owasp.org/index.php/HTML5_Security_Cheat_Sheet

[6] Wu Hanqing. White Hat to Talk about Web Security [M]. Chinese Edition. Beijing: Publishing House of Electronics Industry, 2012 (in Chinese)
(吴瀚清. 白帽子讲 Web 安全[M]. 北京: 电子工业出版社, 2012)

[7] Sun Songbai, Ali Abbasi, Zhuge Jianwei, et al. Reseach on HTML5 security [J]. Computer Applications and Software, 2013, 30(3): 1-6 (in Chinese)
(孙松柏, Ali Abbasi, 诸葛建伟, 等. HTML5 安全研究[J]. 计算机应用与软件, 2013, 30(3): 1-6)

[8] andlabs. POC for Stealing Auto-Complete Suggestions from Google Chrome [EB/OL]. [2016-08-08] http://www.andlabs.org/hacks/steal_autofill.html

[9] De Ryck P, Desmet L, Piessens F, et al. A security analysis of emerging Web standards-Extended version, CW622 [R/OL]. Belgium: Department of Computer Science, KU Leuven, 2012 [2016-08-08]. https://lirias.kuleuven.be/bitstream/123456789/349398/1/CW622.pdf

[10] Preibusch S, Krol K, Beresford A R. The privacy economics of voluntary over-disclosure in Web forms [G] //The Economics of Information Security and Privacy. Berlin: Springer, 2013: 183-209

[11] Yoon S, Jung J H, Kim H K. Attacks on Web browsers with HTML5 [C] //Proc of the 10th Int Conf for Internet Technology and Secured Trans (ICITST). Piscataway, NJ: IEEE, 2015: 193-197

[12] Hanna S, Chul E, Shin R, et al. The Emperor's new APIs: On the (In) secure usage of new client-side primitives [C/OL] //Proc of W2SP'10. Piscataway, NJ: IEEE, 2010 [2016-06-15]. http://www.ieee-security.org/TC/W2SP/2010/papers/p03.pdf

[13] Son S, Shmatikov V. The postman always rings twice: Attacking and defending postMessage in HTML5 Websites [C] //Proc of NDSS'13. Rosten, VA, USA: Internet Society, 2013: 1-14

[14] Li Xiaoyu, Zhang Yuqing, Liu Qixu, et al. Secure cross document messaging scheme based on HTML5 [J]. Journal of Graduate University of Chinese Academy of Sciences, 2013, 30(1): 124-130 (in Chinese)
(李潇宇, 张玉清, 刘奇旭, 等. 一种基于 HTML5 的安全跨文档消息传递方案[J]. 中国科学院大学学报, 2013, 30(1): 124-130)

[15] Schema M, Shekyan S, Toukharian V. Hacking with WebSockets [EB/OL]. [2016-08-08] BlackHat USA 2012. http://www.hakim.ws/BHUSA12/materials/Briefings/Shekyan/BH_US_12_Shekyan_Toukharian_Hacking_Websocket_Slides.pdf

[16] Kulshrestha A. An empirical study of HTML5 Websockets and their cross browser behavior for mixed content and untrusted certificates [J]. International Journal of Computer Applications, 2013, 82(6): 13-18

[17] Wang Xiaoqiang. Research of Csrf Attack and Defense Techniques Based on HTML5 [D]. Chengdu: University of Electronic Science and Technology of China, 2013 (in Chinese)
(王晓强. 基于 HTML5 的 CSRF 攻击与防御技术研究[D]. 成都: 电子科技大学, 2013)

[18] Feross A. Introducing the HTML5 Hard Disk Filler™ API [EB/OL]. [2016-04-04]. http://feross.org/fill-disk/

[19] Kimak S, Ellman J, Laing C. Some potential issues with the security of HTML5 indexedDB [C] //Proc of System Safety and Cyber Security. Piscataway, NJ: IEEE, 2014: 495-502

[20] Matsumoto S, Sakurai K. Acquisition of evidence of Web storage in HTML5 Web browsers from memory image [C] //Proc of the 9th Asia Joint Conf on Information Security. Piscataway, NJ: IEEE, 2014: 148-155

[21] Acar G, Eubank C, Englehardt S, et al. The Web never forgets: Persistent tracking mechanisms in the wild [C] //Proc of ACM Conf on Computer and Communications Security. New York: ACM, 2014: 674-689

[22] Jemel M, Serhrouchni A. Security enhancement of HTML5 Local Data Storage [C] //Proc of 2014 Int Conf and Workshop on the Network of the Future (NOF). Piscataway, NJ: IEEE, 2015. Doi: 10.1109/NOF.2014.7119784

[23] Jemel M, Serhrouchni A. Content protection and secure synchronization of HTML5 local storage data [C] //Proc of the 11th Consumer Communications and Networking Conf (CCNC). Piscataway, NJ: IEEE, 2014: 539-540

[24] Kimak S, Ellman J. The role of HTML5 IndexedDB, the past, present and future [C] //Proc of the 10th Int Conf for Internet Technology and Secured Trans. Piscataway, NJ: IEEE, 2015: 379-383

[25] Johns M, Lekies S, Stock B: Eradicating DNS rebinding with the extended same-origin policy [C] //Proc of the 22nd Usenix Conf on Security. Berkeley, CA: Usenix, 2013: 621-636

[26] Jia Y, Chen Y, Dong X, et al. Man-in-the-browser-cache: Persisting HTTPS attacks via browser cache poisoning [J]. Computers & Security, 2015, 55: 62-80

[27] Lee S, Kim H, Kim J. Identifying cross-origin resource status using application cache [C/OL] //Proc of NDSS'15. San Diego, CA: Internet Society, 2015 [2016-06-15]. http://www.internetsociety.org/sites/default/files/01_1_2.pdf

[28] W3C. Service Workers W3C Working Draft 25 June 2015 [EB/OL]. [2016-07-27]. https://www.w3.org/TR/service-workers/

[29] Tian Y, Liu Y C, Bhosale A, et al. All your screens are belong to us: Attacks exploiting the HTML5 screen sharing API [C] //Proc of the 2014 IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2014: 34-48

[30] Heiderich M, Frosch T, Jensen M, et al. Crouching tiger-hidden payload: Security risks of scalable vectors graphics [C] //Proc of the 18th ACM Conf on Computer and Communications Security. New York: ACM, 2011: 239-250

[31] Stone P. Pixel perfect timing attacks with HTML5 [EB/OL]. Blackhat, 2013. [2016-03-20]. http://media.blackhat.com/us-13/US-13-Stone-Pixel-Perfect-Timing-Attacks-with-HTML5-WP.pdf

[32] Heiderich M, Niemietz M, Schuster F, et al. Scriptless attacks: Stealing more pie without touching the sill [C] //Proc of the ACM Conf on Computer and Communications Security. New York: ACM, 2012: 760-771

[33] Mowery K, Shacham H. Pixel perfect: Fingerprinting canvas in HTML5 [C/OL] //Proc of W2SP'12. Piscataway, NJ: IEEE, 2012 [2016-06-15]. http://www.ieee-security.org/TC/W2SP/2012/papers/w2sp12-final4.pdf

[34] Nakibly G, Shelef G, Yudilevich S. Hardware fingerprinting using HTML5 [DB/OL]. ArXiv e-prints, 2015. [2016-08-08]. http://xueshu.baidu.com/s?wd=paperuri%3A%280758be07676e75354f70444de749a4e6%29&filter=sc_long_sign&tn=SE_xueshusource_2kduw22v&sc_vurl=http%3A%2F%2Farxiv.org%2Fabs%2F1503.01408&ie=utf-8&sc_us=1838650899336679323O

[35] TrandLabs. HTML5 Overview [EB/OL]. [2016-04-16]. http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/reports/rpt_html5-attack-scenarios.pdf

[36] Doty N, Mulligan D K, Wilde E. Privacy issues of the W3C Geolocation API [DB/OL]. ArXiv e-prints, 2010. [2016-08-08]. http://xueshu.baidu.com/s?wd=paperuri%3A%286143a2a92f4fbaa43397ca8e34d3bf0b%29&filter=sc_long_sign&tn=SE_xueshusource_2kduw22v&sc_vurl=http%3A%2F%2Farxiv.org%2Fabs%2F1003.1775&ie=utf-8&sc_us=9799738829075159532

[37] Zalewski M. Geolocation spoofing and other UI woes [EB/OL]. (2010-08-17)[2016-03-20]. http://seclists.org/bugtraq/2010/Aug/201

[38] Kim H, Lee S, Kim J. Exploring and mitigating privacy threats of HTML5 geolocation API [C] //Proc of the 30th Annual Computer Security Applications Conf (ACSAC'14). New York: ACM, 2014: 306-315

[39] Diaz C, Olejnik L, Acar G, et al. The leaking battery: A privacy analysis of the HTML5 Battery Status API [J]. IACR Cryptology ePrint Archive, 2015: 616

[40] cure53. HTML5 Security Cheatsheet [EB/OL]. [2016-04-04]. http://html5sec.org/

[41] Dong G, Zhang Y, Wang X, et al. Detecting cross site scripting vulnerabilities introduced by HTML5 [C] //Proc of the 11th Int Joint Conf on Computer Science and Software Engineering (JCSSE). Piscataway, NJ: IEEE, 2014: 319-323

[42] Jin X, Hu X, Ying K, et al. Code injection attacks on HTML5-based mobile Apps: Characterization, detection and mitigation [C] //Proc of the ACM Conf on Computer and Communications Security. New York: ACM, 2014: 66-77

[43] Chen Y L, Lee H M, Jeng A B. DroidCIA: A novel detection method of code injection attacks on HTML5-based mobile Apps [C] //Proc of the 2015 IEEE Trustcom/BigDataSE/ISPA. Piscataway, NJ: IEEE, 2015: 1014-1021

[44] Mao Jian, Wang Ruilong, Chen Yue, et al. Detecting injected behaviors in HTML5-based Android applications [J]. Journal of High Speed Networks, 2016, 22(1): 15-34

[45] Kumar V. Three Tier Verification Technique to foil session sidejacking attempts [C] //Proc of Asian Himalayas Int Conf on Internet. Piscataway, NJ: IEEE, 2011: 1-4

[46] Unger T, Mulazzani M, Hwirt D, et al. SHPF: Enhancing HTTP(S) session security with browser fingerprinting [C] //Proc of the 2013 Int Conf on Availability, Reliability and Security. Piscataway, NJ: IEEE, 2013: 255-261

[47] Farina J, Scanlon M, Kohlmann S, et al. HTML5 zero configuration covert channels: Security risks and challenges [C] //Proc of the 10th ADFSL Conf on Digital Forensics. Security and Law (ADFSL 2015). Florida: ADFSL, 2015: 135-150

[48] Barenghi A, Beretta M, Federico A D, et al. Snake: An end-to-end encrypted online social network [C] //Proc of IEEE Int Conf on High PERFORMANCE Computing and Communications. Piscataway, NJ: IEEE, 2014: 763-770

[49] Choo H L, Oh S, Jung J, et al. The Behavior-Based Analysis Techniques for HTML5 Malicious Features [C] //Proc of the Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS). Piscataway, NJ: IEEE, 2015: 436-440

[50] Wu Q, Liu X. Research and design on Web application vulnerability scanning service [C] //Proc of the 5th IEEE Int Conf on Software Engineering and Service Science (ICSESS). Piscataway, NJ: IEEE, 2014: 671-674

[51] Akhawe D, Saxena P, Song D. Privilege separation in HTML5 applications [C] //Proc of the 21st USENIX Conf on Security Symp. Berkeley, CA: USENIX, 2012: 23-23

[52] Jin X, Wang L, Luo T, et al. Fine-grained access control for HTML5-based mobile applications in Android [G] //LNCS 7807. Berlin: Springer, 2015: 309-318

[53] Santis A D, Maio G D, Petrillo U F. Using HTML5 to prevent detection of drive-by-download Web malware [J]. Security & Communication Networks, 2015, 8(7): 1237-1255

[54] Dujlovic I, Duric Z. Cross-platform Web based real-time communication in Web TV and video on demand systems [C] //Proc of the 57th Int Symp ELMAR. Piscataway, NJ: IEEE, 2015: 65-68

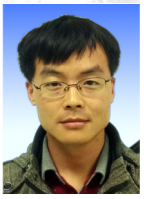
[55] W3C. Web of Things at Industry of Things World [EB/OL]. (2016-03-08)[2016-04-23]. https://www.w3.org/blog/2016/03/w3c-web-of-things-at-industry-of-things-world/

**Zhang Yuqing**, born in 1966. PhD. Professor in the University of Chinese Academy of Sciences. His research interests include network and information system security.

**Jia Yan**, born in 1992. PhD candidate in Xidian University. His research interests include network and system security.

**Lei Kenan**, born in 1992. Master candidate in Xidian University. Her main research interests include network and information security.

**Lü Shaoqing**, born in 1987. PhD and assistant professor. His main research interests include social networks security and data mining.



**Yue Hongzhou**, born in 1987. PhD candidate in Xidian University, accepting joint training in National Computer Network Intrusion Protection Center, University of Chinese Academy of Sciences. His main research interests include network and information security.