基于Cookie的认证机制及其安全性分析

梁雪松
（四川教育学院物理系，四川 成都 610041）

【摘　要】Cookies是由Web服务器生成并存贮于用户计算机硬盘或内存中的文本信息，是实现Web应用认证的主要手段。分析了基于Cookie的认证机制的实现过程与特点，在此基础上指出了该认证机制易遭受的安全威胁，并提出了抵御这些威胁的安全需求。最后讨论了实现安全Cookie认证的具体方法与措施。

【关键词】Cookie；认证；访问令牌；安全套接层
【中图分类号】TP393.08　　【文献标识码】A　　【文章编号】1002-0802(2009)06-0132-03

Cookie-based Authentication Mechanism and Its Security Analysis

LIANG Xue-song
(Dept. of Physics, Sichuan College of Education, Sichuan Chengdu 610041, China)

【Abstract】Cookies are text messages generated by web servers and stored in user's hard driver or RAM, and are the primary means for web application authentication. In this paper, the implementation process and characteristics of cookie-based authentication mechanism are analyzed. The security threats to cookie-based authentication mechanism are pointed out, and the security requirements are proposed. Finally, the specific solutions for implementing secure cookie authentication are discussed.

【Key words】Cookie; authentication; access Token; Secure Socket Layer (SSL)

0 引言

HTTP协议是一个无状态的协议，Web服务器会将接收到的每个HTTP请求都视为相互独立、互不相干的访问请求。然而，维持状态对许多Web应用程序来说是非常重要的。购物车应用就是一个很好的例子，它需要在多个页面请求间关联一个购物车和某个用户。Cookie[1]技术最早于1994年由Netscape公司引入到Navigator浏览器中，是在客户端保持会话状态的实现方案，已成为被业界广泛采用的互联网标准。Cookie机制有效地解决了HTTP的状态管理问题，易于Web服务器实现网站访问统计、站点访问追踪、用户个性化空间定制等功能，同时也是在Web应用中实现用户认证的重要手段。

1 基于Cookie的认证机制概述

1.1 Cookie的组成

Cookie是一段不超过4KB的小型文本数据，由一个名称（Name）、一个值（Value）和其它几个用于控制Cookie有效期、安全性、使用范围的可选属性组成。其中：

(1) Name/Value：设置Cookie的名称及相应的值，对于认证Cookie，Value值包括Web服务器所提供的访问令牌。

(2)Expires属性：设置Cookie的生存期。有两种存储类型的Cookie：会话性与持久性。Expires属性缺省时，为会话性Cookie，仅保存在客户端内存中，并在用户关闭浏览器时失效；持久性Cookie会保存在用户的硬盘中，直至生存期到或用户直接在网页中单击“注销”等按钮结束会话时才会失效。

(3)Path属性：定义了Web站点上可以访问该Cookie的目录。

(4) Domain 属性：指定了可以访问该 Cookie 的 Web 站点或域。Cookie 机制并未遵循严格的同源策略，允许一个子域可以设置或获取其父域的 Cookie。当需要实现单点登录方案时，Cookie 的上述特性非常有用，然而也增加了 Cookie 受攻击的危险，比如攻击者可以借此发动会话定置攻击。因

收稿日期：2009-01-22。
作者简介：梁雪松（1972-），讲师，硕士，主要研究方向为计算机教学和计算机教育研究。

而，目前的浏览器禁止在 Domain 属性中设置.org、.com 等通用顶级域名、以及在国家及地区顶级域下注册的二级域名，比如.org.cn、.com.cn 等等，以减小攻击发生的范围。

(5)Secure属性：指定是否使用HTTPS安全协议发送Cookie。使用HTTPS安全协议，可以保护Cookie在浏览器和Web服务器间的传输过程中不被窃取和篡改。该方法也可用于Web站点的身份鉴别，即在HTTPS的连接建立阶段，浏览器会检查Web网站的SSL证书的有效性。但是基于兼容性的原因（比如有些网站使用自签署的证书），在检测到SSL证书无效时，浏览器并不会立即终止用户的连接请求，而是显示安全风险信息，用户仍可以选择继续访问该站点。由于许多用户缺乏安全意识，因而仍可能连接到Pharming攻击所伪造的网站。

(6)HTTPOnly 属性[2]：用于防止客户端脚本通过document.cookie属性访问Cookie，有助于保护Cookie不被跨站脚本攻击窃取或篡改。但是，HTTPOnly的应用仍存在局限性，一些浏览器可以阻止客户端脚本对Cookie的读操作，但允许写操作（如表1所示）；此外大多数浏览器仍允许通过XMLHTTP对象读取HTTP响应中的Set-Cookie头。

表1 浏览器支持HTTPOnly一览

浏览器名称	禁止读	禁止写
IE 7	√	√
Firefox 3.0	√	√
Opera 9.63	√	×
Safari 3.2	×	×
Chrome 1.0	√	×

1.2 Cookie认证机制的实现过程

基于Cookie的认证过程，主要由以下三个阶段组成：

（1）发布Cookie。

当用户试图访问某Web站点中需要认证的资源时，Web服务器会检查用户是否提供了认证Cookie，如果没有，则将用户重定向到登录页面。在用户成功登录后，Web服务器会产生认证Cookie，并通过HTTP响应中的Set-Cookie头发送给客户端，用于对用户随后的请求进行检查和验证，接着将用户重定向到初始请求的资源。

（2）检索Cookie。

在用户随后的访问请求中，客户端浏览器检索Path和Domain等属性与用户请求资源相匹配的Cookie，并将找到的Cookie通过HTTP请求中的Cookie头提交给Web服务器。

（3）验证Cookie

Web服务器提取客户端浏览器递交的Cookie，验证其中的访问令牌。若合法，则将访问请求的资源发送给客户端浏览器；反之则拒绝用户的访问请求。

Cookie 认证技术简化了用户访问 Web 网站资源的过程，即用户只需在初次登录网站时输入身份信息进行认证，随后便可以访问被授权的所有站点资源，不再需要重复手工提交身份信息。

2 Cookie认证的安全威胁

2.1 Cookie捕获/重放

攻击者可以通过木马等恶意程序，或使用跨站脚本攻击等手段偷窃存放在用户硬盘或内存中的Cookie。借助网络攻击手段，包括在不安全的局域网中被动地监听网络通信；通过攻击网络用户的路由器，或通过搭建恶意的无线路由器等手法，控制路由基础设施，将网络流量重定向到攻击者控制的主机；发动DNS Pharming[3](域欺骗)攻击，通过DNS缓存中毒、DNS应答欺骗、或修改用户端的本地域名解析文件等方法攻击DNS系统，导致用户对合法网站的访问请求被重定向到恶意网站等等，同样可能窃取Cookie。

对于捕获到的认证Cookie，攻击者往往会猜测其中的访问令牌，试图获取会话ID、用户名与口令、用户角色、时间戳等敏感信息；或者直接重放该Cookie，假冒受害者的身份发动攻击。

2.2 会话定置

会话定置（Session Fixation)攻击[4]是指，攻击者向受害者主机注入自己控制的认证Cookie等信息，使得受害者以攻击者的身份登录网站，从而窃取受害者的会话信息。注入Cookie的方法包括：使用跨站脚本或木马等恶意程序；或伪造与合法网站同域的站点，并利用各种方法欺骗用户访问该仿冒网站，从而通过HTTP响应中的Set-Cookie头将攻击者拥有的该域Cookie发送给用户等。

2.3 CSRF攻击

跨站请求伪造（Cross-Site Request Forgery，简称CSRF）[5]是指，攻击者可能利用网页中的恶意代码强迫受害者浏览器向被攻击的Web站点发送伪造的请求，篡夺受害者的认证Cookie等身份信息，从而假冒受害者对目标站点执行指定的操作。

Firefox、Opera等浏览器使用单进程机制，多个窗口或标签使用同一个进程，共享Cookie等会话数据。IE则混合使用单进程与多进程模式，一个窗口中的多个标签，以及使用“CTRL+N”或单击网页中的链接打开的新窗口使用同一进程，共享会话数据；只有直接运行IE可执行程序打开窗口时，才会创建新的进程。Chrome虽然使用多进程机制，然而经测试发现，其不同的窗口或标签之间仍会共享会话数据，除非使用隐身访问方式。因而，用户同时打开多个浏览器窗口或标签访问互联网资源时，就为CSRF攻击篡夺用户的会话Cookie创造了条件。另外，如果一个Web站点提供持久化Cookie，则CSRF攻击将更直接、更容易。

值得注意的是通过实验发现，IE、Chrome 、Safari等浏览器提供的第三方Cookie阻止工具并不完善，只能用于禁止浏览器接收第三方Cookie，因而无助于防止CSRF攻击。

3 Cookie认证的安全需求

从以上分析可以看出， Cookie认证的安全隐患存在于多个方面。抵御这些安全威胁，必须实现以下的安全需求。

3.1 保密性

利用密码技术对Cookie中的信息进行加密处理，以防止信息泄露和保护信息不为非授权用户掌握。

3.2 完整性

保证Cookie中的信息在存储或传输中不备非法插入、删除或修改，保证信息的完整性与真实性。

3.3 可鉴别性

确认Cookie是否来自其授权用户，防止对Cookie的重放或重置。

3.4 防篡夺性

防止利用CSRF攻击篡夺用户的认证Cookie。

4 实现安全Cookie认证的措施

4.1 加强用户安全意识

互联网用户应安装并及时升级反病毒软件，防止通过木马等恶意程序非法读取或篡改存放在用户硬盘或内存中的Cookie。在多人共享的公共计算机上，部署Windows 2000/2003等安全的操作系统，将保存持久性Cookie的文件夹放置在NTFS盘中，并设置NTFS访问权限防止其他用户非法访问。对于网上银行、在线购物等涉及机密信息网站，应尽量避免使用持久性Cookie。为防止CSRF攻击，在访问网上银行等敏感网站时应做到：（1）尽量避免通过同时打开多个浏览器窗口或标签来访问其他的网站；（2）使用一些浏览器提供的隐身访问功能。

4.2 基于用户控制的Cookie安全方案[6][7]

此方案由客户端使用加密算法、MAC（消息认证码）算法、时间戳等技术参与Cookie的保护工作，能有效防止Cookie重放与会话定置。然而，本方案需要修改现有的浏览器或开发相应的浏览器插件；对于非对称密钥解决方案，还需在客户端使用复杂的公钥基础设施（PKI）来创建和管理证书。

4.3 基于服务器控制的Cookie安全方案

此方案由服务器方决定采用何种安全措施保护Cookie，对用户透明，无需改动已有Web浏览器，但是此类方案较易受重放与会话定置攻击。

本文提出了一种服务器控制下的Cookie 安全方案。方案中使用对称加密算法实现Cookie的机密性；使用MAC算法保证Cookie内容的完整性；通过将Cookie与客户端的IP和用户代理头信息捆绑实现Cookie的可鉴别性，有助于防止Cookie重放与会话定置攻击；使用时间窗口进一步降低发生重放与会话定置攻击的概率；在表单中使用隐藏的CSRF令牌，防止利用CSRF攻击篡夺Cookie。此方案面临的主要问题是不利于动态IP用户保存持久性Cookie，在这一方面需要作进一步研究。

4.3.1 Cookie的生成算法

在用户成功登录站点后，Web服务器会将产生认证Cookie通过HTTP响应中的Set-Cookie头发送给客户端，Cookie中使用了Secure和HTTPOnly属性，其Value值的组成如下：

sid||expired time||E_k (data) || HMAC(sid||expired time||data||ip||user agent，k)

其中：

sid：Web站点分配给用户的唯一访问标识符。

expired time:Cookie的过期时间，对于持久性Cookie，其值与Expires属性相同。

E_k (data) ：data可包含用户名、用户角色等令牌信息，并由密钥k加密。

k：由HMAC（sid，sk）生成，其中sk为Web站点具有的唯一密钥，HMAC为HMAC-MD5或HMAC-SHA1等带密钥的MAC算法。

ip：客户端的IP地址。

user agent：HTTP请求中的用户代理头(User- Agent)信息。将用户代理头与Cookie捆绑，可防止IP欺骗，有助于区分使用NAT或代理服务器的网络中的不同用户，使得Cookie重放或会话定置攻击更困难。

4.3.2 Cookie的认证算法

Web服务器从用户的Http请求中获取认证Cookie后，按如下的步骤进行验证：

（1）比较expired time与服务器当前的时间，如果过期，则认证失败；

（2）利用HMAC（sid，sk）计算出解密密钥k；

（3）使用解密密钥k解密出data；

（4）获取用户端的IP地址以及HTTP请求中的用户代理头信息；

（5）计算并验证消息验证码HMAC(sid||expired time||data||ip||user agent，k)，如果验证失败，则拒绝访问请求。

4.3.3 CSRF检测算法

在涉及用户敏感操作的表单中增加一个隐藏的CSRF令牌域，其值为HMAC（sid ||action name，k），其中“action name”表示接收此表单的Web页的名称。在提交表单时，Web服务器重新计算并验证表单中的CSRF令牌，如果验证失败，该请求就被认为是个CSRF尝试并被拒绝。

5 结语

Cookie是实现Web认证服务的主要手段。本文讨论了Cookie认证机制易遭受的安全威胁，分析了多种安全措施与方法，并给出了一种服务器控制下Cookie安全方案，该机制能确保Cookie的机密性、完整性、可鉴别性以及防篡夺性，具有一定的适应性。

（下转第137页）

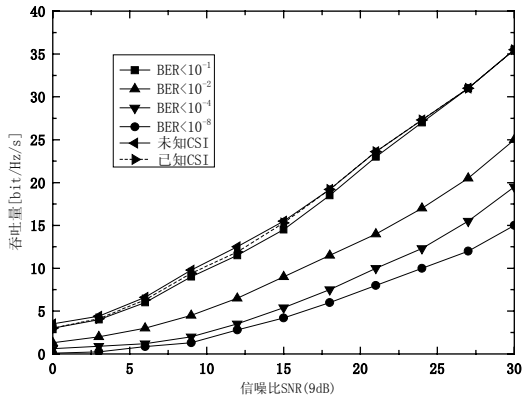

图 3 MIMO 系统的吞吐量随信噪比的变化

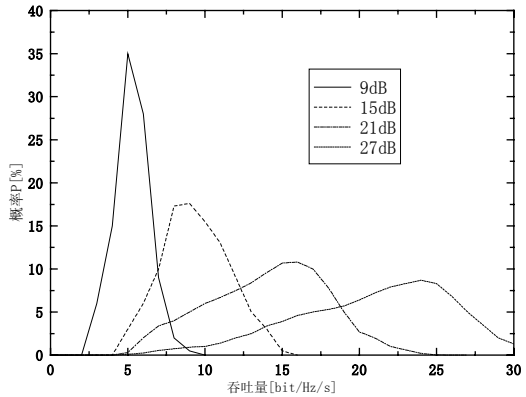

图 4 MIMO 系统的吞吐量分布概率

图 3 表示系统平均吞吐量和平均信噪比 SNR 的依赖关系。实线表示系统理论吞吐量；而点化线表示仅接收端已知信道状态信息的情况下仿真的结果。从图可以看出，在 SNR 较低时，要满足目标 BER 的要求，能通过减少系统传输的信息量取得。具体通过关断一些信道条件恶劣的发射天线，仅让相关性弱的天线发射信息，而不是让所有的天线都传输信息和应用多路复用技术实现；在接收机端，为了提高系统的鲁棒性，应用空间分级技术，接收并处理四个失真的信号，提高接收的可靠性。在 SNR 高的情况下，选择具有带宽效率高的调制方式进行传输，并把数据流分配到所有的天线上，这样系统的吞吐量就增加了。

图 4 给出了在目标误码率 BER 为 10^{-2}，平均信噪比 SNR 分别为 9dB、15dB、21dB、27dB 时的系统吞吐量的概率密度分布。观察发现，SNR 低时系统吞吐量的标准偏差较小；随着 SNR 的增大，系统吞吐量逐渐增大，但标准偏差变大，结果接收的信号包络服从瑞利分布。

4 结语

本文提出了一种用于 MIMO 系统的新的链路自适应算法。算法通过使用矩阵求逆检测技术，具有极低的计算量，解决了使用矩阵奇异值分解 SVD 方法所存在的主要缺点，显著地减少了控制信息的传输开销，大大提高了系统的吞吐量，传输的可靠性也具有显著改善，特别适用于恶劣信道环境的应用。

参考文献

[1] David G, Robert W H Jr. Adaptive Modulation and MIMO Coding for Broadband Wireless Data Networks[J]. IEEE Communications Magazine, June 2002:108-115.

[2] Hwang K J,Lee S K,Chang K H. Adaptive rate MIMO system using space-time block mapping[J]. VTC03-Spring, 2003;1(4):774-778.

[3] Lan Z, Dubey V K. Transmit diversity and combining scheme for spatial multiplexing over correlated channels[J]. VTC-04-Spring,2004;1(3):380-383.

[4] CHUNG S T, COLDSMITH A J. Degrees of freedom in adaptive modulation: A unified view[J]. IEEE Trans on Communications, 2001, 49(9):1561-1571.

[5] ALOUINI M S, COLDSMITH A J. Adaptive modulation over Nakagami fading channels [J]. Kluwer Journal on Wireless Communications, 2000, 13(1-2):119-143.

[6] CATREUX S,ERCEG V, GESBERT D, et al. Adaptive modulation and MIMO coding for broadband wireless data networks [J]. IEEE Communications Magazine, 2002, 40(6):108-11.

（上接第 134 页）

参考文献

[1] Kristol D,Montulli L. RFC 2109,HTTP State Management Mechanism [S]. IETF, Feb 1997.

[2] Microsoft Corporation. Mitigating Cross-site Scripting With HTTP-only Cookies[EB/OL]. http://msdn.microsoft.com/en-us/library/ ms533046. aspx.

[3] NGSSoftware Corporation. the Pharming Guide[EB/OL]. http://www.ngssoftware.com/ papers/ ThePharmingGuide. pdf, July 2005.

[4] Mitja Kolšek. Session Fixation Vulnerability in Web-based Applications[EB/OL].http://www.acros.si/papers/session_fixation.pdf, Dec 2007.

[5] Adam Barth, Collin Jackso, John C. Mitchell. Robust Defenses for Cross-Site Request Forgery[C]. CCS' 08, 2008, Alexandria Virginia USA:ACM Press , 2008:75-88.

[6] 吴建武.基于公钥证书的cookies 安全实现方案[J].微计算机信息，2006，22(7-3)：136-172.

[7] 李景峰，祝跃飞，张栋.用户控制下 Cookies 安全研究与实现[J].计算机工程，2005， 31(14):150-152.