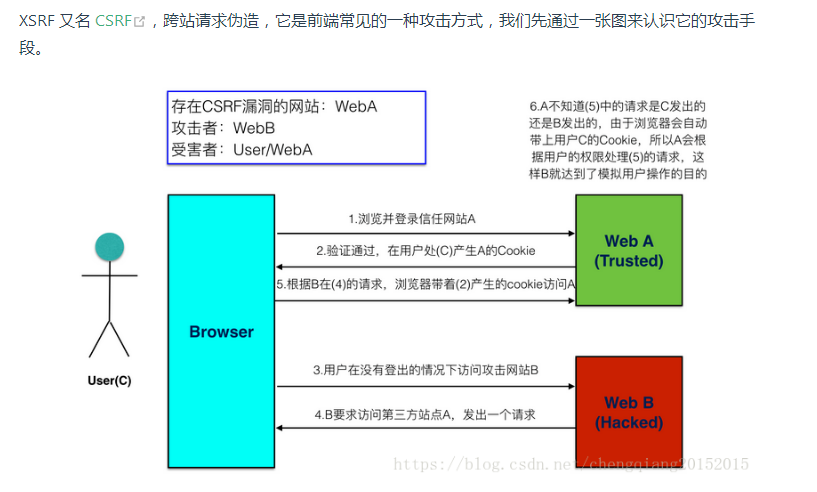
**xsrf**: 跨站请求伪造(Cross-site request forgery)



防御：token，Referer

CSRF 的防御手段有很多，比如验证请求的 referer，但是 referer 也是可以伪造的，所以杜绝此类攻击的一种方式是服务器端要求每次请求都包含一个 token，这个 token 不在前端生成，而是在我们每次访问站点的时候生成，并通过 set-cookie 的方式种到客户端，然后客户端发送请求的时候，从 cookie 中对应的字段读取出 token，然后添加到请求 headers 中。这样服务端就可以从请求 headers 中读取这个 token 并验证，由于这个 token 是很难伪造的，所以就能区分这个请求是否是用户正常发起的。

攻击细节

跨站请求攻击，简单地说，是攻击者通过一些技术手段欺骗用户的浏览器去访问一个自己曾经认证过的网站并运行一些操作（如发邮件，发消息，甚至财产操作如转账和购买商品）。由于浏览器曾经认证过，所以被访问的网站会认为是真正的用户操作而去运行。这利用了web中用户身份验证的一个漏洞：**简单的身份验证只能保证请求发自某个用户的浏览器，却不能保证请求本身是用户自愿发出的**。

**例子**

假如一家银行用以运行转账操作的URL地址如下：http://www.examplebank.com/withdraw?account=AccoutName&amount=1000&for=PayeeName

那么，一个恶意攻击者可以在另一个网站上放置如下代码： <img src="http://www.examplebank.com/withdraw?account=Alice&amount=1000&for=Badman">

如果有账户名为Alice的用户访问了恶意站点，而她之前刚访问过银行不久，登录信息尚未过期，那么她就会损失1000资金。

这种恶意的网址可以有很多种形式，藏身于网页中的许多地方。此外，攻击者也不需要控制放置恶意网址的网站。例如他可以将这种地址藏在论坛，博客等任何[用户生成内容](https://baike.baidu.com/item/%E7%94%A8%E6%88%B7%E7%94%9F%E6%88%90%E5%86%85%E5%AE%B9)的网站中。这意味着**如果服务端没有合适的防御措施的话，用户即使访问熟悉的可信网站也有受攻击的危险**。

透过例子能够看出，攻击者并不能通过CSRF攻击来直接获取用户的账户控制权，也不能直接窃取用户的任何信息。他们能做到的，是**欺骗用户浏览器，让其以用户的名义运行操作**。

XSS: 跨站脚本攻击（Cross Site Scripting）

跨站脚本攻击（XSS），是最普遍的Web应用安全漏洞。这类漏洞能够使得攻击者嵌入恶意脚本代码到正常用户会访问到的页面中，当正常用户访问该页面时，则可导致嵌入的恶意脚本代码的执行，从而达到恶意攻击用户的目的。-------------恶意的代码注入