

一种安全的单点登录系统口令同步方案

张秋余, 蔡志鹏, 袁占亭

(兰州理工大学计算机与通信学院, 兰州 730050)

摘要: 针对现有单点登录(SSO)系统难以有效支持口令同步的问题, 设计一个安全性和扩展性更好的 SSO 系统模型, 利用最优非对称加密填充算法修改明文, 采用 RSA 加密算法产生数字证书, 在此基础上改进混合密码传输协议, 以更好地实现 SSO 系统中服务器、客户端代理服务器及认证机构三者之间的传输应用。通过对口令同步的实现, 验证该方案相比原协议即时性更强、口令密钥安全性更高, 能有效抵抗选择密文攻击。

关键词: 单点登录系统; 口令同步; 混合密码传输协议; 最优非对称加密填充算法; RSA 算法

Secure Password Synchronization Scheme for Single Sign-on System

ZHANG Qiu-yu, CAI Zhi-peng, YUAN Zhan-ting

(School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China)

[Abstract] In order to make the current Single Sign-on(SSO) schemes effectively support the application of password synchronization, this paper designs a secure model with better expansionary. It uses Optimal Asymmetric Encryption Padding(OAEP) encryption algorithm to modify the plaintext and creates a digital certificate through RSA encryption algorithm. Hybrid Cryptograph Transfer Protocol(HCTP) is improved to make it better realize the transmission applications among the server, the client proxy server and the Certificate Authority(CA) institutions of transfer in SSO system. The realization of password synchronization test verifies the stronger instantaneity and the higher security in the password key compared with HCTP, which can effectively resist chosen-ciphertext attack.

[Key words] Single Sign-on(SSO) system; password synchronization; Hybrid Cryptograph Transfer Protocol(HCTP); Optimal Asymmetric Encryption Padding(OAEP) algorithm; RSA algorithm

DOI: 10.3969/j.issn.1000-3428.2011.17.040

1 概述

单点登录(Single Sign-on, SSO)^[1]技术的产生使用户的工作效率更高、网络安全性更强、系统的管理效率更高。目前已有许多单点登录系统, 如 Liberty Authentication SSO 系统和 Net Passport 系统。但是在现阶段的单点登录产品中, 如 CA 公司的 eTrustSSO、Novell 公司的 Securelogin SSO 系统, 没有完全解决口令同步问题, 口令同步是单点登录技术中的重点和难点。文献[2]提出了一种 B/S 架构的单点登录模型并讨论了部分口令同步技术的解决方案, 但是还未实现在 FTP、Telnet 的应用中进行口令同步。文献[3]提出了多模式应用的单点登录方案, 但没有对口令同步功能的实现展开讨论。文献[4]提出了一种用于口令同步的通用混合密码传输协议, 可应用于任何 2 个能够提供数字证书的服务器之间交换共享密钥或 Web 认证中不同服务器之间的认证和数据交换, 但口令由于使用 RSA 算法加密而容易受选择密文攻击, 且文中没有讨论口令同步在单点登录环境中如何实现。

借鉴上述成果, 本文提出了一种能够在安全的单点登录系统中实现口令同步的方案。

2 单点登录模型

2.1 SSO 模型设计思想

为了满足单点登录系统的应用要求, 并实现更高的安全性和更强的扩展性, 同时考虑在多模式下实现口令同步, 本文对文献[2-3]的 SSO 系统模型进行了融合与改进, 设计了如图 1 所示的 SSO 模型, 其中, 虚线框是对不同功能模块的划分。

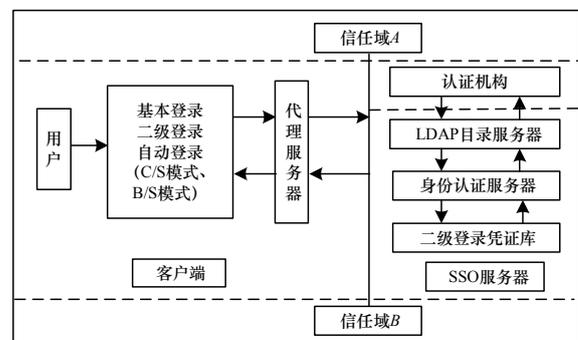


图 1 单点登录模型

2.2 单点登录的实现

在客户端方面, 本文考虑常用的 B/S 和 C/S 2 种模式。在认证方面, 对 B/S 模式来说, 用户访问 Web 页面时可以通过控件处理客户端事件, 而对 C/S 模式来说, 可以利用提供组件对象模型接口来实现。2 种模式在单点登录原理上是一致的。本文以 B/S 为例, 分 2 种情况讨论 SSO 服务的实现过程: (1)对于未注册过的用户, 其流程如下: 用户首先需要在系统中进行注册, 通过身份认证服务器验证用户身份的有效性。在授权策略范围内, 设定用户名和口令, 建立该用户的

基金项目: 甘肃省自然科学基金资助项目(0803RJZA024)

作者简介: 张秋余(1966—), 男, 研究员、CCF 高级会员, 主研方向: 信息隐藏与隐写分析, 网络与信息安全; 蔡志鹏, 硕士研究生; 袁占亭, 教授、博士生导师

收稿日期: 2011-01-25 **E-mail:** zhangqylz@sina.com

初始凭证库信息,并在 LDAP 目录服务器中登记该用户项,这样就注册成功。在口令同步功能上,用户首次注册和修改口令的过程原理是一样的,都得把新生成的参数分配到用户登录处,也就是经过改进后的混合密码传输协议,将用户名和口令的特征参数传输给客户端的登录处,达到口令同步效果。(2)对于已注册过的用户,其流程如下:用户通过浏览器向某信任域发出访问资源请求。客户端应用程序自动分析检测该网页中的输入框是否有用户登录过用户名和口令的记录。若发现用户没有登录,将用户访问重定向到单点登录服务器的用户认证服务,生成被访问站点的用户登录界面,对用户进行认证。认证成功后,用户和站点之间通过会话密钥建立会话,进行通信,实现自动登录。若用户输入了用户名和口令,则马上弹出对话框,提示用户是否保存该网页的登录凭证,当授权应用的登录凭证已经建立好后,可以通过客户端应用程序自动完成授权应用的登录。当用户在客户端应用程序主窗口中双击其图标后,自动启动 IE 实例下载该网页,将用户名和口令自动填入网页中,并自动完成提交,实现透明登录。

通过上述 SSO,用户只需使用一次用户名/口令就能访问多域信任资源,这样既简化了用户的操作,又降低了口令泄漏的危险,所以,该 SSO 系统可以有效地防御中间人攻击和重放攻击,并且改进的模型具有良好的扩展性,可与不同的授权系统和认证代理结合,且支持跨域认证。

3 基于最优非对称加密填充算法的口令加密

最优非对称加密填充(Optimal Asymmetric Encryption Padding, OAEP)算法^[5]可以对明文进行修改,从而提高口令的安全性。算法具体步骤如下:

(1)待加密的消息 M 被填充: 1)客户修改口令时,设新口令长度为 n ,消息为 M , K_0 为可选参数集 P 作为散列函数 H 的输入长度; 2)输出用 K_1 个 0 加以填充以获得期望的长度放入整体数据块内; 3)获得消息 M 期望长度 $K=n+K_0+K_1$ 。

(2)随机选择一个种子 $seed$ 作为掩码生成函数 MGF 的输入,然后输出散列值与数据块进行按位异或运算,产生掩码数据块 $maskedDB$ 。

(3)该 $maskedDB$ 反过来作为 MGF 的输入产生一个散列值,与种子进行异或运算,得到掩码种子 $maskedseed$ 。

(4)该 $maskedseed$ 和 $maskedDB$ 连接起来构成加密后的消息 EM 。

在以往的 SSO 系统中,攻击者在用户修改口令过程中实施 CCA 攻击是利用了 RSA 的以下性质:

$$E(PU, M1) \times E(PU, M2) = E(PU, M1 \times M2)$$

攻击者首先选择一些明文,运用公钥加密,然后通过私钥解密获得明文。本文采用此算法使密文随机化,由此,攻击者无法利用 RSA 的性质发起安全威胁,从而提高了新口令的安全性。

4 基于 RSA 算法的 EM 签发数字证书加密

RSA 算法随机选择 2 个大素数 p 、 q ,且 p 不等于 q ,然后计算 $n: n=p \times q$; 随机选择加密密钥 e ,其满足 $1 < e < \phi(n)$, $\text{gcd}(e, \phi(n))=1$,并计算 $d: d=e^{-1}, 0 < d \leq n$ 。

假定用户 B 已经公布了其公钥 $PU=\{e, n\}$,用户 A 要发送消息 EM 给 B 。首先用户 A 加密计算 $C=(EM)^e \bmod n$,并发送密文 C ;然后在接收端上,用户 B 解密计算 $EM=C^d \bmod n$,得到消息 EM 。

本文选用 RSA 算法进行数字签名产生数字证书。用户向

CA 管理员发送请求验证身份并附带 EM 消息,CA 接到消息后,产生一对公钥/私钥,转化为证书形式,并对此进行签名,得到用户证书: $CA=E(PR_A, PU_B, ID_A, PU_A, t_A, S_A)$,其中, PR_A 是私钥; ID_A 是包含 A 的标识; PU_A 为 A 的公开密钥; t_A 为时间有效期; S_A 为数字签名: $S_A = \text{sign}(PR_A, PU_B, ID_A, PU_A, t_A) = [H(PR_A, PU_B, ID_A, PU_A, t_A)]_R^d \bmod n_R$ 。最后,CA 管理员将用户身份信息和证书保存到 LDAP 目录服务器中。

为了能够实时、简洁、快速地实现同步,用户 A 和 B 采用 Diffie-Hellman 密钥交换算法^[6]产生主密钥,步骤如下:

(1)用户 A 的密钥产生: 1)选择秘密的 $X_A, X_A < q$ 。2)计算公开的 $Y_A: Y_A = \alpha^{X_A} \bmod q$ 。

(2)用户 B 的密钥产生: 1)选择秘密的 $X_B, X_B < q$ 。2)计算公开的 $Y_B: Y_B = \alpha^{X_B} \bmod q$ 。

5 SSO 系统中口令同步的实现

结合上面所建立的实际单点登录模型、加密算法和数字证书,对原混合密码传输协议进行改进,使它适合当前 SSO 系统中的服务器 B 、客户端代理服务器 A 以及 CA 机构三者之间的传输应用。新协议流程如下:

(1)客户端选定新的口令后,通过代理服务器 A ,发送一条带有时间戳 t_A 的消息给 CA ,以请求获取数字证书和与会话 SSO 服务器通信的公钥 PU_B 、会话密钥 K ;同时,与 A 检索 B 的公钥一样, B 以同样的方法向 CA 申请 PU_A 。

(2)CA 管理员根据接收到的请求和消息,通过数字签名产生用户证书,同时发送加密的消息给 A 与 B ,这样, A 与 B 分别收到以下形式的证书: $E(PR_A, PU_B, ID_A, PU_A, t_A, S_A)$, $E(PR_B, PU_B, ID_B, PU_A, t_B, S_B)$ 。取出证书中的各元素,验证签名是否有效,只有验证成功、 A 得到会话主密钥后, A 与 B 才能进行会话。

(3) A 用 B 的公钥 PU_A 对含有其标识 ID_A 临时交互号 N_1 的消息加密,并发送给 B 。其中, N_1 用来唯一标识本次交易。

(4)当 B 收到 CA 后,检验密钥参数,核对 ID_A 是否相符、 t_A 是否有效,然后验证 S_A 的签名。若相符,把密钥存放在二级凭证库里,并且发送一条加密消息给 A ,该消息包含 A 的临时交互号 N_1 和 B 产生的新临时交互号 N_2 ,用会话密钥加密命令字 "get"。

(5) A 收到消息后,解密得到各参数,然后 A 用 B 的公钥对 N_2 加密,并返回给 B ,从而使 B 确信与其通信的是 A ,而且会话完毕。这样,客户端 A 新的口令与 SSO 服务器保持一致性和同步性,口令同步得以实现。

上述过程中涉及的证书交换如图 2 所示。

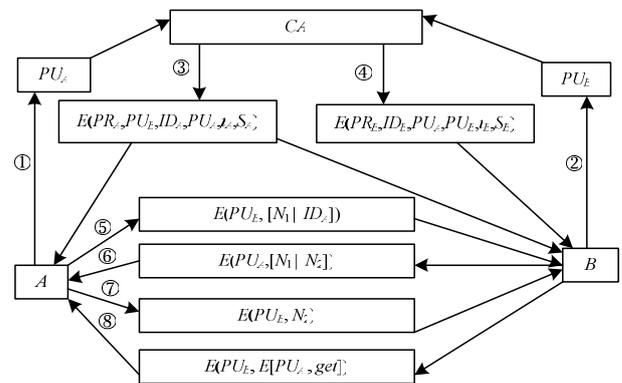


图 2 公钥证书的交换过程