此文档的原文是：<http://drops.wooyun.org/papers/58>  因内容较详细和具有指导性，但担心网站后续不提供服务，就将内容粘贴上来，供学习。

0x00 相关背景介绍

由于应用越来越多的需要和其他的第三方应用交互，以及在自身应用内部根据不同的逻辑将用户引向到不同的页面，譬如一个典型的登录接口就经常需要在认证成功之后将用户引导到登录之前的页面，整个过程中如果实现不好就可能导致一些安全问题，特定条件下可能引起严重的安全漏洞。

0x01 成因

对于URL跳转的实现一般会有几种实现方式：

META标签内跳转 javascript跳转 header头跳转

通过以GET或者POST的方式接收将要跳转的URL，然后通过上面的几种方式的其中一种来跳转到目标URL。一方面，由于用户的输入会进入Meta，javascript，http头所以都可能发生相应上下文的漏洞，如xss等等，但是同时，即使只是对于URL跳转本身功能方面就存在一个缺陷，因为会将用户浏览器从可信的站点导向到不可信的站点，同时如果跳转的时候带有敏感数据一样可能将敏感数据泄漏给不可信的第三方。

譬如一个典型的登录跳转如下：

<?php

$url=$\_GET['jumpto'];

header("Location: $url");

?> 如果jumpto没有任何限制，所以恶意用户可以提交

<http://www.wooyun.org/login.php?jumpto=http://www.evil.com> 来生成自己的恶意链接，安全意识较低的用户很可能会以为该链接展现的内容是[www.wooyun.org](http://www.wooyun.org/)从而可能产生欺诈行为，同时由于QQ，淘宝旺旺等在线IM都是基于URL的过滤，同时对一些站点会一白名单的方式放过，所以导致恶意URL在IM里可以传播，从而产生危害，譬如这里IM会认为[www.wooyun.org](http://www.wooyun.org/)都是可信的，但是通过在IM里点击上述链接将导致用户最终访问evil.com。

0x02 攻击方式及危害

恶意用户完全可以借用URL跳转漏洞来欺骗安全意识低的用户，从而导致“中奖”之类的欺诈，这对于一些有在线业务的企业如淘宝等，危害较大，同时借助URL跳转，也可以突破常见的基于“白名单方式”的一些安全限制，如传统IM里对于URL的传播会进行安全校验，但是对于大公司的域名及URL将直接允许通过并且显示会可信的URL，而一旦该URL里包含一些跳转漏洞将可能导致安全限制被绕过。

如果引用一些资源的限制是依赖于“白名单方式”，同样可能被绕过导致安全风险，譬如常见的一些应用允许引入可信站点如youku.com的视频，限制方式往往是检查URL是否是youku.com来实现，如果youku.com内含一个url跳转漏洞，将导致最终引入的资源属于不可信的第三方资源或者恶意站点，最终导致安全问题。

0x03 实际案例

a WooYun: <http://www.wooyun.org/bugs/wooyun-2010-0104> 百度一个URL跳转漏洞

通过url跳转突破IM的安全校验，从而传播恶意URL

b WooYun:<http://hi.baidu.com/breachme/item/c227a49c9019d4d97a7f0179>  第五次遇到利用淘宝网网址跳转的网钓

通过url跳转欺骗交易用户对URL的信任，从而实现欺诈

c <http://hi.baidu.com/rayh4c/blog/item/8fde4b23ffa2045e9822edb9.html> 暴一个百度X吧跨站的细节

通过URL跳转绕过应用程序对引用资源的限制，从而导致安全漏洞

d WooYun: <http://www.wooyun.org/bugs/wooyun-2010-0198> 豆瓣电台认证绕过及csrf防范策略绕过漏洞

借助未验证的URL跳转，可以将应用程序内部敏感的数据传递到不安全的第三方区域

0x04 修复方案

理论上讲，url跳转属于CSRF的一种，我们需要对传入的URL做有效性的认证，保证该URL来自于正确的地方，限制的方式同防止csrf一样可以包括：

1 referer的限制

如果确定传递URL参数进入的来源，我们可以通过该方式实现安全限制，保证该URL的有效性，避免恶意用户自己生成跳转链接

2 加入有效性验证Token

我们保证所有生成的链接都是来自于我们可信域的，通过在生成的链接里加入用户不可控的Token对生成的链接进行校验，可以避免用户生成自己的恶意链接从而被利用，但是如果功能本身要求比较开放，可能导致有一定的限制。