Atitit Cookie安全法

目录

[1. cookie分为 会话cookie 和 持久cookie ， 1](#_Toc7701)

[1.1. 安全措施 1](#_Toc15756)

[1.2. 3. cookie的同源策略 2](#_Toc17414)

[1.3. 安全类库法提供cookie api 3](#_Toc22568)

[2. Php cookie设置和使用 3](#_Toc25215)

# cookie分为 会话cookie 和 持久cookie ，

会话cookie是指在不设定它的生命周期 **expires**时的状态，前面说了，浏览器的开启到关闭就是一次会话，当关闭浏览器时，会话cookie就会跟随浏览器而销毁。当关闭一个页面时，不影响会话cookie的销毁。会话cookie就像我们没有办理积分卡时，单一的买卖过程，离开之后，信息则销毁。

## 安全措施

保存id使用uname而不是uid，因为可以类推id

****对保存到cookie里面的敏感信息必须加密，或者签名防止篡改****

****四、给Cookie设置有效期****

1、如果不设置有效期，万一用户获取到用户的Cookie后，就可以一直使用用户身份登录。

2、在设置Cookie认证的时候，需要加入两个时间，一个是“即使一直在活动，也要失效”的时间，一个是“长时间不活动的失效时间”，并在Web应用中，首先判断两个时间是否已超时，再执行其他操作。

Salt法

## 3. cookie的同源策略

了解了Cookie的结构之后我们还需要知道关于Cookie的同源策略：

首先Cookie的同源策略是靠三元组name、domain、path来进行判断的，其中还有httponly和secure的限制。相对应的我们应该能想到Web的一个同源策略,它是靠scheme、domain、port来判断。

Web sop ,,cookie sop



如果设置了httponly

那么就无法通过js读写 [http://test.com](https://link.zhihu.com/?target=http://test.com" \t "https://zhuanlan.zhihu.com/p/_blank)的cookie

一般情况下在同一个一级域名下共享cookie时,它的domain一般写做:domain=”.test.com”;

## 安全类库法提供cookie api

# Php cookie设置和使用

 $\_GET['loginacc']

function setCookie4login()

{

    setcookie("logincookie", $\_GET['loginacc'], time() + 3600 \* 24, "/");

    setcookie("loginacc", $\_GET['loginacc'], time() + 3600 \* 24, "/");

    echo "ok";

}

浅谈cookie安全 - 知乎.html