1. Cookie without HttpOnly flag set

如果在Cookie上设置了HttpOnly属性，则客户端JavaScript无法读取或设置Cookie的值。 这种措施通过阻止某些客户端攻击（例如跨站点脚本），通过阻止它们通过注入的脚本来简单地捕获cookie的值，使其利用起来略为困难。

1. robots.txt file

robots.txt的存在本身并不表示任何类型的安全漏洞。 但是，它通常用于标识站点内容的受限区域或私有区域。 因此，文件中的信息可以帮助攻击者确定站点的内容，尤其是在某些未标识的位置未从站点其他位置链接的情况下。 如果应用程序依靠robots.txt保护对这些区域的访问，并且未对它们进行强制访问控制，则这将带来严重的漏洞。

Frameable response (potential Clickjacking) （框架响应（潜在的点击劫持））

攻击者控制的网页可能会在攻击者页面的iframe中加载此响应的内容。 这可能会启用“点击劫持”攻击，其中攻击者的页面将目标应用程序的界面与攻击者提供的其他界面覆盖在一起。 通过诱使受害者用户执行诸如鼠标单击和击键之类的操作，攻击者可以使他们在目标应用程序内不经意地执行操作。 该技术允许攻击者规避针对跨站点请求伪造的防御，并可能导致未经授权的行为

1. User agent-dependent response

应用程序的响应似乎系统地取决于请求中User-Agent标头的值。 此行为本身并不构成安全漏洞，但可能指向应用程序中可能包含漏洞的附加攻击面。 因为应用程序为台式机和移动用户提供了不同的用户界面，所以经常会出现这种现象。 移动接口通常没有经过针对漏洞的全面测试，例如跨站点脚本，并且通常具有更简单的身份验证和会话处理机制，其中可能包含完整接口中不存在的问题。 要查看备用User-Agent标头提供的界面，您可以在Burp Proxy中配置匹配/替换规则，以修改所有请求中的User-Agent标头，然后以正常方式使用浏览器浏览应用程序。

跨站点脚本攻击

1. Cross Site Scripting Flaws

这是一种常见的攻击，当攻击脚本被嵌入企业的Web页面或其它可以访问的Web资源中，没有保护能力的台式机访问这个页面或资源时，脚本就会被启动，这种攻击可以影响企业内成百上千员工的终端电脑。

1. 缓存溢出问题（Buffer Overflows）

这个问题一般出现在用较早的编程语言、如C语言编写的程序中，这种编程错误其实也是由于没有很好地确定输入内容在内存中的位置所致。

1. 注入式攻击（Injection Flaws）

如果没有成功地阻止带有语法含义的输入内容，有可能导致对数据库信息的非法访问，在Web表单中输入的内容应该保持简单，并且不应包含可被执行的代码。

1. 异常错误处理（Improper Error Handling）

当错误发生时，向用户提交错误提示是很正常的事情，但是如果提交的错误提示中包含了太多的内容，就有可能会被攻击者分析出网络环境的结构或配置。

1. 不安全的存储（Insecure Storage）

对于Web应用程序来说，妥善保存密码、用户名及其他与[身份验证](https://wenwen.sogou.com/s/?w=%E8%BA%AB%E4%BB%BD%E9%AA%8C%E8%AF%81&ch=ww.xqy.chain" \t "_blank)有关的信息是非常重要的工作，对这些信息进行加密则是非常有效的方式，但是一些企业会采用那些未经实践验证的加密解决方案，其中就可能存在安全漏洞。

1. 程序拒绝服务攻击（Application Denial of Service）

与拒绝服务攻击 (DoS)类似，应用程序的[DoS攻击](https://wenwen.sogou.com/s/?w=DoS%E6%94%BB%E5%87%BB&ch=ww.xqy.chain)会利用大量非法用户抢占应用程序资源，导致合法用户无法使用该Web应用程序。

1. 不安全的配置管理（Insecure Configuration Management）

有效的配置管理过程可以为Web应用程序和企业的网络架构提供良好的保护

**Content type incorrectly stated**

内容类型错误

**Email addresses disclosed**

邮件地址泄漏