阐述了 XSS 的基础知识，剖析了 XSS 的攻击原理和危害
讲解了 XSS 测试工具、发掘 XSS 漏洞技术、XSS Worm 防御、Flash 应用安全
用典型案例演示了 XSS 跨站防御方案及防御 XSS 攻击方法

XSS 跨站脚本攻击剖析与防御

邱永华 编著

人民邮电出版社
POSTS & TELECOM PRESS

XSS

跨站脚本

攻击剖析与防御

邱永华 编著

人民邮电出版社

北京

图书在版编目（CIP）数据

XSS跨站脚本攻击剖析与防御 / 邱永华编著. -- 北京 : 人民邮电出版社, 2013.9
ISBN 978-7-115-31104-7

Ⅰ. ①X… Ⅱ. ①邱… Ⅲ. ①计算机网络－安全技术 Ⅳ. ①TP393.08

中国版本图书馆CIP数据核字(2013)第103720号

内 容 提 要

本书是一本专门剖析 XSS 安全的专业书，总共 8 章，主要包括的内容如下。

第 1 章 XSS 初探，主要阐述了 XSS 的基础知识，包括 XSS 的攻击原理和危害。第 2 章 XSS 利用方式，就当前比较流行的 XSS 利用方式做了深入的剖析，这些攻击往往基于客户端，从挂马、窃取 Cookies、会话劫持到钓鱼欺骗，各种攻击都不容忽视。第 3 章 XSS 测试和利用工具，介绍了一些常见的 XSS 测试工具。第 4 章 发掘 XSS 漏洞，着重以黑盒和白盒的角度介绍如何发掘 XSS 漏洞，以便帮助读者树立安全意识。第 5 章 XSS Worm，讲解了 Web 2.0 的最大威胁——跨站脚本蠕虫，剖析了 Web 2.0 相关概念和其核心技术，这些知识对于理解和预防 XSS Worm 十分重要。第 6 章 Flash 应用安全，就当前的 Flash 应用安全做出了深入阐述。第 7 章 深入 XSS 原理，讨论一些比较深入的 XSS 理论。第 8 章 防御 XSS 攻击，介绍了一些防范 XSS 攻击的方法，例如，运用 XSS Filter 进行输入过滤和输出编码，使用 Firefox 浏览器的 Noscript 插件抵御 XSS 攻击，使用 HTTP-only 的 Cookies 同样能起到保护敏感数据的作用。

本书适合网站管理人员、信息/网络安全或相关工作从业者、软件开发工程师，以及任何对 Web 安全技术感兴趣的读者。

◆ 编　　著　邱永华
责任编辑　张　涛
责任印制　程彦红　杨林杰

◆ 人民邮电出版社出版发行　　北京市崇文区夕照寺街 14 号
邮编　100061　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京艺辉印刷有限公司印刷

◆ 开本：800×1000　1/16
印张：17.25
字数：413 千字　　2013 年 9 月第 1 版
印数：1-3 500 册　　2013 年 9 月北京第 1 次印刷

定价：49.00 元

读者服务热线：(010)67132692　印装质量热线：(010)67129223
反盗版热线：(010)67171154

序

在 2012 年年末的时候，邱永华先生找到我，希望我能为他的新书作序，我也得以有幸提前拜读到了本书的手稿。

在我看来，本书的完成，是一件值得高兴的事情。因为这是国内第一本专门阐述 XSS 的著作，本书的问世，为学习 Web 安全的新人提供了充分的学习材料，也为安全从业者提供了一份不可多得的参考手册，最终也必然将推动大家对 XSS 安全技术的重视。

长期以来，XSS 攻击的危害都没有得到大多数的开发者的正确认识，甚至有的网络安全工作者也认为 XSS 的“危害不大”。造成这种误解的原因是多方面的。

XSS 攻击的危害与具体业务场景密切相关。不同的业务场景，会导致不同的网络安全问题，有的场景危害大，有的场景危害小。而 XSS 作为一种漏洞类型，在描述其定义时很难定位到具体的场景里去。XSS 攻击的危害程度大小，依赖于业务场景的重要程度。

我在阿里巴巴工作期间，曾经负责处理淘宝、支付宝的钓鱼欺诈案件。当时就发现很多案件中，XSS 漏洞被用于网购钓鱼。诈骗者将一个 XSS 链接通过即时通信软件发送给用户，用户单击后，会自动跳转到钓鱼网站的页面，最终造成资金损失。在这样的案件中，诈骗者利用 XSS 使得链接的域名是真实合法的网站，从而绕过了所有安全软件的检测。我当时曾经粗略估算过一个 XSS 漏洞造成的损失，如果算上用户损失的金额，以及网站的修复成本，在案件最猖獗的时候，每个 XSS 漏洞会带来超过 50 万人民币的损失。

除此之外，曾经有多起网络犯罪利用 Web Mail 的 XSS 漏洞窃取目标用户的邮箱，而这样的定点渗透攻击所造成的损失往往难以衡量。

XSS 攻击与浏览器也密切相关，在不同的浏览器上有着不同的表现。随着互联网的发展，浏览器的版本更新非常迅速。因此，想要熟练地掌握 XSS 防范技巧，需要对不同浏览器的特性进行深入地了解。

相对于攻击服务端的漏洞来说，XSS 的攻击目标是客户端。通常来说，网站开发者、网站安全工程师会更重视攻击网站服务器的安全漏洞。但是，站在用户的角度，或者说从整个互联网安全的角度来看，XSS 的安全性应该得到很好重视。

XSS 攻击能够控制目标用户的浏览器做任何事情，因此，也会造成用户数据、用户隐私的泄露。在数据时代这一点显得尤为敏感。但是，今天很多网站的用户数据并未得到妥善的保管，很多爬虫、第三方抓取软件都或多或少地能从网站上抓取到一些用户数据，这也许会使得 XSS 的危害看起来不是那么的突出。但本书将会告诉你，XSS 能做到的事情可能会远远出乎你的想象，加以防范是很重要的。

可以毫不夸张地说，几乎每个网站都或多或少地存在一些 XSS 漏洞。这些众多的 XSS 漏洞就像互联网里的一片片“雷区”，谁也不知道它们什么时候会“爆炸”，造成损失。

随着 JavaScript 和 HTML5 技术的发展，越来越多的网站和移动应用开始使用更加高级的前端技术，因此，也必然会催生 XSS 攻击的升级。XSS 攻击在未来十年可能会产生质的变化，而且也注定会是互联网安全领域内的一个值得长期关注的热点。

如果以前 XSS 安全被忽视的话，那么希望，从本书起，能够引起足够的重视。

研究 XSS，从本书开始！

吴翰清（网名：刺）

于杭州

前　言

如果说现代 Web 安全中有什么不容忽视的事实，那便是 XSS（Cross-site Scripting）跨站脚本成为了 Web 威胁之首。长期以来，跨站脚本作为最常见的计算机安全漏洞，在世界各地的网站上疯狂肆虐，即使最知名的网站，如 Google、Facebook、微软官方网站等，也曾遭受过 XSS 漏洞攻击。然而，由于 XSS 属于被动式攻击且不易利用，所以一直不被重视。

“XSS？不就是弹出个对话框给自己看吗？！”

“反正 XSS 不能窃取我的 root 权限。”

“跨站脚本是在客户端执行，XSS 漏洞关我什么事！”

“XSS 等同于鸡肋漏洞。”

……

这些就是 XSS 跨站脚本给大部分人留下的印象。

作为一名 Web 应用安全研究者，我对 XSS 的态度迥然不同，并且认为 XSS 跨站脚本的危害不亚于缓冲区溢出、代码执行、SQL 注射等安全漏洞。众所周知，XSS 技术的运用方式灵活多变，利用它可以劫持浏览器用户的会话、窃取客户端 Cookies、网络钓鱼等，在特定场景下或配合其他漏洞，威力会更大！尤其是近年来，结合 Ajax 动态网页技术、Web 2.0 信息分享模式和社交网络，XSS 衍生出类似蠕虫般具有自我复制能力的攻击形态，不但能在短时间内造成大量客户端用户受到攻击，还能使受害的客户端用户对服务器产生的大量请求，形同对服务器的分布式拒绝服务攻击（DDos）。

令人遗憾的是，目前国内还没有一本专门讲述 XSS 技术的书籍，以致于人们对这类漏洞、攻击缺乏一定程度的了解。

幸运的是，网络上始终有一群“跨站师”致力于 XSS 技术研究，并且无私地分享着他们的技巧，于是大家逐渐对 XSS 漏洞的危害产生了更深层次的认识。

本书通过讲述有关跨站脚本的知识，读者可以深刻地感受到跨站脚本的强大，并且详尽地了解许多与 XSS 相关的内容，例如，在什么环境下可以触发 XSS，利用 XSS 漏洞可以做什么，如何防范此类攻击等。自始至终，本书贯穿着许多案例分析，读者可以在实际环境中进行安全测试。需要注意的是：本书的内容仅供学习之用，希望读者不要使用其中的代码和技术对其他网站发动攻击，否则后果自负。

由于这是本人第一次写书，其中难免有所错漏，欢迎读者斧正与交流！与此同时，希望本书可以对大家有所帮助。

归根结底，本书的真正目的是为了让大家理解 XSS 跨站脚本的危害并加以防范。

本书结构

全书总共 8 章，读者可以通过浏览目录以进一步了解各章的内容。在本书结尾，附上相关资料以及参考文献。

第 1 章 XSS 初探 带读者走进 XSS 跨站脚本的世界，此章主要阐述了 XSS 的基础知识，包括 XSS 的攻击原理、危害以及一些常用技巧，理解本章对学习后面的内容至关重要。

第 2 章 XSS 利用方式剖析 就当前比较流行的 XSS 利用方式进行深入阐述，这些攻击往往基于客户端，从挂马、窃取 Cookies、会话劫持到钓鱼欺骗，各种攻击都不容忽视。

第 3 章 XSS 测试和工具剖析 介绍了一些常见的 XSS 测试和利用的工具，前面 4 节主要讲述可测试 XSS 的工具，后面 4 节讲述 XSS 的利用平台。

第 4 章 发掘 XSS 漏洞 着重以黑盒和白盒的角度介绍如何发掘 XSS 漏洞，黑盒环境下可手动发掘 XSS 漏洞，也可以利用一些自动化测试工具；白盒环境下则可以通过分析代码的方式发掘 XSS 漏洞。

第 5 章 XSS Worm 剖析 讲解了 XSS 的终极利用方式，也是 Web 2.0 的最大威胁——跨站脚本蠕虫，此章还介绍了 Web 2.0 相关概念及其核心技术、浏览器的安全等，这些知识对理解 XSS Worm 十分重要。

第 6 章 Flash 应用安全 就当前的 Flash 应用安全进行深入阐述。尽管安全社区需经常修补一些 XSS、CSRF 和其他注入漏洞，但是，Flash 的应用中提供了一种新的攻击类型，尤其是那些无防备的和未经严格测试的 Flash 应用程序。

第 7 章 深入 XSS 原理 讨论一些比较深入的 XSS 理论，其中涉及许多特殊的 XSS 技巧和应用场景，这些 XSS 将会对传统的跨站防御方案提出挑战。同时，此章还会讲到其他类型的 Web 安全漏洞，这些漏洞均与 XSS 息息相关。

第 8 章 防御 XSS 攻击 介绍了一些防范 XSS 攻击的方法，例如运用 XSS Filter 进行输入过滤和输出编码、使用 Firefox 浏览器的 Noscript 插件抵御 XSS 攻击等，而使用 HTTPOnly 的 Cookies 同样能起到保护敏感数据的作用。

读者

本书的读者应当熟悉网页技术，包括 HTML 和 JavaScript，如果您还具备一些编程经验和网络安全的相关知识，便能更轻松地理解其中某些章节的内容。

本书读者对象包括：

- 网站管理人员；
- 信息/网络安全或相关从业者；
- 程序员/软件开发工程师，包括 Web 开发人员；
- 教授 Web 安全技术相关内容的教师；
- 任何对 Web 安全技术感兴趣的人。

致谢

出书是个浩大的工程，在本书写作期间，一直遇到很多困难，经历数月的时间才最终艰难完成！借此机会感谢所有使本书能够顺利出版提供帮助的朋友。

写书之前阅读和参考了大量 XSS 技术方面的文档，因此，本书汇聚了国内外许多安全研究员的研究成果和专业技能，在此一并表示感谢。

特别感谢吴翰清（网名刺）和钟晨鸣（网名余弦）先生，为本书提出的宝贵意见和指正。感谢我的家人和朋友，尤其是我的父母，你们为我付出很多！

在此，我要感谢活跃在 Web 安全圈子里的跨站师们，你们的公开文章和技巧让我学到了很多技术。

联系方式

邮箱：cnn4ry@gmail.com。

cnn4ry@163.com。

新浪微博：http://weibo.com/cnn4ry。

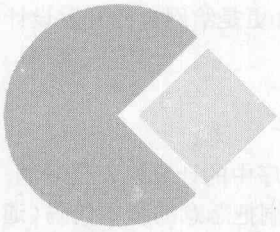
腾讯微博：http://t.qq.com/cnn4ry。

声明

本书仅限于讨论网络安全技术，请勿用作非法用途，严禁利用本书所提到的漏洞和技术进行非法攻击，否则后果自负，本人和出版商不承担任何责任！

编者

目　录



第1章 XSS初探

XSS（Cross-Site Scripting）跨站脚本自1996年诞生以来，如今已经历十多年的演化。在各种网络安全漏洞中，XSS一直都被OWASP（Open Web Application Security Project）组织评为十大安全漏洞中的第二威胁漏洞。也有黑客把跨站脚本当做新型的“缓冲区溢出攻击”，而JavaScript则是新型的ShellCode。

2011年6月份，国内最火的信息发布平台“新浪微博”爆发了XSS蠕虫攻击，新浪微博的XSS蠕虫爆发仅执续了16分钟，感染的用户就达到将近33000个，危害十分严重！

XSS最大的特点就是能注入恶意的HTML/JavaScript代码到用户浏览的网页上，从而达到劫持用户会话的目的。由于HTML代码和客户端JavaScript脚本能在受害者主机上的浏览器任意执行，这样等同于完全控制了Web客户端的逻辑，在这个基础上，黑客或攻击者可以轻易地发动各种各样的攻击。

在这一章中，我们会带你进入XSS的世界。

首先，我们会介绍一些跨站脚本的基本原理和相关知识，其中提到的许多概念和技巧也许算不上新颖，却是非常基础和重要的。关于什么是XSS跨站脚本，它会给网站和用户带来什么危害，以及它有什么特点，它的类型……本章将一一讲解。

在了解XSS的攻击原理之后，我们会展示一个简单的XSS漏洞挖掘示例，并且介绍强大的XSS攻击脚本列表——XSS Cheat sheet。

最后，我们会讨论一些XSS Expliot的构造技巧，包括如何运用各种手段绕过服务端程序对跨站脚本的防御，对XSS代码进行有效编码处理以及对XSS Shellcode进行存储和调用。

1.1 跨站脚本介绍

随着Web技术的蓬勃发展，XSS跨站脚本无疑已经变成最流行和影响严重的Web安全漏洞，并且广泛存在于各类Web系统之中。

网络上研究跨站脚本技术的人也逐渐多起来，从而催化了相关技术文章的大量涌现。具有讽刺意味的是，尽管人们已经开始关注XSS，却依然无法改变它到处泛滥的事实，这完全归结于XSS的独特之处。

在正式探讨XSS前，我们不得不提到它的核心——JavaScript。JavaScript最初是打算作为一个脚本接口，用于浏览器客户端加载的网页和服务端的应用之间，自1995年引入以来已变成Web开

发中一个不可或缺的重要部分。再加上 Ajax 等技术的日渐流行，JavaScript 更是给网页的开发设计带来了无限惊喜，这些技术无限地扩充了今天的网络安全领域。

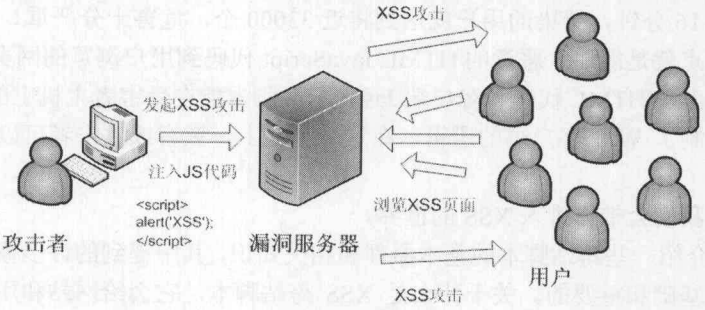
1.1.1　什么是 XSS 跨站脚本

跨站脚本（Cross-Site Scripting，XSS）是一种经常出现在 Web 应用程序中的计算机安全漏洞，是由于 Web 应用程序对用户的输入过滤不足而产生的。攻击者利用网站漏洞把恶意的脚本代码（通常包括 HTML 代码和客户端 Javascript 脚本）注入到网页之中，当其他用户浏览这些网页时，就会执行其中的恶意代码，对受害用户可能采取 Cookie 资料窃取、会话劫持、钓鱼欺骗等各种攻击。

由于和另一种网页技术——层叠样式表（Cascading Style Sheets，CSS）的缩写一样，为了防止混淆，故把原本的 CSS 简称为 XSS。

通常情况下，我们既可以把跨站脚本理解成一种 Web 安全漏洞，也可以理解成一种攻击手段。

XSS 跨站脚本攻击本身对 Web 服务器没有直接危害，它借助网站进行传播，使网站的大量用户受到攻击。攻击者一般通过留言、电子邮件或其他途径向受害者发送一个精心构造的恶意 URL，当受害者在 Web 浏览器中打开该 URL 的时候，恶意脚本会在受害者的计算机上悄悄执行，流程如图 1-1 所示。



▲图 1-1　XSS 攻击流程

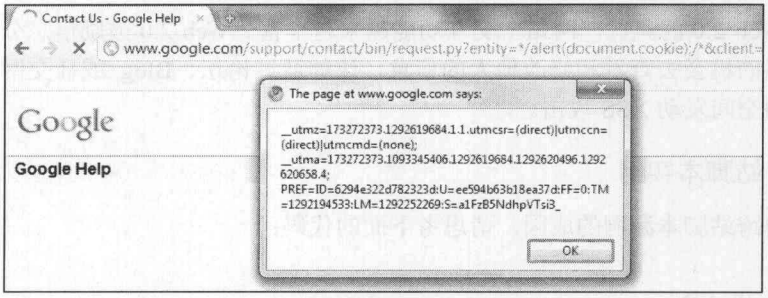
开放式 Web 应用程序安全项目（Open Web Application Security Project，OWASP）是世界上最知名的 Web 安全与数据库安全研究组织，该组织分别在 2007 年和 2010 年统计过十大 Web 安全漏洞，如图 1-2 所示。

OWASP Top 10 – 2007 (Previous)	OWASP Top 10 – 2010 (New)
A2 – Injection Flaws	A1 – Injection
A1 – Cross Site Scripting (XSS)	A2 – Cross-Site Scripting (XSS)
A7 – Broken Authentication and Session Management	A3 – Broken Authentication and Session Management
A4 – Insecure Direct Object Reference	A4 – Insecure Direct Object References
A5 – Cross Site Request Forgery (CSRF)	A5 – Cross-Site Request Forgery (CSRF)
<was T10 2004 A10 – Insecure Configuration Management>	A6 – Security Misconfiguration (NEW)
A8 – Insecure Cryptographic Storage	A7 – Insecure Cryptographic Storage
A10 – Failure to Restrict URL Access	A8 – Failure to Restrict URL Access
A9 – Insecure Communications	A9 – Insufficient Transport Layer Protection
<not in T10 2007>	A10 – Unvalidated Redirects and Forwards (NEW)
A3 – Malicious File Execution	<dropped from T10 2010>
A6 – Information Leakage and Improper Error Handling	<dropped from T10 2010>

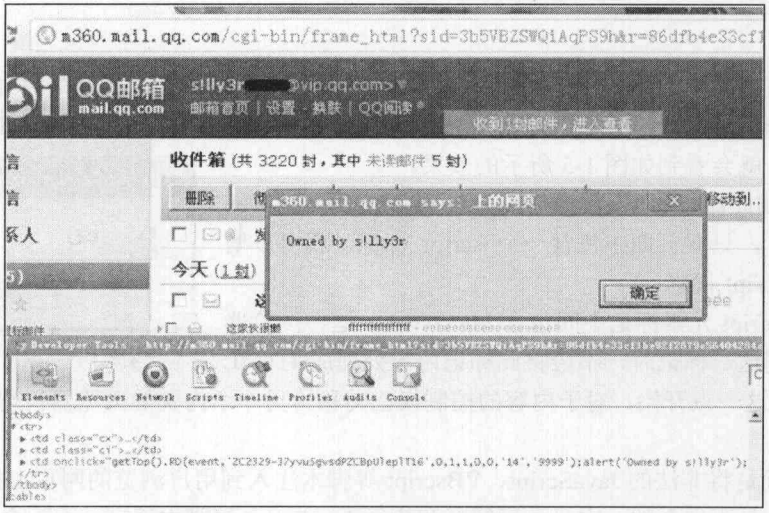
▲图 1-2　OWASP 十大安全漏洞统计

从图 1-2 中我们看到，在 2007 年 OWASP 统计的所有安全威胁中，XSS 跨站脚本就高居第二位。

时至今日，XSS 依然是网站漏洞中最容易出现的一种，据说在现今的各大网站中都存在此漏洞，包括 Google、腾讯等大型网站都频繁出现过，分别如图 1-3 和图 1-4 所示。



▲图 1-3　Google 的 XSS 漏洞



▲图 1-4　QQ 邮箱 XSS 漏洞

为什么 XSS 跨站漏洞会如此普遍和流行？这是由多个因素造成的。

（1）Web 浏览器本身的设计是不安全的。浏览器包含了解析和执行 JavaScript 等脚本语言的能力，这些语言可用来创建各种格式丰富的功能，而浏览器只会执行，不会判断数据和程序代码是否恶意。

（2）输入与输出是 Web 应用程序最基本的交互，在这过程之中若没做好安全防护，Web 程序很容易会出现 XSS 漏洞。

（3）现在的应用程序大部分是通过团队合作完成的，程序员之间的水平参差不齐，很少有人受过正规的安全培训，因此，开发出来的产品难免存在问题。

（4）不管是开发人员还是安全工程师，很多都没有真正意识到 XSS 漏洞的危害，导致这类漏

洞普遍受到忽视。很多企业甚至缺乏专门的安全工程师，或者不愿意在安全问题上花费更多的时间和成本。

（5）触发跨站脚本的方式非常简单，只要向 HTML 代码中注入脚本即可，而且执行此类攻击的手段众多，譬如利用 CSS、Flash 等。XSS 技术的运用如此灵活多变，要做到完全防御是一件相当困难的事情。

（6）随着 Web 2.0 的流行，网站上交互功能越来越丰富。Web 2.0 鼓励信息分享与交互，这样用户就有了更多的机会去查看和修改他人的信息，比如通过论坛、Blog 或社交网络，于是黑客也就有了更广阔的空间发动 XSS 攻击。

1.1.2　XSS 跨站脚本实例

如果不明白跨站脚本漏洞的成因，请思考下面的代码：

```
<html>
 <head>test</head>
 <body>
   <script>alert("XSS")</script>
 </body>
</html>
```

这是一段很简单的 HTML 代码，其中包括一个 JavaScript 语句块，该语句块使用内置的 alert() 函数来打开一个消息框，消息框中显示 XSS 信息。把以上代码保存为 HTM 或 HTML 文件，然后用浏览器打开，就会看到如图 1-5 所示的效果。

▲图 1-5　浏览器弹出消息框

JavaScript 如此流行的原因之一是：把 JavaScript 加入到 Web 页面中非常简单，只要页面中包含一个 script 标记，就可以增加想要的 JavaScript 代码。

HTML 的 script 元素标记中间包含 JavaScript，这使浏览器知道：当它遇到这一标记时，不应将此标记内容处理成 HTML 或 XHTML，从这一点开始，对于内容的控制权已转移给另一个内置的浏览器代理——脚本引擎处理。

XSS 攻击就是将非法的 JavaScript、VBscript 等脚本注入到用户浏览的网页上执行，而 Web 浏览器本身的设计是不安全的，它只负责解释和执行 JavaScript 等脚本语言，而不会判断代码本身是否对用户有害。

如想利用 XSS 弹出恶意警告框，代码为：

```
<script>alert("XSS");</script>
```

XSS 输入也可能是 HTML 代码段，如要使网页不停地刷新，代码为：

```
<meta http-equiv="refresh" content="0;">
```

嵌入其他网站的链接，代码为：

```
<iframe src=http://www.test.com width=0 height=0></iframe>
```

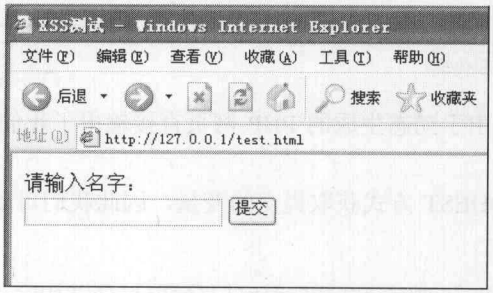
现在，相信读者已经大概了解了跨站脚本的基本概念，接下来，我们以一个简单的 PHP 网页

为例，来具体说明 XSS 攻击的原理。PHP 网页的作用是让用户输入名字并且显示在页面上。

提供用户输入信息的 HTML 文档的代码如下：

```
<html>
 <head>
  <title> XSS 测试 </title>
 </head>
 <body>
   <form action="XSS.php" method="POST">
     请输入名字：<br>
     <input type="text" name="name" value=""></input>
     <input type="submit" value="提交"></input>
 </body>
</html>
```

当用浏览器打开这个网页的时候，会显示如图 1-6 所示的效果。

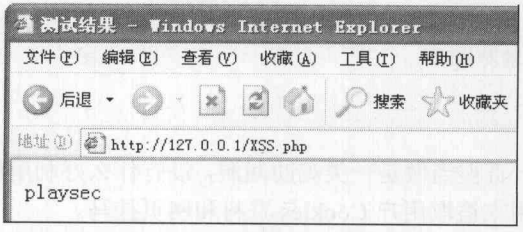


▲图 1-6 XSS 测试页面

后台 PHP 的处理代码如下：

```
<html>
 <head>
  <title> 测试结果 </title>
 </head>
 <body>
   <?php
     echo $_REQUEST[name];
   ?>
 </body>
</html>
```

以上代码使用$_REQUEST[name]获取用户输入的 name 变量，然后直接 echo 输出。打开测试页面，随便输入一些信息，例如 playsec，然后单击【提交】按钮，此时返回如图 1-7 所示的页面。

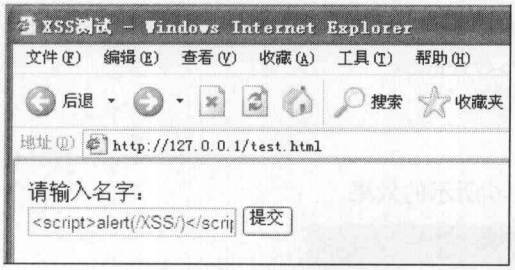


▲图 1-7 页面返回输入的用户名

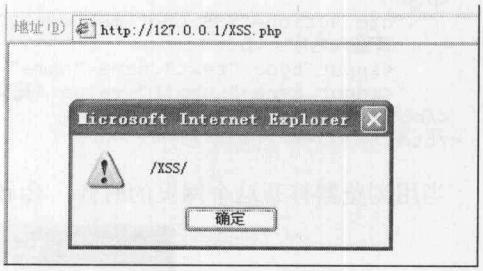
从上图中可以看到，页面把我们刚刚输入的 playsec 完完整整地输出来了，那么，再尝试输入一些 HTML/JavaScript 代码，如在文本框中输入：

```
<script>alert(/XSS/)</sctipt>
```

如图 1-8 所示，单击【提交】按钮后，结果如图 1-9 所示。



▲图 1-8　输入 XSS 代码



▲图 1-9　执行脚本代码

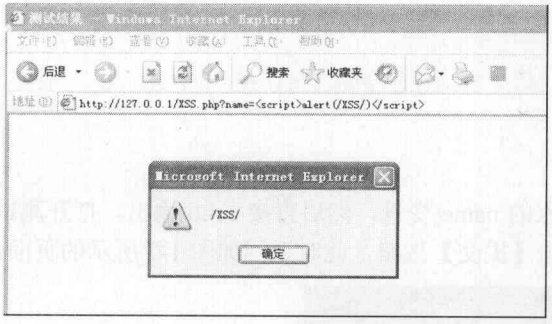
从图 1-9 中可以得知，由于动态生成的 PHP 网页直接输出了我们的测试代码，从而导致一个 XSS 的生成。

PHP 代码中使用$_REQUEST 方式获取提交的变量，因此我们可以用 GET 方式触发 XSS，即直接在浏览器访问：

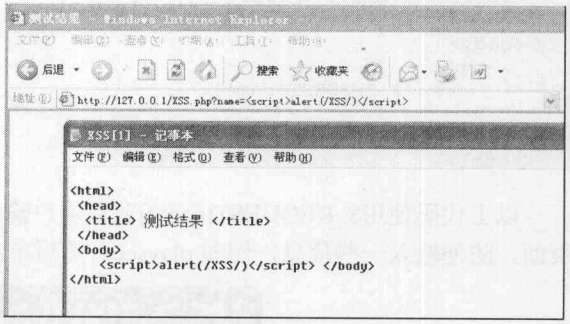
```
http://127.0.0.1/XSS.php?name=<script>alert(/XSS/)</script>
```

然后浏览器会弹出一个对话框，如图 1-10 所示。

这时在页面单击鼠标右键，从弹出的快捷菜单中选择“查看源文件”命令，可以打开如图 1-11 所示的 HTML 代码片段。



▲图 1-10　触发 XSS



▲图 1-11　查看页面的源文件

1.1.3　XSS 漏洞的危害

以往，XSS 跨站脚本一直被当做是一类鸡肋漏洞，没有什么好利用的地方，只能弹出对话框而已，稍微有点危害的就是用来盗取用户 Cookies 资料和网页挂马。

通常情况下，攻击者通过注入如 alert('xss')之类的 JavaScript 代码来证明 XSS，该代码能够导致