华章科技
ELSEVIER 爱思唯尔

信息安全技术丛书

Web应用漏洞侦测与防御

揭秘鲜为人知的攻击手段和防御技术

[美] Mike Shema 著　　齐宁 庞建民 张铮 单征 等译

Hacking Web Apps

Detecting and Preventing Web Application Security Problems

- 国际知名网络安全专家亲笔撰写，全面揭示Web应用常见安全漏洞及应对策略，Amazon广受好评。
- 从浏览器安全和站点安全的角度，全面、系统解读黑客攻击和漏洞利用的技术细节和方法，以及防御这些攻击的最佳方式。

机械工业出版社
China Machine Press

信息安全
技术丛书

Web应用漏洞侦测与防御

揭秘鲜为人知的攻击手段和防御技术

Hacking Web Apps

Detecting and Preventing Web Application Security Problems

[美] Mike Shema 著　齐宁 庞建民 张铮 单征 等译

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

Web 应用漏洞侦测与防御：揭秘鲜为人知的攻击手段和防御技术 /（美）希马（Shema, M.）著；齐宁等译 . —北京：机械工业出版社，2014.8
（信息安全技术丛书）
书名原文：Hacking Web Apps：Detecting and Preventing Web Application Security Problems

ISBN 978-7-111-47253-7

I. W… II. ①希… ②齐… III. 计算机网络－安全技术 IV. TP393.08

中国版本图书馆 CIP 数据核字（2014）第 148495 号

本书版权登记号：图字：01-2013-6488

Web 应用漏洞侦测与防御：
揭秘鲜为人知的攻击手段和防御技术

[美] Mike Shema 著

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码：100037）
责任编辑：吴 怡 责任校对：殷 虹
印 刷：北京市荣盛彩色印刷有限公司 版 次：2014 年 8 月第 1 版第 1 次印刷
开 本：186mm × 240mm 1/16 印 张：15.5
书 号：ISBN 978-7-111-47253-7 定 价：69.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88378991 88361066 投稿热线：（010）88379604
购书热线：（010）68326294 88379649 68995259 读者信箱：hzjsj@hzbook.com

Forword 译 者 序

当前，随着智慧城市、物联网、大数据等社会信息化进程的推进，我国政府、金融、海关、交通、能源和现代服务及工业控制等关键领域均实现了信息化，但不容忽视的现实是，这些关键领域信息基础设施或设备却几乎都被外国少数公司的产品垄断，而且绝大部分核心元器件、高端芯片、基础工具软件等也都严重依赖进口，网络空间安全形势非常严峻。

网络空间的攻击技术也在迅速发展，且已进入自动化程度高、攻击速度快、攻击工具复杂、潜伏更隐蔽的高级阶段。网络犯罪、网络恶意攻击几乎时时刻刻都在上演，而导致这些问题发生的最大安全隐患正来自于网络的核心：Web 应用程序。

本书的每一章都针对某类攻击方式，不仅探究其攻击方法、机理及影响，而且还给出可能采取的应对措施。通过对本书的阅读和理解，有助于了解如何对 Web 应用从多角度实施安全保护措施，做到防患于未然，使得网站拥有者或网站开发人员有信心面对安全相关的威胁与挑战。

本书由齐宁、庞建民、张铮、单征进行主要章节的翻译，杨劲、王俊超、于锦涛、党林玉、薛飞也参与了本书的部分翻译工作。

本书作者是信息安全领域的领军人物之一。虽然我们付出了许多努力，查阅了大量参考资料，力求在充分理解的基础上尽可能准确地完成翻译，为读者奉献一本优秀的译著，但由于书中涉及的知识和技术范围很广，错误或不当之处仍难避免，敬请读者和同仁谅解，诚挚希望读者发现错误时使用电子邮件与我联系：qining2005@126.com。

齐宁

2014 年 4 月

前　言 Preface

关于 Web 的格言或警句中，你最喜欢哪一个？这些格言或警句很可能会涉及 Web 安全或网站所面临的威胁。本书通过黑客最经常利用的 8 组安全弱点及漏洞，试着阐明复杂难解的 Web 安全性问题。对读者来说，其中一些攻击可能很熟悉，而另外一些攻击可能是出乎意料或者比较陌生的，因为这些攻击并未登上头条新闻或进入到前十大新闻中。攻击者可能会针对一些底层公共特性进行利用，这也是为什么诸如跨站脚本攻击和 SQL 注入攻击引起了人们广泛关注的原因。更厉害的攻击者可能会把网站设计的工作流或假设中的模糊之处当作目标，利用这些结果会获得巨大经济收益，但可能只对某个网站有效，这种方式的优势在于不会像 SQL 注入等攻击那样暴露出攻击者的信息。

在 Web 上，信息就是金钱。信用卡显然对黑客有巨大的价值，不断出现非法信用卡交易网站对盗来的信用卡通过论坛、用户反馈和卖家评价等进行交易。我们的个人信息、密码、电子邮件帐户、线上游戏账户等都有其价值，更不用说我们尽力使这些内容保密所付出的努力的价值。目前的现实就是经济间谍以及国家资助的网络攻击被人们普遍关注和夸大，其实更缺乏的是可靠的公共信息。（对 Web 安全而言，“网络战争”存在与否无关紧要，我们更加关注的是如何应对这些。）真实世界中人、公司或国家之间的几乎所有骗局、欺骗、诡计等在 Web 上都存在，因为非法获取 Web 上有价值的信息很诱人，可以是为了荣誉、国家、金钱，或纯粹出于好奇心。

本书学习要点

本书的每一章都给出了针对 Web 应用程序的不同攻击实例。首先会探究这些攻击所采用的方法，然后展示这些攻击潜在的影响，这些影响可能是针对网站的安全性或用户的隐私。攻击甚至可能不会将重点放在攻破某个 Web 服务器，而是聚焦于攻击浏览器。Web 安全性对应用程序和浏览器的影响是类似的，毕竟，它们都是存放信息的地方。

每章中接下来的内容会从攻击的不同角度给出相应的应对措施。应对措施是很棘手的事物，在设计出好的防御措施之前，必须要理解攻击的工作方式，还要清楚这些应对措施的局限性以及不能覆盖的漏洞。安全是网站的整体属性，不是单个保护措施的累加。本书中某些应对措施会被反复提到，有些应对措施可能仅出现一次。

本书面向的读者

使用 Web 收取电子邮件、购物或工作的人都将会从本书受益，他们可以了解网站如何泄露个人信息以及怎样隐藏恶意内容。网站开发人员肩负着网站安全性的最大责任，同时用户也有自己的责任。用户的责任主要体现在维护最新的浏览器、注意密码的使用、警惕类似社会工程这类非技术攻击。

Web 应用程序开发人员和安全专家将会从本书介绍的 Web 攻击背后的技术细节和方法中受益。提升网站安全性的第一步就是要能够理解应用程序面临的威胁，理解不好的编程习惯会导致安全弱点，该弱点会导致漏洞，而漏洞会导致数百万用户密码从未加密的数据库中泄漏出去。另外，有几章深入探讨了与编程语言或支撑特定站点的技术无关的有效应对措施。

行政管理层能够从本书中理解 Web 站点面临的威胁，而且会看到在很多案例中，仅仅需要一个浏览器并动动脑筋，这种简单攻击就能对站点和它的用户带来负面影响。这同样说明尽管很多攻击易于执行，但好的应对措施仍需时间和资源才能得到正确实现。这些要点为分配资金和资源以提高网站安全性，进而保护 Web 站点管理的大量信息财富提供有力的论据。

本书假定读者对 Web 具有基本的了解。Web 安全攻击主要依赖于通过操纵 HTTP 数据包来注入有效负载或利用协议中的缺陷。攻击者同样需要掌握 HTML 来操纵表单或注入代码，从而控制浏览器。这并不是粗略理解攻击或学习黑客如何攻破某个网站的先决条件。例如，一开始只要熟悉 HTTP 协议默认使用 80 端口进行不加密传输，安全套接层（SSL）或安全传输层协议（TLS）协议使用 443 端口进行加密传输，使用 https:// 的站点指定 TLS 传输。如果开发人员或安全专家希望对攻击和防御的方法进行深入研究，则必须了解更多技术细节。本书力求呈现准确的信息，但并不追求严格地区分使用只有细微差别的术语，如 URL 同链接可以互换使用，Web 站点和 Web 应用程序也是如此。但愿对破解的概念以及应对措施的描述足够清楚，习惯于阅读标准和规范的人不会对本书不加区分地引用“HTML 标签”和“HTML 元素”等术语感到不满。我们的目的是了解破解并从学习中感到快乐。

已经熟悉 Web 基本概念的读者可以跳过下面的两节。

现代浏览器

在本书中很少提及具体的浏览器版本。最主要的原因是绝大多数的攻击者通过标准的HTML进行攻击，或只是针对与浏览器无关的服务器端技术。缓冲区溢出攻击和恶意软件关注浏览器的具体版本，但针对Web站点的攻击很少关注这一点。另一个原因是浏览器开发人员一般都采取了自动升级方法，或至少有非常快的发布流程，意味着浏览器基本能够保持最新状态，这对用户的安全而言是一个有利的趋势。最后，就像我们在第1章中将要介绍的，HTML5仍旧是一个新兴标准。在本书中，“现代浏览器”是指任何支持HTML5部分特性的浏览器或呈现引擎（记住，可以通过各类设备访问HTML）。可以肯定地说，当你阅读本书时，如果你的浏览器在最近两个月内进行了更新，那么它就是现代浏览器，如果浏览器是一年前的，它也很有可能是现代浏览器，但如果超过一年，那就放下书去更新吧，这对你会有帮助。

曾几何时，开发Web应用程序时，程序员还要考虑不同浏览器的市场份额或必须考虑浏览器呈现中的怪异模式，这已经成为了过去。类似IE6这种已经“逝去”的浏览器仍然能够呈现当今的大部分网站，是工程设计及标准（网络、HTTP、HTML等）值得称道的一个壮举。然而，已经没有必要在今天继续使用这些老古董了。如果微软希望IE6消失，网站就没有理由再乐于支持它，事实上，对于那些内容和使用都要求更高程度的安全性及隐私保护的网站，应当主动拒绝过时的浏览器的访问。

“域”搞定一切

Web浏览器在不同的平台上已经经过数次的更新换代：Konqueror、Mosaic、Mozilla、Internet Explorer、Opera、Safari等。浏览器的核心中有一个呈现引擎，微软IE的引擎为Trident，Safari和Chrome的引擎为Webkit，Firefox的引擎为Gecko，Opera的引擎为Presto。这些引擎负责把HTML呈现为DOM（Document Object Model，文档对象模型）、执行JavaScript、提供Web页面布局，最终提供安全的浏览体验。

同源策略（Same Origin Policy，SOP）是浏览器的基本安全边界。内容的能力和可视性受限于最初加载资源的源内。不像低成本恐怖电影中，来自某个区域的恶魔会在另一个区域中肆虐，浏览的上下文应当从它建立起就受限于其所在的域了。域是为浏览上下文获取资源的协议、主机名（host）和端口号（port）的组合。我们将会多次讲到SOP，第1章先介绍HTML5如何放宽SOP的限制。

所需背景知识

本书远不能详细地涵盖相关主题。多种攻击方法和应对措施都涉及密码学，例如哈希、

盐值、对称加密、随机数等。有一些章节涉及数据结构、编码、算法等知识，还有些章节涉及正则表达式。这些概念会描述得足够清楚，即使你第一次接触它们，也应该可以知道它们同黑客攻击及应对措施的关联。当需要更多背景知识时，会给出一些建议阅读的资料。本书应当会引起你对此类主题的好奇。一名好的安全从业者或 Web 开发人员应当对很多主题都很熟悉，即便对更深的数学或理论细节可能还有些模糊。

对本书而言，最重要的安全工具是 Web 浏览器，它往往是对 Web 站点进行攻击所需要的唯一工具。Web 应用程序漏洞利用所涉及的技术即包括复杂的缓冲区溢出，也包括对 URI 中单个字符的操纵。第二重要的工具是能够发送原始 HTTP 请求的工具，下面的这些工具可以对浏览器做极好地补充。

Netcat 是网络安全工具的鼻祖，它执行一个基本的功能，即打开网络 socket。该命令的威力在于可以向 socket 发送任何内容并且捕获其响应。多数 Linux 系统或 OS X 默认提供该工具，通常作为 nc 命令。它在 Web 安全中的最简单用法如下所示：

```
echo -e "GET/HTTP/1.0"|netcat -v mad.scientists.lab 80
```

Netcat 的缺点之一是不支持 SSL 协议。幸运的是，OpenSSL 命令提供了相同的功能，只需对命令行进行略微改动。例子如下：

```
echo -e "GET/HTTP/1.0"|openssl s_client -quiet -connect mad.scientists.
   lab:443
```

同命令行工具相比，本地代理提供了一个用户界面更加友好的 Web 安全评估方案。命令行便于自动化，但代理更有助于分析 Web 站点的弱点并理解 Web 请求背后所发生的事情。

风险、威胁、弱点、漏洞、漏洞利用

一些读者可能会注意到本书刻意避免了对所介绍的攻击手段进行排名。就像《 Animal Farm 》（译者注：英国著名作家乔治 • 奥威尔的一部反乌托邦寓言小说）中的 Napoleon 和 Snowball 一样，某些 Web 安全漏洞比其他漏洞更重要。风险、影响和威胁等概念要求与 Web 应用程序的上下文和环境相关的信息，不能一概而论。

威胁可能来自黑客、匿名者（Anonymous，用大写 A 代表）、犯罪集团、海啸、磁盘故障、绊掉电源线、心怀不满的程序员等任何能够对网站带来负面影响的因素，他们就像演员一样，以你的网站作为舞台。

Dan Geer 对安全有一个形象的描述“没有什么意外不能被缓解”[⊖]。从这里看，风险可以从预期、检测和防御某些事情的能力的角度来衡量。影响风险的因素很多：威胁、Web 站点

⊖ http://harvardnsj.org/2011/01/cybersecurity-and-national-policy/

或被保护信息的价值、你认为目前 Web 站点有多么安全，网站被攻击之后恢复的难易程度，等等，这些因素中很多都很难度量。

如果你的 Web 站点存在漏洞，那么这就是一个 bug。威胁可能来自漫无目的的黑客或一名能力很强的、执着的黑客。根据度量标准的不同，风险可以是高或低。风险也可以有区别，有可能是用来注入一个指向恶意软件的 iframe，或者用来给网站设一个后门以窃取用户的证书。在任何情况下，修复漏洞总是好的。通常，修复 bug 要比确定利用它来进行攻击的威胁更容易。事实上，如果这个 bug（不管是否与安全有关）难以修复，那么就表示那里存在着高风险。

不对漏洞进行评级并不代表我们漠视这个概念。威胁建模是一个有助于思考针对 Web 站点的潜在安全问题或攻击的杰出工具。OWASP 网站总结了建造这些模型的不同方法，网址是 https://www.owasp.org/index.php/Threat_Risk_Modeling。微软的 STRIDE（http://www.microsoft.com/security/sdl/adopt/threatmodeling.aspx）是一个很好的面向威胁的参考文献。Common Weakness Enumeration（常见弱点枚举）采取的是另一种方式，列出了威胁所针对的各类编程错误，网址是 http://cwe.mitre.org/。

本书的结构

本书共有八章，描述针对网站以及浏览器的黑客攻击。每章都提供了针对真实站点进行攻击的实例，接着会研究漏洞利用的细节。读者不需要严格按顺序阅读本书。很多攻击方法都互相关联或组合，从而使特定的应对措施失效。这就是为什么我们要理解网络安全的各个方面，尤其需要注意的是 Web 安全既包含浏览器安全，也包括站点安全。

第 1 章：HTML5

一个新标准的产生意味着会出现新的漏洞，同时也意味着利用原有漏洞会有新方法。该章介绍 HTML5 标准的部分主要 API 和特性。HTML5 可能还不是正式标准，但它现在已经广泛用在浏览器和 Web 站点中。它不仅牵扯到安全，而且对信息的隐私性也有影响。

第 2 章：HTML 注入及跨站脚本攻击

该章描述了在 Web 站点中最广泛且最容易利用的漏洞之一。XSS 漏洞就好像 Web 中的蟑螂，总是潜伏在站点意想不到的角落中，不论站点的大小或名气，也不论其安全团队多么老练，都要面对这个问题。该章展示了如何只通过浏览器和最基本的 HTML 知识就可以利用 Web 中最常见的漏洞，另外还说明了 Web 站点和浏览器之间的紧耦合从安全角度而言是如何变成一种脆弱关系的。

第 3 章：跨站请求伪造

第 3 章继续介绍以 Web 站点和浏览器为目标的漏洞。跨站请求伪造（Cross-Site Request Forgery，CSRF）攻击欺骗被攻击者的浏览器，使其发送并非代表用户意愿的请求。这些攻击很隐蔽并且很难阻挡，毕竟从技术角度而言，每个 Web 页面都容易受到 CSRF 攻击。

第 4 章：SQL 注入攻击及数据存储操纵

该章的重点是 Web 应用程序以及驱动这些应用程序的数据库。SQL 注入攻击被人们认为是信用卡盗窃的根源。该章揭示了基于这个简单漏洞，可以有很多其他漏洞利用的方法，还展示了与成功的攻击带来的巨大影响相比，它的应对措施相对简单且易于实现。即便你的网站中没有 SQL 数据库，它仍旧可能容易受到类似 SQL 注入的数据注入、命令注入等攻击。

第 5 章：攻破身份认证模式

第 5 章包含了计算机安全中最古老的攻击之一：针对登录提示进行密码暴力破解。但是暴力破解攻击并不是攻破网站身份认证模式的唯一途径。该章还介绍了其他攻击手段以及对网站进行保护的应对措施。

第 6 章：利用设计缺陷

第 6 章介绍了一种更有趣的攻击类型，它不需要攻击者有很强的技术实力，只需要有基本的好奇心即可。针对网站业务逻辑的攻击变化很多，但是大部分攻击采用类似的技术，或者利用网站设计中的失误，使攻击者可以获得直接的经济收益。该章讨论了站点如何组成一个整体，攻击者如何为了个人利益找出漏洞，以及开发人员在面对这类没有合适编程清单的问题时能做些什么。

第 7 章：利用平台弱点

即使一个 Web 站点的编码非常安全，它的安全性也会因为配置不当而减弱。该章解释了服务器管理员可能犯下哪些错误，从而把网站暴露在攻击之下。该章还介绍了如果网站开发人员在站点的某些区域把安全建立在假定之上，而不是深思熟虑的举措之上，就可能给攻击者留下立足点。

第 8 章：攻击浏览器和隐私

最后一章将 Web 安全问题重新引回到浏览器。该章包含了恶意软件发展成为 Web 威胁的历史。该章还讲述了用户在网站的安全脱离自己掌控的情况下如何保护自身的安全。

进一步的目标

学习新的安全技术或巩固已有技术的最佳途径就是亲手实践。本书为找出并阻止漏洞提供了实例和方法描述。强化本书中的知识的最佳方法之一就是针对真实 Web 应用程序运用这些知识。通常，在网络上随意选择一个网站进行攻击是不道德的，甚至是非法的。然而，在这方面的安全观念正在慢慢改变。如果对 Google 的某些 Web 特性进行了尽责的测试，Google 会提供现金奖励[⊖]。Twitter 同样公正地对待负责的测试[⊜]。这些举措并不意味着可以对网站进行全权授权地攻击，尤其是盗取信息或侵犯他人隐私的攻击。但是，你会发现很难找到其他欢迎反馈或报告漏洞的更复杂的网站。

有一些培训网站，例如 Google 的 Gruyere（http://google-gruyere.appspot.com/）、OWASP 的 WebGoat（https://www.owasp.org/index.php/Webgoat）、DVWA（http://www.dvwa.co.uk/）。更妙的是，还可以搜索 SourceForge（http://www.sf.net/）、Google Code（（http://code.google.com/）和 GitHub（https://github.com/）来寻找开源 Web 应用程序。你可以下载并安装一些 Web 应用程序。部署一个网站的过程（修复 bug 或调整设置来安装 bug）可以帮助你建立起真实 Web 站点概念、编程模式及系统管理的经验。这些基本知识对理解安全性而言，比收集攻击检查列表更为重要。在你安装完成 PHP、Python、.NET、Ruby、Web 应用程序之后，就可以开始寻找漏洞了。可能会有 SQL 注入问题，或者没有过滤 POST 数据以防止跨站脚本攻击。不要一味寻找 Web 应用程序的最新发布版本，寻找有着在最新版本中已经修复的 bug 的老版本，对比不同版本之间的差异，能够使你发现别人是如何采用的应对措施，甚至有些情况下你可以发现某些应对措施并不恰当。

为数众多的移动应用以及 Web 公司的巨大估值，使得 Web 安全问题在未来相当长的一段时间内仍然会受到极大的重视。欢迎访问本书配套网站 http://deadliestwebattacks.com/，网站中提供了一些代码实例、意见建议、新闻、新技术以及本书的更新。

开始你的破解之旅吧！

⊖ http://googleonlinesecurity.blogspot.com/2010/11/rewarding-web-application-security.html

⊜ http://twitter.com/about/security

Contents 目　录

第 1 章 Chapter 1

HTML5

本章内容

- HTML5 的新增特性
- 使用及滥用 HTML5 的安全性考虑

书面语言的历史至少可以追溯到 5000 年前，当时的苏美尔人使用楔形文字来记录账簿、法律以及清单，这种原始的石刻标记语言为现代的超文本标记语言（HTML）开辟了道路。类似维基百科这样的网站，不就是收集了拜占庭法律、《吸血鬼猎人巴菲》剧集列表、《星际迷航》中的外星人名单等内容吗？由此可见，人类喜欢使用书面语言来记录各种信息。

很大程度上，HTML 是基于多种事实上的实现发展起来的标准。很少有浏览器定义 HTML 是什么，这意味着 HTML 标准在一定程度上体现了真实世界。如果你根据规范来书写网页，那么浏览器将会按照你的期望来适当地渲染它。在早期的演化发展过程中，标准的缺点是当时的页面并不统一，不同的浏览器有着不同的怪异（quirk）模式，导致产生类似如下的脚注："建议使用 IE 4 浏览本网页"、"建议使用 Mosaic 浏览本网页"。这些怪异模式成为了开发人员的噩梦，导致出现不良设计模式（例如常见的通过用户代理嗅探来侦测性能而不是通过特性测试）或过度依赖于插件（例如 Shockwave）。标准中还包含一些很少使用的标签（<acronym>）、糟糕的 UI 设计（<frame> 和 <frameset>）或非常令人厌恶的标签（<bgsound> 和 <marquee>）。HTML2 试图澄清某些差异，于 1995 年 11 月成为标准。HTML3 未能与 HTML2 合并为可接受的标准。HTML4 于 1999 年 12 月提出。

8 年之后，HTML5 作为公共草案出现，过了一年左右才大受欢迎。现如今，在 HTML4