Get来的漏洞

小米安全工程师　吕伟
网络ID：呆子不开口

Get方法定义 – Requests data from a specified resource

The GET Method

Note that the query string (name/value pairs) is sent in the URL of a GET request:

```
/test/demo_form.asp?name1=value1&name2=value2
```

Some other notes on GET requests:

- GET requests can be cached
- GET requests remain in the browser history
- GET requests can be bookmarked
- GET requests should never be used when dealing with sensitive data
- GET requests have length restrictions
- GET requests should be used only to retrieve data

Get请求的风险

根据HTTP规范，GET用于信息获取，是安全的和幂等的。所以从get请求被设计和现实场景使用的结果来看，有如下特性

- 可能会被重放
- 到处出现，容易泄露

所以get请求的使用应该遵循

- 不应有增删改的操作
- 不应包含敏感信息

实现不符合别人对你的预期，就可能产生漏洞

- 隐私泄露，csrf，账号被盗……

Get请求可能出现的地方

- 浏览器地址栏
- 被云加速的cdn服务商收集
- 被运营商或网络设备收集重放
- 被网络嗅探
- 用户的收藏夹
- **http协议的referrer中**
- web服务器日志
- 浏览器历史记录
- 搜索引擎爬到，或者不规范收集
- 被用户邮件或微信分享出去
- 各种可能的地方，甚至山岗上田野上……//一个黑客，盗取了get请求后，拷贝在U盘里，路过一个山岗时，被大灰狼吃掉了，U盘掉在了山岗上

用get实现增删改的风险

- 会被重放，导致服务端资源状态发生改变
  - 浏览器的重新打开可能会重放请求
  - 爬虫或安全扫描会重放你的请求
  - 获取到你get请求的各种势力会重放此请求，如安全厂商，搜索引擎 //除了山岗上那个黑客，他已经被狼吃了

- 可能容易形成csrf漏洞，页面有大量发起检索请求的需求，导致referrer信任关系可能被利用，token可能被偷
  - 允许用户发表第三方链接、图片等
  - 存在js端的跳转漏洞跳到第三方
  - Get请求中防护的token更容易被偷

用get传输敏感信息的风险

被偷！

然后！！

被搞！！！

常见的敏感信息

- 隐私信息
  - http://weibo.com/lvwei
- 校验信息
  - https://mp.weixin.qq.com/cgi-bin/home?t=home/index&lang=zh_CN&token=371767643
- 认证信息
  - http://XXX.XXXXXX.XXX/index.php?ticket=**********
  - http://XXX.XXXXXX.XXX/index.php?gsid=**********

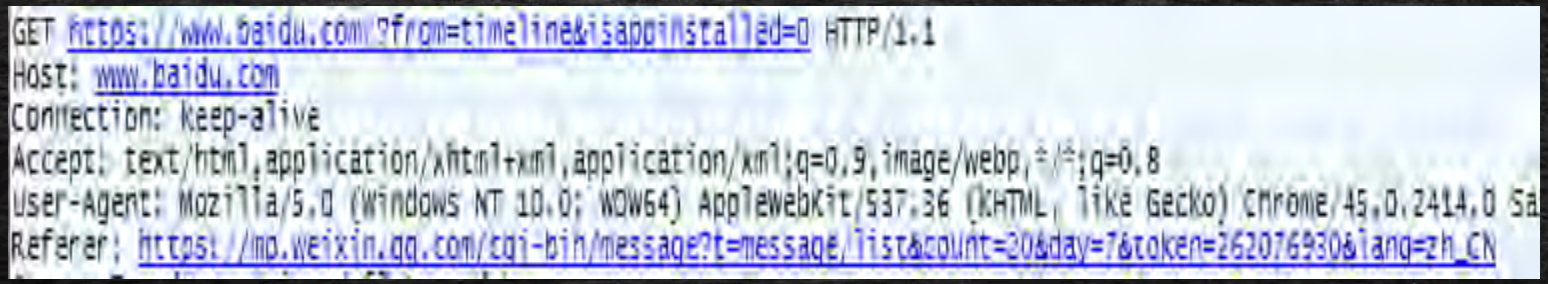
隐私信息泄露举例

微博首页url会有用户ID信息，就是说一条链接的主人会通过referrer知道哪些用户访问了它

但它可能会帮你逮微博马甲、捉奸在网……

如果你发现你的男朋友和你的男同事在凌晨一点，都访问了你发的链接，并且IP一样。这个时候，作为一个男子汉，你可能要考虑下，应该哭多大声才不会吵到邻居……

防伪信息泄露举例



微信公众平台后台的操作大多是post，csrf的防护有token和referrer

但在每个页面的get请求中，也会有这个token

这样的token很容易被referrer偷，见上图，token已经发给了第三方域了。防护体系被削弱了

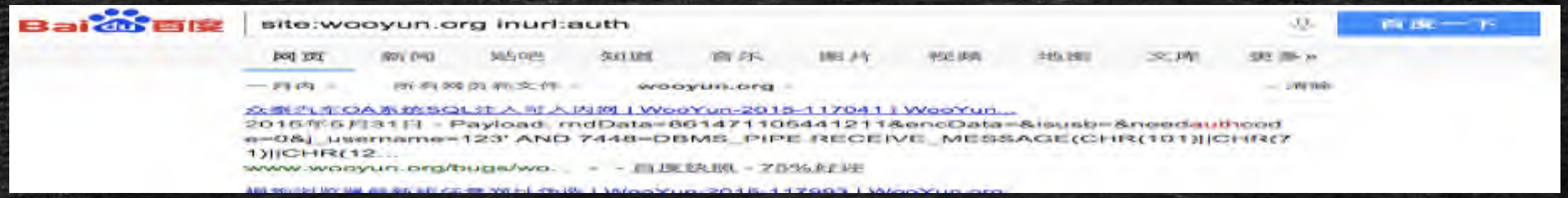
认证信息泄露举例

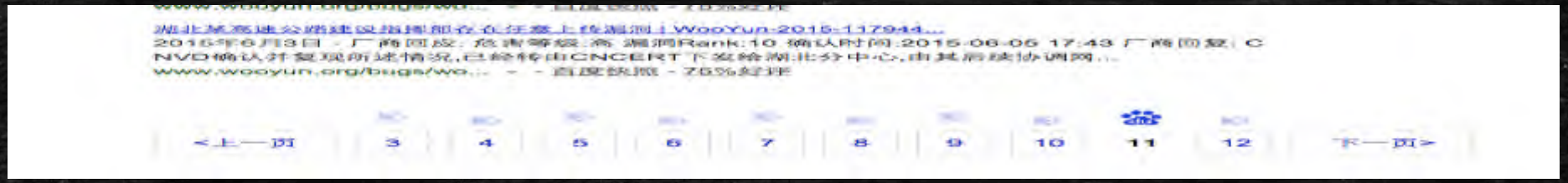
左图是现在的乌云的厂商用户的查看漏洞详情的临时页面，原来是没有查看密码的，是可以通过地址栏里那个含有auth信息的get请求直接查看的

但若一漏洞详情页包含了一个优酷的视频，这个查看详情的链接会在优酷的视频页显示。因为优酷显示了referrer信息，见右图

详情见 http://www.wooyun.org/bugs/wooyun-2010-0102609

认证信息泄露举例





一个月内泄露的乌云厂商用户的临时查看链接十二页，应该不可能全是厂商管理员分享出去的

我有一个猜测，但不一定错：可能被百度云加速收集，用来帮助用户进行搜索的seo优化

最敏感的信息——认证信息

使用get请求认证的一些场景

- App给内嵌页面请求加上认证信息，参数名如sid、gsid
- 单点登陆从sso拿ticket信息，参数名如ticket、auth
- 网站绑定第三方账号登陆，由第三方给的登陆凭证

Xss偷不了httponly的cookie?

- 大站会使用httponly，但大站基本都有单点登录
- 你可以试试偷上面的这些认证信息

referrer

Xss不好找?
用referrer吧
它不产生漏洞
它只是漏洞的搬运工

App给内嵌页面请求加上认证信息，怎么偷

- 当我们在一个app内打开其公司产品的一些链接，会被加上认证信息去让用户自动登陆
  - 微博客户端、QQ客户端、微信客户端都曾有或现在正有此问题
  - 一般会加上参数sid、gsid、key
  - 例子：http://www.wooyun.org/bugs/wooyun-2010-027590
  - 例子：http://www.wooyun.org/bugs/wooyun-2010-070454
  - 例子：之前的一个手机qq的漏洞，找一qq域下论坛发一张图，然后把此页发给手机qq上好友，他点击就会被盗

- 怎么偷
  - 找能发自定义图片的页面
  - 找能发自定义iframe的页面
  - 找一个js端的跳转漏洞
  - 如果只允许白名单域的图片或iframe，那就再找一个302跳转的漏洞
  - Xss漏洞获取地址栏信息
  - 用户甚至会通过app的分享功能把认证信息分享到邮件或朋友圈

通行证或绑定的第三方站的登陆凭证，怎么偷

- 从通行证或第三方获取登陆ticket
  - 形式为类似http://passport.wangzhan.com/sso/getst.php?url=http://wangzhan.cn/a.php
  检验是白名单域后，然后302跳转到
  http://wangzhan.cn/a.php?ticket=XXXXXXXXXXXXXXX
  然后页面服务端使用此ticket去sso验证此用户身份，再在本域种认证cookie
  - 例子 http://www.wooyun.org/bugs/wooyun-2010-0124352
- 怎么偷
  - 找能发自定义图片的页面
  - 找能发自定义iframe的页面
  - 找一个js端的跳转漏洞
  - 如果图片和iframe的src值只允许白名单域的图片或iframe，那就再找一个302跳转的漏洞
  - Xss漏洞获取地址栏信息

通行证或绑定的第三方站的登陆凭证，怎么偷

- 从通行证或第三方获取登陆ticket
  - 形式为类似http://passport.wangzhan.com/sso/login.php?url=http://wangzhan.cn/a.php

  然后302跳转到 http://wangzhan.cn/login.php?ticket=XXXXXXXXXXXXXXXX&url=http://wangzhan.cn/a.php 此时会种上认证cookie

  然后页面会使用js跳转到 http://wangzhan.cn/a.php
  - 例子 某绑定了微博账号后可以自动登陆的网站
- 怎么偷
  - 找一个302跳转的漏洞。因为js的跳转会带referrer，然后再通过302跳转把referrer传给我们能控制的页面
  - Xss漏洞获取referrer

通行证或绑定的第三方站的登陆凭证，怎么偷

- 从通行证或第三方获取登陆ticket
  - 形式为类似http://passport.wangzhan.com/sso/login.php?url=http://wangzhan.cn/a.php

  然后302跳转到 http://wangzhan.cn/login.php?ticket=XXXXXXXXXXXXXXXX&url=http://wangzhan.cn/a.php 此时会中上认证cookie

  然后页面会再302跳转到 http://wangzhan.cn/a.php

- 怎么偷
  - Xss漏洞，使用iframe，种超长cookie，阻止最后的302，然后读取iframe.contentWindow.location.href

跨域从通行证获取到的凭证，怎么偷

- 跨域从通行证获取登陆ticket
  - 形式为类似http://www.wangzhan.com/sso/getst.php?callback=jsonp
  然后通行证会返回个jsonp格式的数据，里面包含认证信息
  - 例子 http://www.wooyun.org/bugs/wooyun-2010-0124352

- 怎么偷
  - Xss漏洞，去跨域请求此接口得到数据
  - 可能存在jsonp劫持漏洞

修复方案

不要使用get进行非读操作，不要使用get传输敏感信息！！！

至于怎么修复现有的漏洞，由于时间关系就不细讲了，有兴趣的可以以后和我交流探讨

谢　谢

欢迎白帽子加入小米安全部
欢迎白帽子来小米安全中心提交安全漏洞