跨站请求伪造攻击实验

一【实验目的】

- 了解跨站请求伪造攻击的概念及攻击原理
- 理解它与跨站脚本攻击的区别
- 掌握跨站请求伪造攻击的具体防范措施

二【实验环境】

- 操作机和靶机：Windows 10 操作机
- 网络拓扑结构：单一操作机
- 访问方式：本地访问操作机

三【实验原理】

CSRF（Cross-Site Request Forgery，跨站请求伪造）攻击，指攻击者盗用用户身份，通过伪装来自受信任用户的请求来利用受信任的网站，以被盗用户的名义发送邮件、发表评论等。

攻击原理：

1、用户访问受信任站点 A，并在浏览器中产生相关的 cookie。

2、用户在不退出站点 A 的情况下访问了危险站点 B，B 站点收到用户请求后返回一些攻击性的代码要求访问站点 A。

3、浏览器在用户不知情的情况下携带 cookie 信息向站点 A 发出请求，站点 A 根据用户的 cookie 信息以用户的权限处理请求

4、站点 B 就将自己伪造成用户身份登录站点 A，导致站点 B 的恶意代码可以被执行。

跨站请求伪造攻击的防范方法：

1、验证 HTTP Referer。在 HTTP 协议的请求头部含有一个字段 Referer，它记录了本次请求的来源地址。可以通过校验 Referer 是否以本域作为来源来判断请求的真伪。

2、加密 cookie 信息。在敏感操作的提交内容中，添加一个对 cookie 进行 Hash 后的值，服务器端对 Hash 值进行校验，若通过则是合法的用户请求。

3、使用令牌。添加一个隐藏表单域记录随机的令牌，在求的参数中包含该令牌。服务器端执行操作前验证这个令牌，如果请求中没有令牌或者内容不正确，则认为可能是伪造请求攻击而拒绝该请求。

四【实验步骤】

（1）打开【tools\跨站请求伪造攻击】文件夹，如图 1 所示。

图 1

（2）右击【PhpStudyLite】压缩文件，选择【全部解压缩】，解压路径为默认路径。进入安装路径下，双击【phpStudy Lite】应用程序，启动 phpStudy。如图 2 所示。

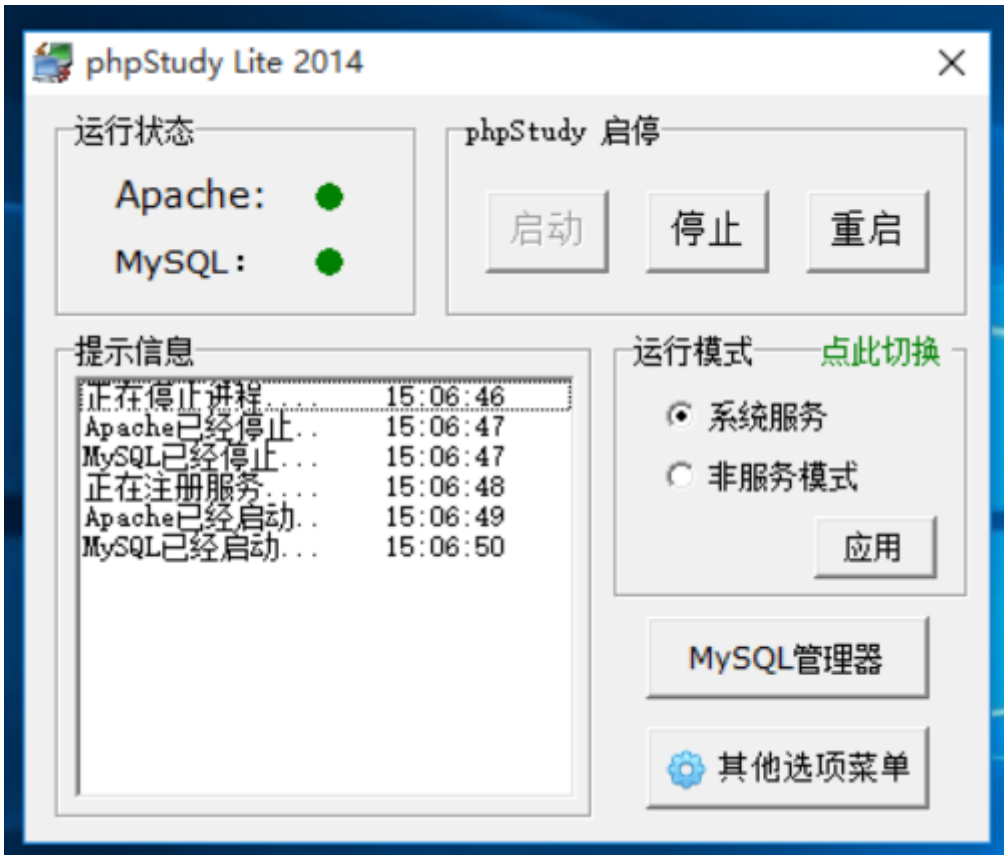

图 2

（3）打开浏览器，在地址栏中输入【http://127.0.0.1】 ，可以看到 phpStudy 探针的页面，说明 phpStudy 启动且运行正常。如图 3 所示。

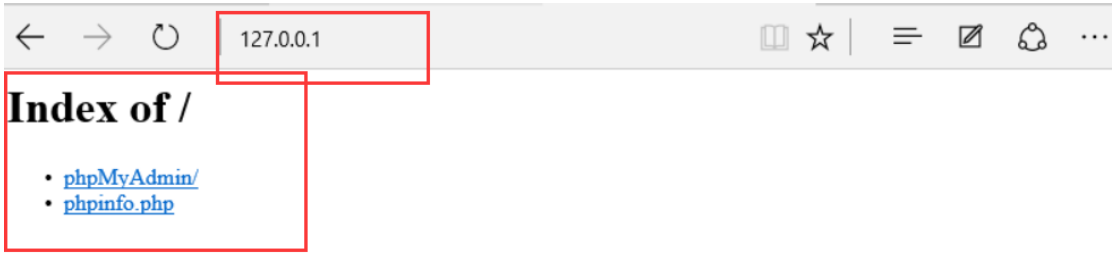

图 3

（4）将【跨站脚本伪造攻击】文件夹下的【CSRF】文件夹复制到 phpStudy 安装路径的【WWW】文件夹下。如图 4 所示。

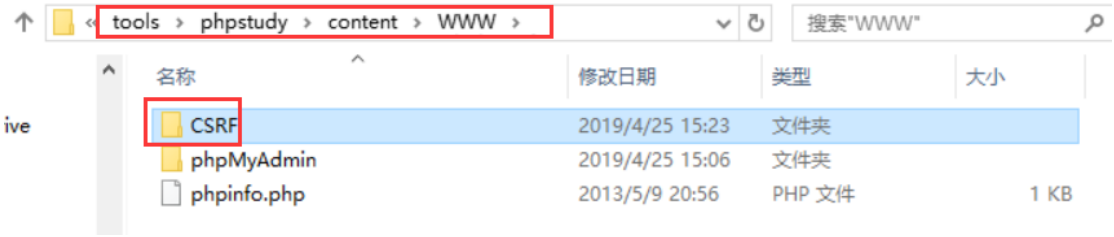

图 4

（5）打开 phpStudy 界面，在其中选择“MySQL 管理器”，打开 phpMyAdmin 数据库管理界面，根据数据库的默认设置，用户名和密码均为【root】，输入后登录。如图 5 所示。

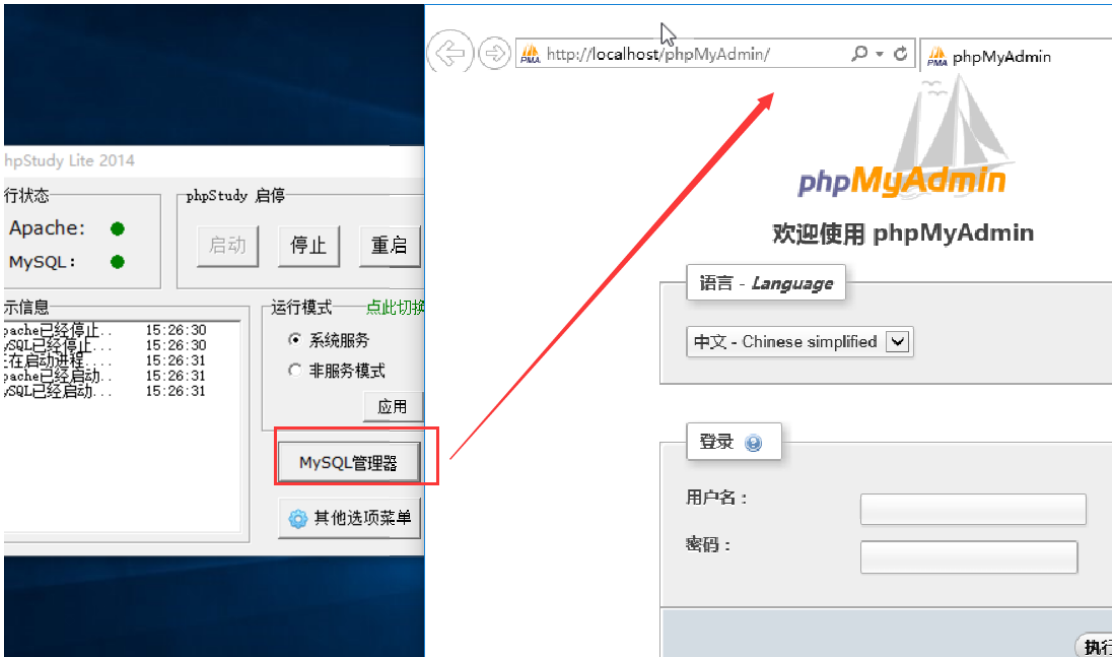

图 5

（6）点击【数据库】选项卡，在【新建数据库】处输入数据库名【bbs】，点击创建按钮。如图 6 所示。

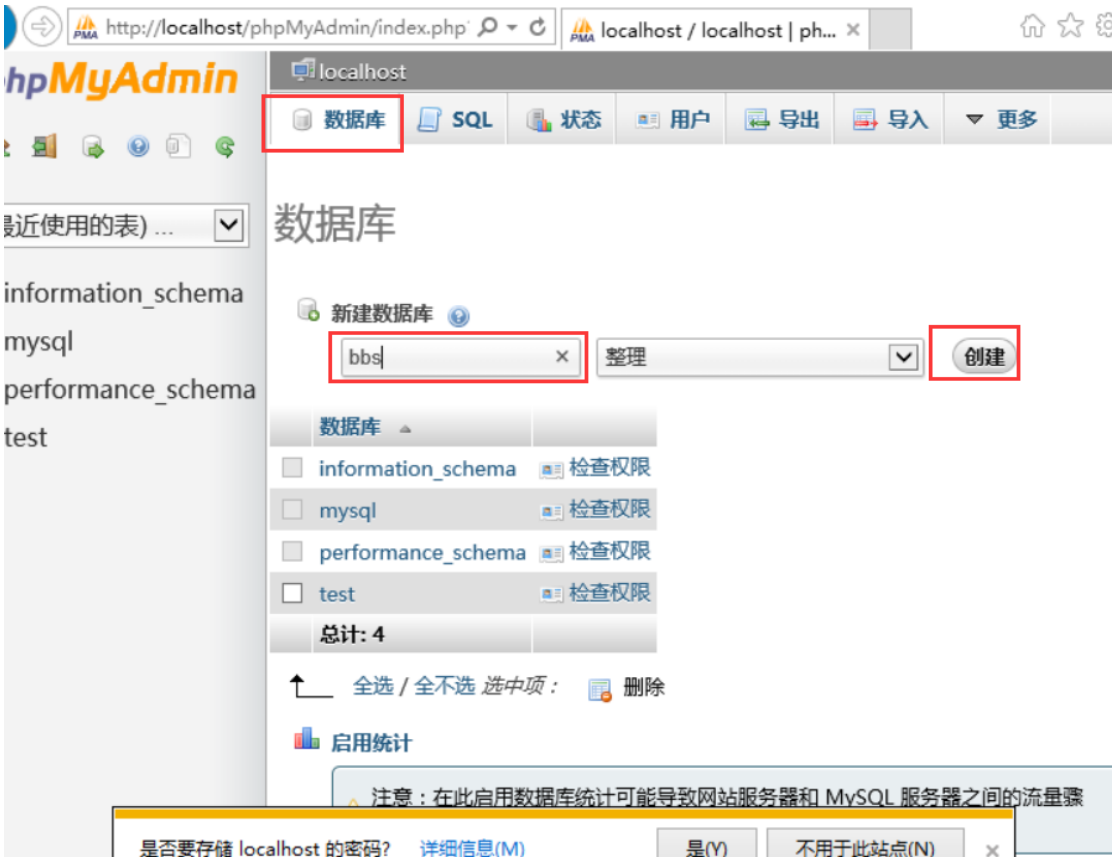

图 6

（7）成功创建新数据库后，在左边的数据库列表中找到新建的数据库【bbs】，点击进入。如图 7 所示。

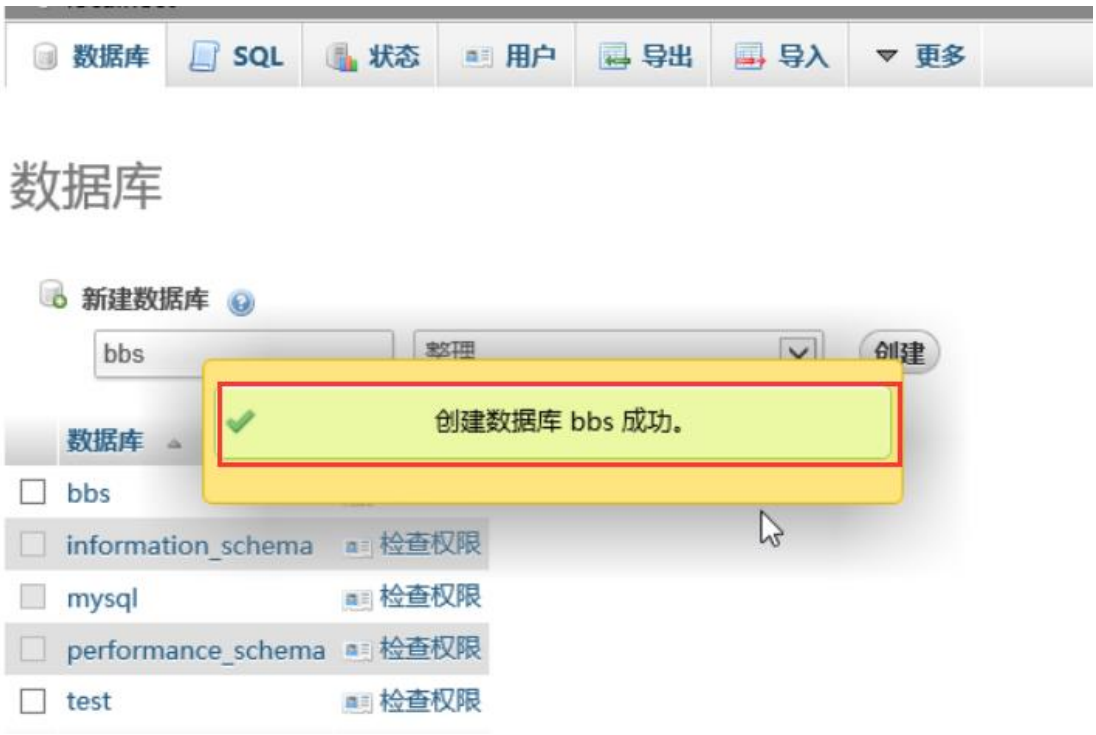

图 7

（8）在页面上方菜单栏中选择【导入】选项卡，在【要导入的文件】中选择【从计算机中上传】的【浏览】按钮，在 phpStudy 的安装目录下的【WWW】文件夹中找到【CSRF】文件夹，并进入其中的【PHP】文件夹下，选择【postmessage.sql】数据库文件，点击页面底部的【执行】按钮。如图 8 所示。

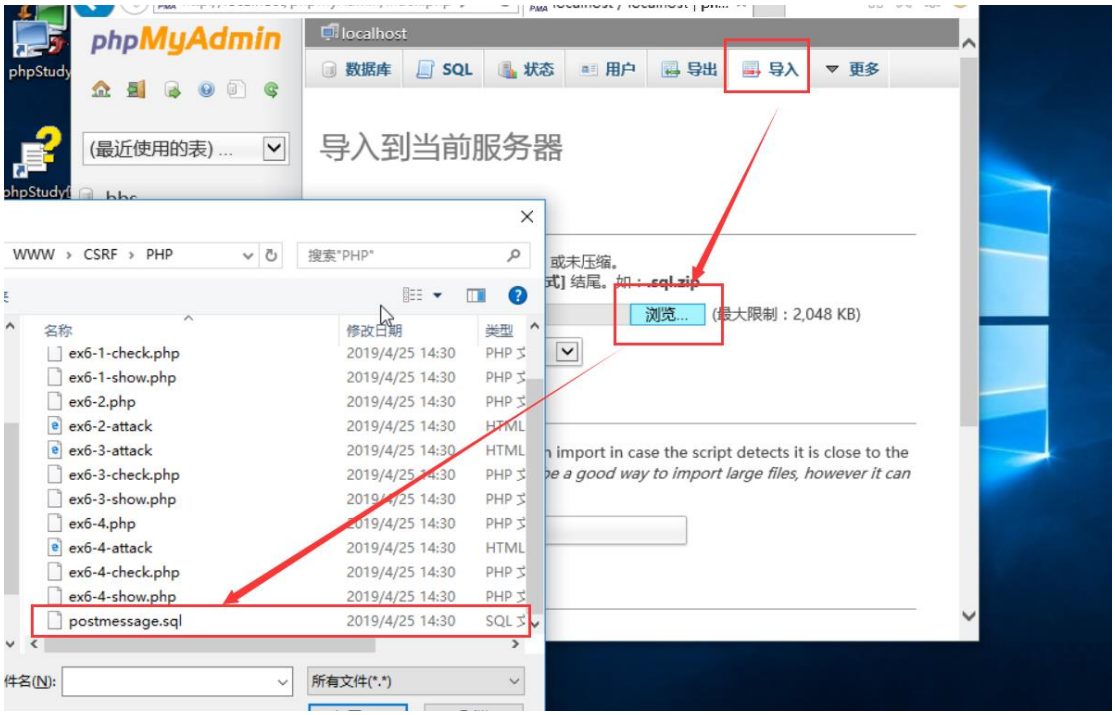

图 8

（9）打开 IE 浏览器，在地址栏输入【http://127.0.0.1/CSRF】，进入模拟演示网页。如图 9 所示。

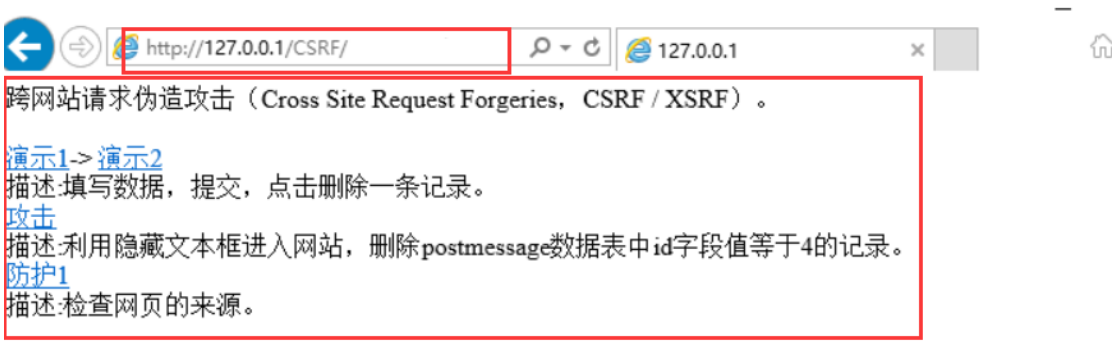

图 9

（10）点击【演示 1】，打开一个简单形式的留言板。如图 10 所示。

图 10

（11）填写留言信息后，点击【确认】会将留言的数据插入到 MySQL 数据库 ch 里的 postmessage 数据表中。如图 11 所示。

图 11

（12）留言后可以查看所有留言内容。如图 12 所示。

图 12

（13）用户在还没有单击【删除】按钮时，代码中【delete】的值为 NULL，因此删除按钮的程序代码不会执行。当单击了【删除】按钮后，会对同一个文件送出资料，因此【删除】按钮的程序代码会被执行。如图 13 所示。

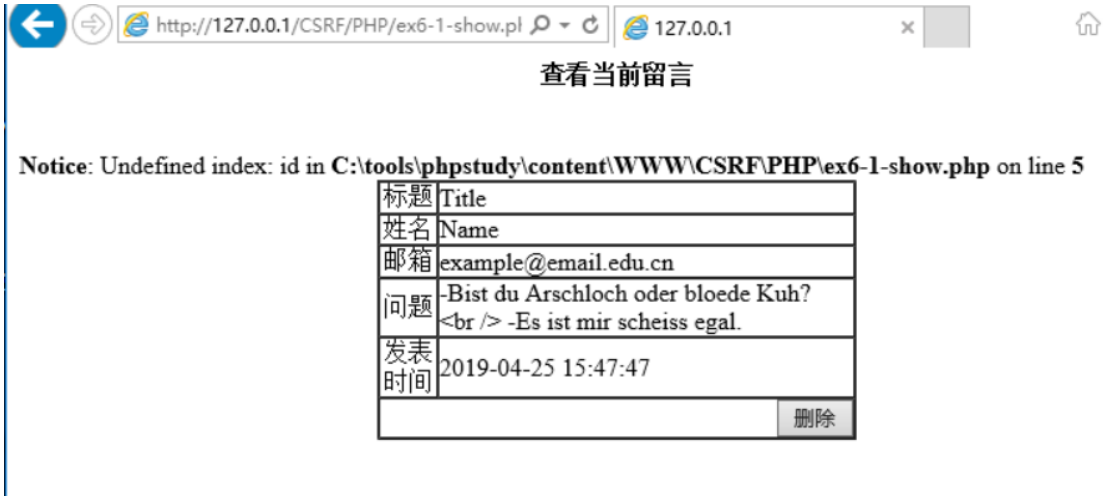

图 13

（14）回到首页点击【攻击】，postmessage 数据表中制定 id 的留言被删除。如图 14 所示。

图 14

（15）打开 ex6-1-attack.html 文件查看源码。在 ex6-1-attack.html 文件的表单中，我们将 action 属性值设置为 ex6-1-show.php 文件。所以单击表单的 submit 按钮后，就会执行 ex6-1-show.php 文件。如图 15 所示。

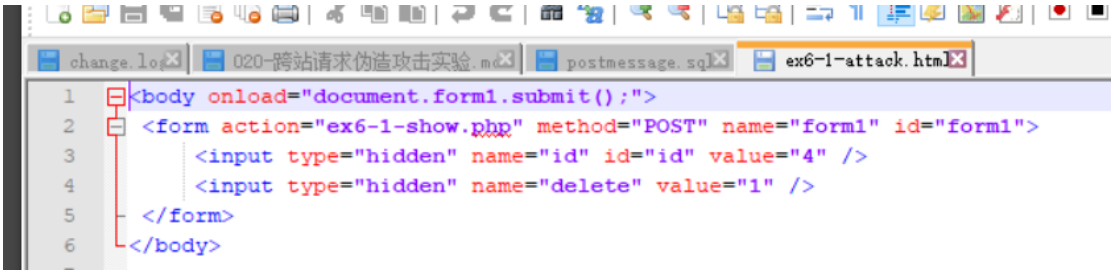

图 15

五【实验总结】

通过此次实验，了解了跨站请求伪造攻击的基本原理，通过具体的实验实现跨站请求伪造攻击并根据攻击过程分析相应的防御措施

思考：

试描述 CSRF 和 XSS 攻击的具体过程，并总结两种攻击方式的异同点。

答：

CSRF 攻击过程：

1. 用户 C 打开浏览器，访问受信任网站 A，输入用户名和密码请求登录网站 A；

2. 在用户信息通过验证后，网站 A 产生 Cookie 信息并返回给浏览器，此时用户登录网站 A 成功，可以正常发送请求到网站 A；

3. 用户未退出网站 A 之前，在同一浏览器中，打开一个 TAB 页访问网站 B；

4. 网站 B 接收到用户请求后，返回一些攻击性代码，并发出一个请求要求访问第三方站点 A；

5. 浏览器在接收到这些攻击性代码后，根据网站 B 的请求，在用户不知情的情况下携带 Cookie 信息，向网站 A 发出请求。网站 A 并不知道该请求其实是由 B 发起的，所以会根据用户 C 的 Cookie 信息以 C 的权限处理该请求，导致来自网站 B 的恶意代码被执行。

XSS 攻击过程：

1. 黑客准备攻击字符串，构造攻击 URL

2. 用户上当误点击攻击 URL

3. 用户敏感数据被发送到黑客接收网站

4. 黑客利用敏感数据做坏事

异同点：

CSRF：需要用户先登录网站 A，获取 cookie。

XSS：不需要登录。

CSRF：是利用网站 A 本身的漏洞，去请求网站 A 的 api。

XSS：是向网站 A 注入 JS 代码，然后执行 JS 里的代码，篡改网站 A 的内容