# 网络安全攻击与防护研究

## 1. 引言

### 1.1 Web 安全技术的背景

随着 Internet 的快速发展，人们对它的依赖也越来越强，但是正是由于 Internet 的开放性，以及在设计的时候对于信息的保密和系统的安全考虑的不完善，造成了现在网络的攻击与破坏事件越来越多，给人们的日常生活和经济活动造成了非常大的麻烦。由于 Web 服务作为当今 Internet 上使用最广泛的服务，所以 Web 站点被黑客入侵的事情屡有发生，Web 安全问题已引起社会各界的极大重视。

然而由于社交网络、购物网站、微博等等一系列新颖的互联网产品相继推出，基于 Web 环境的网络应用也越来越广泛，企业信息化过程中的各种应用大部分都需要架设在 Web 平台上，所以 Web 应用的飞速发展也引起了黑客们的高度关注，随之而来的就是 Web 网络安全的问题，黑客们利用网站的操作系统漏洞和一些 Web 服务程序中的 SQL 注入漏洞得到 Web 服务器的操控权。这样，他们不但可以篡改网页内容，还可以窃取内部的重要数据，更严重的是在网页中植入自制的恶意代码程序，使得那些访问网站的访问者受到攻击。正因如此，这也使得越来越多的用户关注 Web 的安全问题，对 Web 应用安全的关注度也逐渐升温。

### 1.2 Web 安全的概述

安全是什么？什么样的情况下会产生安全问题？我们要如何看待安全问题？只有搞明白了这些最基础，最简单的问题，才能明白一切防御技术的出发点，才能明白为什么我们要这样做。

那么安全问题是怎么产生的呢？举个例子：当我们 出去旅游或者出差的时

候，无论是用什么交通方式，都要经过安检这一环节，安检就是为了将一些危险品过滤掉，这样才能保证旅客们真正的安全。

同样，从安全的角度看，我们可以将不同信任域之间建立一个信任边界。那些从高等级信任域往低等级信任域流的数据时不需要进行检查的，而反之低等级信任域到高等级信任域是要进行检查的。这也就是为什么我们从车站，机场出来不需要检查，而想要再次进入时需要再次进行检查。

# 2. Web 安全现状和未来发展

## 2.1 Web 安全现状

在日益发展的今天，Internet 已经成为了一个非常重要的基础平台，很多企业都将自己的应用架设在互联网平台上，为客户提供更方便、快捷的服务。这些应用在功能和性能上，都在不断的完善和提高，然而在非常重要的安全性上，却没有得到足够多的重视。而且现在网络技术发展的越来越成熟，黑客们也将他们的注意力从以往对网络服务器的攻击逐渐转移到了 Web 应用端的攻击上。根据 2014 年 12 月份 Gartner 的最新调查，信息安全攻击有 75% 都是发生在 Web 的应用层而非网络层上。同时，数据也显示，三分之二的 Web 站点都相当脆弱，非常容易受到攻击。然而绝大多数企业都是将大量的投资花费在网络和服务器的安全上，没有从真正意义上保证来解决 Web 应用本身的安全，给黑客有可乘之机。那么黑客主要用什么手段进行 Web 攻击呢，就在近期有统计过 Web 网络遭受攻击的手段除了简单的扫描之外，最严重的还是 SQL 注入和 XSS 跨站攻击了，这两中攻击手段是目前来说危害较大的了。

而且我国对 Web 安全的重视程度远远不够，很多地方对于 Web 安全意识还很薄弱。因此，了解 Web 网络的攻击手段和一些必要的防护措施迫在眉睫。下

面将会着重介绍以下当下流行的攻击手段和防护措施。

### 2.2 Web 安全的未来发展

Web 原来被设想为通用的应用，它可以在任何地方运行的最低标准的应用，但是这样的应用它的功能就要被内容所限制了。

而 App 提供了另外一种替代的方法，也就是在后台执行代码、进行数据缓存供离线使用，采用推送通告等，这些都是网站无法做到的。

但是 Web 并没有坐以待毙，它也正在快速的变化来应对 app 的威胁。拥有 18 年 Web 经验的 Haakenson 专家认为，2015 年将会是有史以来 Web 发展最令人兴奋的一年，他预测了 2015 年 Web 的发展会有以下一些重要的趋势：

1、ServiceWorkers 可以让网站安装 JS 的文件，JS 文件会在一个独立于页面的环境下运行。这样的 javascript 的脚本就可以提供跨页面的持续性，还可以来侦听页面的请求，然后在无需通过网络的情况下返回内容。这样一来，内容就可以在不需要在百分之百连接的情况下进行缓存、转换或者用有创意的新方式提供出来。

2、传感器访问能够赋予页面对用户环境的感知能力。一直以来，Web 页面掌握的用户情况十分有限，通常只有用户的屏幕尺寸还有浏览器的类型等。但现在的各种标准把环境、光亮、摄像头等各种数据都提供出来了。这就使得网站可以在跳出页面之外掌握更为丰富的信息。

3、推送通告可以让网站与用户保持处于连接的状态，哪怕用户关闭了网站的浏览器标签页，也可以接受到网站的推送报告。

4、ServiceWorkers 与推送通告的结合也能产生非常重要的效应。推送未必就要把通告给用户，也可以是执行任意 ServiceWorker 代码。例如：某个你很感兴

趣的节目又出新的内容了，这是就可以触发推送给某个 ServiceWorker，让后者把内容预加载到缓冲，这样你坐车回家的时候就可以使用离线浏览了。

不过有人也许会对浏览器是否具备这种能力感到担忧。但是这种担心大可不必，这些功能的使用都需要经过权限检查，在使用的时候也会有可以见到的提示信号。比方说，用麦克风进行录音时，标签的图标上会显示录音的符号，这样用户可以知道也可以随时取消。

另一个重要趋势是 Web 的安全化。像推送这样的新功能过于强大，需要用 HTTPS 来进行一个保护措施。HTTPS 还可以防止恶意的植入代码，给网站和用户造成长期影响。

正如 SANS 研究员 Ed Skoudis 所说：“当一切不同的事物进入这个环境中，如果你不知道漏洞出现在哪里，那你便无法抵御它。”该论断对于 Web 安全同样适用，传统的 Web 威胁防御方式是基于对木马、病毒等安全威胁的认识经验而来，而新的威胁已经完全了解了这些传统的做法，攻击时根据实际情况改变策略，就可以轻松绕过防御，造成破坏。

特别是黑客对于安全漏洞的利用更加使 Web 安全威胁进入了一个新的阶段。仅仅在 2014 年，就出现了“心脏流血”漏洞、Bash 漏洞等特大型漏洞，利用这些漏洞，黑客很有可能完全控制网站服务器等的目标系统。更可怕的是，这些漏洞在被发现之前都是远离安全研究人员视线，这些漏洞很有可能正在或者已经被黑客用来攻击目标系统。而且，地下黑客市场的兴起让黑客可以快速获取漏洞攻击套件，对用户造成重大的威胁。

越来越多的攻击案例证明，Web 安全事故很有可能导致组织遭受重大的损失。在今年 4 月份法国电视 5 台遭受的重大网络攻击中，黑客袭击了法国电视 5

台的官方网站，导致大量电视台服务被中断，以及重要的网站内容被篡改成黑客声明，攻击波及人数高达两亿。有分析指出，黑客很有可能通过对未知网站漏洞的利用，或者对网站服务器的控制来实现攻击目标，这一事件给我们提了一个醒，让我们意识到对网站威胁进行监控、预警的重要性。

而在专家看来，Web 安全并非只是对已知漏洞、脚本、木马的清除与防护，而是越来越多的指向对未知安全威胁方面的防范，以达到防范于未然的效果。这就需要 Web 服务提供商建立一个完整的 Web 安全体系，将安全情报搜集、安全预警放在 Web 防护的重要位置，在安全事故发生之前就消灭威胁的根源，这样不仅可以有效的防范 Web 安全威胁，还能在最大的程度上减少因 Web 安全事故导致的损失。”

## 3. 常用的 Web 攻击手段

Web 的攻击手段多种多样，然而目前主流的手段有 XSS 跨站攻击，SQL 注入以及网站的钓鱼等。下面主要就是介绍一下这三类的攻击手法。

### 3.1 XSS 跨站

跨站脚本攻击 XSS 时客户端脚本安全中的头号敌人，OWASP TOP 10 威胁曾经多次把 XSS 列为榜首。

XSS 攻击，通常是指黑客通过对网页的修改，并且在网页中插入了一些自己制作的恶意脚本或者病毒，当用户在浏览网页时，就可以控制用户浏览器的一种攻击。在以前刚开始的时候，这种攻击是需要跨域的，所以也就有了跨站攻击之说，但是互联网发展到今天，随着 javascript 的日益完善，现在的 XSS 是否跨域已经不是那么重要了，但是因为好多人都习惯了叫跨站攻击为 XSS，所以 XSS 这个名字就被保留了下来。

XSS 漏洞有很多种，大概可以分为：反射性，存储型两大类，而从某种意义上来讲存储型 XSS 漏洞的危害性要更大一些。

如何判断反射型和存储型的 XSS：简单来讲，反射型的 XSS 是需要用户手工去点击的，所以反射性的 XSS 也被称之为“非持久型的 XSS”，而存储型的只要打开网页就会自动出现的，因此具有很强的稳定性。

当然 XSS 也不可能是空穴来风，每件事情的发生都必然会有原因的存在，所以 XSS 形成的原因也有很多种：

① 用户提交的，未经过过滤的信息直接写到网页上 ② 网页或 FLASH 文件调用 js 时,参数未经过过滤 ③ 网页给用户设置 cookie 时让用户可以修改参数 ④ 其他类型的 XSS

有人说 XSS 的危害性并不大，认为不过就是弹出一个弹框而已，没有什么大不了的，但是事实却不是这样，黑客正是利用弹窗来窃取用户的 cookie 等信息的。

### 3.2 SQL 注入

所谓 SQL 注入，就是构造 SQL 命令加入到 Web 表单提交或输入在域名以后以及在页面请求查询的字符串，最终成功欺骗服务器，从而执行恶意的 SQL 命令，达到入侵的效果。

然而，SQL 注入攻击的本质，就是把用户输入的数据当做命令来执行，然而完成这一个动作则需要两个很关键的条件，第一个就是用户能够控制输入内容，第二个就是原程序要执行代码，并且要与用户输入的数据进行合并。

简单来说，就是利用现有应用程序，将恶意的 SQL 命令插入到后台的数据库中并执行，它可以通过在 Web 表单中输入一串恶意的 SQL 语句得到一个存在安

全漏洞的网站上的数据库，而不是按照设计师的想法去执行 SQL 语句。

下面是一则关于雅虎的 SQL 注入事件，当然这次的事件也给广大厂商和用户敲响了安全警钟。

SQL 注入安全事件：

新浪科技讯 北京时间 2012 年 7 月 14 日凌晨消息，雅虎已经成为一个安全漏洞的受害者，导致数十万个用户密码泄露，但看起来用户在自我保护方面做得也不够多。

雅虎的泄露数据已经被发到黑客网站 D33D 上，其中包括 45.3 万个用户密码。黑客称，他们利用 SQL 注入技术渗透雅了虎的子域。

业界人士指出，如果能从这个安全漏洞中学到一件事情，那就是用户需要在设置密码的问题上更有创造性。黑客表示，他们希望此次事件能成为一个警报信号，让网站关注自身安全性的问题；与此同时，个人用户也应将此作为一个警报信号来增强自身的个人密码。雅虎网站上最流行的密码就是“123456”，目前已经发现了 2295 个例子。

当然，如果想要深入的了解 SQL 注入，那么就应该要知道他的一些手法，也就是所谓的攻击技巧。

一般来说，SQL 注入的方式分为两种：一种是盲注，还有一种是显注。 在很多的时候，Web 服务器关闭了错误的提示，就是为了不让攻击者轻易的进行注入，于是盲注因此而诞生。所谓的“盲注”，就是在服务器没有返回错误信息的时候进行的注入。

最简单的盲注就是尽量自己构造一些简单的条件语句，然后根据页面返回的信息来判断 SQL 语句是否能够顺利的执行。

当然当我们找到并且确定一个注入点时，我们可以使用运行在 windows 下 python 环境中的 sqlmap 工具，也可以使用 kail 系统中自带的 sqlmap。

sqlmap 支持五种不同的注入模式，我们可以根据自己的实际需求来针对性的使用这款工具：

1、基于布尔的盲注，即可以根据返回页面判断条件真假的注入。

2、基于时间的盲注，即不能根据页面返回内容判断任何信息，用条件语句查看时间延迟语句是否执行（即页面返回时间是否增加）来判断。

3、基于报错注入，即页面会返回错误信息，或者把注入的语句的结果直接返回在页面中。

4、联合查询注入，可以使用 union 的情况下的注入。 5、堆查询注入，可以同时执行多条语句的执行时的注入。

一旦有符合 sqlmap 可以注入的条件，那么就可以将整个 url 扔进 sqlmap 进行注入，如果想知道详细的情况可以用-v3 来获得，当然 sqlmap 中还有好多命令，这里就不一一赘述了。

### 3.3 网络钓鱼

#### 3.3.1. 什么是网络钓鱼攻击？

网络钓鱼就是伪造一些政府或者企业机构，甚至是银行等向用户发送一些欺骗性的电子邮件，并且一步一步的诱导被害人将自己的敏感信息泄露出来，如：用户名，密码等重要的信息。然而通常最普遍而且最不容易使人察觉的往往是那些与原网站相差不多，几乎一模一样的网站，从而制造一个假象，使用户产生错觉，从而在该网站上留下那些敏感数据，更重要的是这种攻击往往不会让受害者察觉。

网络钓鱼主要的一些手常见手段如下： ① 欺骗性的电子邮件和伪造的 Web

站点 ② 攻陷服务器 - 加入恶意脚本

③ 架设钓鱼网站 - 目标:知名金融机构及商务网站 ④ 发送大量欺骗性垃圾邮件

⑤ 滥用个人敏感信息：资金转账 - 经济利益，冒用身份 - 犯罪目的 3.3.2. 网络钓鱼的特点

① 虚假性：钓鱼攻击者一般都是利用自己制作的网站来模仿那些官网，别且用了一些非常相似的域名来增加真实性。

② 针对性：一般与钓鱼攻击者息息相关的企业都是一些银行或者商业机构，购物网站等等。因为这都涉及到大量的网络资金流动。

③ 多样性：网络钓鱼是一种针对人性的弱点来攻击的，钓鱼者不会仅仅只是用钓鱼邮件，网站钓鱼，还有其他很多很多。

④ 可识别性：网络钓鱼并不是防不胜防，因为有些真实网站会有自己的一些独特的资源，比如说数字证书，专属的域名等，这些都是无法伪造出来的。

**3.3.3. 常见的类型**

恶意软件的钓鱼： 引入电子邮件的附件，如从网上下载的恶意软件，或者利用已知的漏洞例如 word、excel 一些办公软件，甚至是一些做了后门的免费 vpn 等 (劫持流量)

网页木马：通过攻陷的网站插入恶意脚本、恶意插件收集用户信息、凭据等 系统重构：修改用户的 PC 用户恶意目的的设置，例如修改收藏夹里的 url 为恶意的钓鱼网站，例如某银行网址 bankibc.com 为 banikcb.com、 给用户浏览器添加外部 Web 代理(一种中间人攻击另外讨论)、修改系统 hosts 文件、及 DNS 的配置导向钓鱼网站、这样用户的信息就会被窃取

DNS 劫持:修改本地 dns 或 hosts 文件或篡改域名系统把用户导向钓鱼网站，其结果是:用户并不知道他们正在进入的机密系统已经被掉包 .eg(攻击域名服务商篡改 DNS 记录、缓存投毒、利用 CSRF 修改路由 dns 解析)

重定向钓鱼：url 重定:

如:http://example.com/example.php?url=http://malicious.example.com

网络流量重定向:在获得服务器权限以后将指定端口重定向到另外一个地址 中间人钓鱼:包括 Wifi 伪热点、SSL 中间人钓鱼、ARP 等、伪基站钓鱼 搜索引擎钓鱼：创建富有吸引力的站点、利用搜索引擎竞价排名，致钓鱼网站置顶。(例如虚假的银行提供较低的信贷成本更高的利率、电商提供较低的产品价格)

# 4. 攻击原理

### 4.1 XSS 跨站攻击原理

XSS 并不是属于主动攻击的手段。攻击者首先要构造一个具有跨站的页面，利用各种各样的方式让用户来浏览这个页面，触发对被攻击站点的页面请求。此时，如果受害者已经在被攻击站点登录，那么攻击者就会持有该站点 cookie。这样该站点会认为受害者发起了一个页面请求。而实际上这个请求是在受害者不知情的情况下发起的，由此攻击者在一定程度上达到了冒充被攻击者的目的。

### 4.2 SQL 注入攻击原理

SQL 注入攻击指的是构造恶意的 SQL 语句，并且插入到 Web 应用程序，而这些恶意的 SQL 语句都是 SQL 语法里的一些组合，通过执行恶意的 SQL 语句从而完成攻击者所要的操作，所以被攻击的主要原因是程序没有细致地过滤用户输入的数据，致使非法数据侵入系统。

根据相关技术原理：SQL 注入可以分为平台层注入和代码层注入。平台注入由不安全的数据库配置或数据库平台的漏洞所致；代码层注入主要是因为程序员

没有对输入的数据进行编码和过滤，所以执行了非法的数据查询。由此可以总结出，SQL 注入的产生原因一般有以下的几方面：①不当的类型处理；②不安全的数据库配置；③不合理的查询集处理；④不当的错误处理；⑤转义字符处理不合适；⑥多个提交处理不当。

### 4.3 钓鱼网站攻击原理

通过一系列的手段将钓鱼网站发布在互联网上，吸引不知情的用户进行访问、输入隐私信息等操作，然后窃取用户的个人信息，严重的还可能进行网络敲诈活动，给用户利益造成非常严重的损害。由于这种攻击方式类似于日常的钓鱼活动，因此通常被称为“钓鱼网站”。

然而就在近日，根据 360 安全中心统计，我国钓鱼网站新增 98 万家，同比增长 95.9%，其中有关网购类钓鱼网站达 39.27 万 家，占年新增钓鱼网站总量的 40.8%。而曾一度被认为危害最高的网购木马却日趋衰落，钓鱼网站已取代了木马成网络安全头号杀手。

钓鱼网站具体表现是：

1. 页面制作

钓鱼网站首先要模拟出以假乱真的网站，并且诱使用户登录。大多数的网页一般都含有看似复杂的文字、视频、一系列的链接以及 Flash 动画等等，但是这些内容可以通过浏览器的“文件另存为”功能来获取。即使有些内容无法获取到也没有关系，因为没有人会清楚地记得网页上什么位置会显示哪些内容。所以黑客往往将制作好的网页放到一个可以被公开访问的服务器上，这样任何人都可以通过互联网访问到这个页面。

2. 后台技术

钓鱼网站的目的是要获取到用户的个人敏感数据，所以如何捕获用户在网页上的输入是关键一步。通常，黑客在获取到网页内容后，网页代码会根据自己的需求来进行修改。正常的网页会将用户的输入传送到后台数据库，并进行校验动作，而黑客则不需要校验这，只需要将用户输入传送到特定的后台即可。这里的“后台”指的是黑客用来存放数据的一个容器，可以是文档，也可以是数据库，还可以利用特定程序将用户输入的内容通过电子邮件发送到黑客的电子信箱中。当然极少数的黑客会利用操作系统的漏洞，植入木马程序到目标机中，当在用户浏览钓鱼网站时，就会自动的记录用户在输入时的键盘记录，并且会直接发送给黑客。

然而网络钓鱼的方法也是很简单的，主要就是看用户的安全意识。因为这种攻击防不胜防，不经意间就走上了钓鱼者铺好的路上。

## 5. Web 安全防护

有攻击就有防护，对于每一种攻击的方式，都可以有相应的安全防护。于是，对于 XSS 攻击的防护也应运而生。

上章只是初步的针对具体的案例所提出来的防范措施，接下来就这些攻击而言，有个总体的防护措施，可以根据实际情况选择不同的应对措施。

### 5.1 XSS 跨站的安全防护

1. 阻止攻击者利用在网站上发布任意攻击语句，首先代码里要对用户输入的地方需要仔细检查长度和对”<”,”>”,”;”,”’”等字符做过滤；其次任何内容被写入到页面之前都必须进行转义，以免有遗漏。如果这一方面做好，那么至少可以防范超过一半的 XSS 攻击。

2. Cookie 防盗

第一，避免直接在 cookie 中泄露用户隐私。

第二，通过用 cookie 和系统 IP 绑定来降低 cookie 泄露后的麻烦。这样，就算攻击者得到 cookie，也没有任何的实际价值。

3. 尽量采用 POST 而非 GET 提交表单

POST 操作是不可能绕开 javascript 的使用，这会给攻击者增加不少的难度，减少可利用的跨站漏洞。

4. 将单步操作改为多步，在多步操作中引入效验码。

多步操作中每一步都产生一个验证码作为表单元素嵌在中间页面，下一步操作时这个验证码被提交到服务器，服务器检查这个验证码是否匹配。

5. 引入用户交互简单的一个看图识数可以堵住几乎所有的非预期特权操作。 6. 只在允许 anonymous 访问的地方使用动态的 javascript。

7. 对于用户提交信息的中的内容，检查是否会重定向回本站、是不是真的图片、视频，排除一切的可疑操作。

9. 内部管理网站的问题很多时候，内部管理网站往往疏于关注安全问题，只是简单的限制访问来源。这种网站往往对 XSS 攻击毫无抵抗力，需要多加注意。

频，排除一切的可疑操作。

9. 内部管理网站的问题很多时候，内部管理网站往往疏于关注安全问题，只是简单的限制访问来源。这种网站往往对 XSS 攻击毫无抵抗力，需要多加注意。

**5.2 SQL 注入的安全防护**

1.永远不要信任用户的输入。对用户的输入进行校验，可以通过正则表达式，或限制长度；也可以对单引号和双"-"进行转换等。

2.千万不要使用动态拼装 sql，可以使用参数化的 sql 或者直接使用存储过程

进行数据的查询，存储或者读取。

3.永远不要使用管理员权限进行数据库连接，为每个应用使用单独权限连接数据库。

4.不要把机密信息直接存放在数据库中，最好是要加密或者 hash 掉密码和敏感的数据信息。

5.应用返回的异常信息应该给出尽可能少的提示，最好使用自定义的错误返回信息对原始错误信息进行掩盖。

6.sql 注入的检测方法一般采取辅助软件或者网站平台来检测，软件一般采用 sql 注入检测工具 sqlmap，能够有效的进行 SQL 注入。

**5.3 钓鱼网站的安全防护**

第一、核对网站域名

假冒网站和真实网站有细微区别，有疑问时要仔细辨别其不同之处，比如在域名方面，假冒网站通常将英文字母被替换为数字，这样的仿造域名，一般粗心的用户是完全察觉不出来的。

第二、比较网站内容

假冒网站上的字体样式不一致，并且模糊不清。而且一般仿冒网站的上没有可以点击链接，用户可自主选择点击栏目或图片中的各个链接看是否能打开。

第三、查看安全证书

大型的电子商务网站都有安全证书。所以这类的网站网址大多数都是以“https”开头的，如果发现不是“https”开头，应谨慎对待。

我相信，如果大家都能够掌握这些防护措施，那么我们被攻击的几率将会大大的减少，这样也保证了我们自己的财产安全。