一种匿名口令鉴别构件系统①

周 楠[1,2], 张立武[2]

[1](中国科学院大学, 北京 100190)
[2](中国科学院软件研究所, 北京 100190)

摘 要: 作为隐私保护的重要手段, 匿名鉴别机制引起了各界的广泛关注, 口令鉴别作为应用最广泛的鉴别方式, 研究设计仅依赖于口令的匿名鉴别系统具有重要意义. 在此背景下, 国际标准化组织 ISO/IEC JTC1/SC27 启动了 ISO/IEC20009-4 标准项目, 专门针对基于口令的匿名鉴别机制开展标准化工作, 目前收录了三种匿名口令鉴别协议. 本文基于该标准中的 SKI 协议研究设计了一种匿名口令鉴别构件系统, 同时系统可支持标准中收录的其它两种协议. 本文针对该构件系统的安全性、匿名性以及性能方面进行了系统的分析设计, 从系统初始化、匿名分组构建到标准协议执行等各环节进行了安全保护, 填补了标准本身到实际应用的差距. 本文主要解决了 SKI 协议应用时面临的安全及效率问题, 包括: 协议中“公告板信息不一致”将导致合法用户认证失败、群组公告信息的“首次请求”面临超长等待延时等问题. 构件系统通过引入“双重公告信息”、“请求绑定会话”、“Cache 机制”等多种手段解决了上述问题. 最后, 我们对匿名口令鉴别构件系统的性能进行了实验分析. 目前尚未见国内外同类系统被提出.

关键词: 匿名鉴别; 口令鉴别; 隐私保护; 构件; 国际标准

A Kind of Anonymous Password Authentication Component System

ZHOU Nan[1,2], ZHANG Li-Wu[2]

[1](University of Chinese Academy of Sciences, Beijing 100090, China)
[2](Institute of Software, Chinese Academy of Sciences, Beijing 100090, China)

Abstract: Anonymous authentication has attracted widespread attention of the public as an important means of privacy protection. It is significant to achieve anonymous mechanism based on password which is the most common method of user authentication and remains very widely used in cyberspace. In this scenario, ISO/IEC JTC1/SC27 launched the project of ISO/IEC20009-4 to prepare the standards for PAEA(password-based anonymous entity authentication) mechanisms. There are 3 kinds of protocols for PAEA specified in the ISO/IEC2009-4. This paper designs and specifies a kind of anonymous password entity-authentication component system based on the SKI mechanism, which is one of three kinds of protocols for PAEA in ISO/IEC20009-4. And the system can support the other two protocols at the same time. We analyze the security, anonymity and performance of the system and enhance the security of the processes of system initialization, construction of anonymous group and execution of the protocol. Our work has closed the gap between the theory and the application of the mechanisms. In this paper, we mainly solve the security and performance problems of SKI mechanism when SKI is applied in practice, including “Inconformity of Bulletin Information” which will lead to failure in authentication and the long latency of the “First Query” of a group’s public bulletin information etc. And we solve all of them by introducing the solutions of “Dual Bulletin Information Scheme”, “Cache System”, “Query-Bound-Session Mechanism”, etc. At last, we analyzethe performance of the system by do the testing experiences. For now, no domestic and foreign similar systems have been proposed.

Key words: anonymous authentication; password authentication; privacy protection; component; international standard

① 基金项目: 国家自然科学基金(61472409,61303247);国家自然科学基金重点项目(91118006);国家高技术研究发展计划(863)(2012AA01A403);国家重点基础研究计划(973)(2013CB338003)

收稿时间:2016-03-07;收到修改稿时间:2016-04-24 [doi:10.15888/j.cnki.csa.005442]

1 引言

匿名鉴别在验证用户身份合法性的同时对鉴别方隐藏了用户的具体身份信息, 从而防止服务提供商将用户行为与特定的用户进行关联, 保护了用户的隐私.

在众多的鉴别方法中, 口令鉴别因其无需任何额外的设备即可完成鉴别过程, 简单方便的特性使得其成为了当下应用最广泛的鉴别形式. 因此为口令鉴别提供匿名机制有了重大的实际意义. 在此形势下, 国际标准化组织 ISO/IEC JTC1/SC27 启动了 ISO/IEC20009-4《信息技术 安全技术 匿名实体鉴别: 基于弱秘密的机制》标准[1], 负责开展针对弱秘密(通常指口令)标准化工作, 其收录的三种匿名口令鉴别协议分别为: SKI[2]、YZ[3]、YZW[4]. 该标准化进程现已进入 DIS(Draft International Standard)阶段.

口令鉴别在提供便利性的同时, 其低熵值弊端为攻击者通过离线字典攻击、暴力破解等提供了方便. 基于口令实现匿名实体鉴别面临双重挑战, 既要鉴别时不能泄露具体身份, 又要面临弱秘密的离线字典攻击等问题, 无法直接采用面向强秘密设计的匿名实体鉴别机制, 因此利用弱秘密完成匿名实体鉴别, 必须进行精心设计.

就目前而言, 对于标准[1]收录的各个鉴别机制的实现尚且没有公开的工作. 协议标准侧重于理论, 在进行实际应用中需要仍然面临着诸多问题. 如性能优化, 协议适用性以及协议与上层应用间的消息交互方式等问题都有待解决. 更重要的是我们发现协议在实际应用时由于现实环境的复杂性而可能导致协议认证失败、客户端等待时间过长等问题. 例如, 在现实中, 同一次鉴别过程的服务器和客户端需要异步向公告板发出请求, 两个请求到达时间若跨越了公告板的更新时间则会产生“公告板信息不一致”的问题, 该问题最终将导致合法用户认证失败.

针对以上问题, 我们设计和实现了一种匿名口令鉴别构件系统, 该系统以构件形式为互联网应用提供使用纯口令的匿名鉴别服务. 系统基于 SKI 协议实现, 而作为国际标准, SKI 协议仅仅具有理论上的完备性, 其在实际应用中, 仍然面临诸多问题. 我们发现并解决了诸如“公告板信息不一致”、“首次请求延时长”等协议在实际应用时存在的问题. 鉴于协议的复杂性, 其在执行时需要较大的资源开销, 我们从客户端和服务端分别提出了多种优化手段提升系统性能改善用户体验. 通过对系统进行分层式设计并引入“消息适配器”实现了剥离通信层的架构, 使得系统能够支持多种通信模式以及兼容更多协议. 另外, 我们将 SKI 协议特点与现实中的应用需求相结合提出了“多级化匿名身份管理体系”, 丰富了匿名口令鉴别的应用场景. 最后, 我们设计实验测试和评价了该系统在性能以及用户体验等方面的表现. 结果显示, 不仅该系统所需的开销较小, 并且其鉴别过程的低延时也不会对用户造成困扰, 应用性强, 具有较高的实用价值.

1.1 相关工作

匿名凭证系统最早由 Chaum[5]于 1985 年提出, 并于 1987 年给出了一个具体的实现方案[6], 但其可行性较差, 之后匿名凭证系统发展较为缓慢, 仅有一些原型方案被提出. 随着知识证明技术与群签名、环签名等隐私保护技术研究的深入, 一些基于 PKI 系统、可信平台模块(TPM)的匿名鉴别方案陆续被提出. 其中包括 1999 年, Lysyanskaya 提出一个假名系统的实现方案[7]. 2000 年 Stefan Brands 提出的证书颁发出示方案[8], 该方案基于 PKI 证书, 同时可实现属性的可选择泄露与证书验证的不可关联性. 2001 年 Jan Camenisch 提出一套基于强 RSA 困难假设的匿名凭证的颁发、展示协议, 并且在 2002 年给出了原型实现——idemix 系统[9]. Idemix 能够实现用户在不公开凭证的情况下证明自己拥有该凭证, 此外用户还可以证明凭证中包含的属性的简单性质即属性的选择性泄露. 之后, Jan Camenisch 由这套协议构造了 CL 群签名方案[10], 并给出了一个用于可信计算领域 TPM 认证的 idemix 的轻量级实现. 2004 年, Jan Camenisch 又给出了基于双线性映射的匿名凭证系统的实现方案[11]. 2008 年, Tsang 等人针对黑名单检测问题, 基于强 RSA 假设的 CL 群签名方案实现了一个匿名凭证系统 PEREA[12]. 该系统利用用户登录票据为素数的特点, 通过使用户维护一个自己的登录票据列表, 实现了服务器进行用户黑名单检测的计算量与名单大小无关的目标.

以上方案均对计算开销、硬件条件、基础设施支撑等方面提出了较高的要求. 对于普通互联网用户而言, 其往往仅持有 PC 或移动设备作为其硬件平台且注重用户体验. 而基于纯口令的匿名身份鉴别机制一方面不对用户的硬件做额外要求, 另一方面口令作为鉴别手段在使用便利性方面有着天然的优势. 因此基

于口令的匿名鉴别系统具有较高的应用价值.

本文的后续章节安排如下: 第 2 节介绍本文相关的理论知识, 包括标准协议 SKI 及其原型协议 VEAP 的说明; 第 3 节阐述匿名口令鉴别构件系统的研究与设计工作, 描述系统的架构组成; 第 4 节结合实际应用情况分析了系统安全性和性能, 并提出相关方案对系统在应用时产生的问题进行了修补; 第 5 节给出设计实验测试了系统性能, 并评估了系统可用性. 第 6 节总结了本文的工作.

2 预备知识

2.1 VEAP 协议

2010 年 SeongHan Shin[2]等人提出 VEAP 协议, ISO/IEC 20009-4 收录的 SKI 机制即是在此协议的基础上发展而来的.

协议工作在阶为素数 q 的有限循环群 G =< g > 上,g 为群的生成元. 用户 U_i 与服务器 S 之间预先共享口令 pw_i. 假设在协议运行前, 通信双方就用户群 $U=\{U_1,\ldots,U_n\}$ 已达成共识. 协议包括协议准备与交互鉴别两个阶段. 其中 $g:\{0,1\}^*\to G$ 为全值域散列函数 H_1、H_2、H_3 为三个随机散列函数.

1) 协议准备. 在该阶段服务器对用户群 U 的每个用户 U_j 使用其与服务端共享的密钥的散列结果加密服务器随机主密钥 MS 生成密文 C_j.

步骤 0: 服务器 S 选择随机数 $x\in Z_p^*$ 以及一个随机主密钥 $MS\in\{0,1\}^l$, 并计算 Diffie-Hellman 公共值 $X\equiv g^x$. 然后, 服务端为每个用户 $U_{j(1\le j\le n)}$ 计算 $W_i\leftarrow g(U_i,pw_i)$ 并且产生对称密钥 $K_j\equiv(W_j)^x$, 然后产生使用 κ_j 加密 MS 的密文 $C_j=\varepsilon_{K_j}(i,pw_i)$.

2) 交互鉴别.

步骤 1: 用户 U_i 选择随机数 $a\in Z_p^*$ 并计算 $A\equiv W_i\times g^a$. 用户将 $<A,\ U>$ 发送给服务器 S.

步骤 2: 服务器收到 A 之后使用 x 计算 A^x.并产生认证标识符 $V_S\leftarrow H_1(U\|S\|TRANS\|MS)$, 其中 $TRANS=A\|A^x\|X\|\{C_j\}_{1\le j\le n}$. 然后服务器 S 发送 $<S,X,A^x,\{C_j\}_{1\le j\le n},V_S>$ 给客户端.

步骤 3: 用户 U_i 收到 S 发送的消息后, 计算 $K_i\equiv A^x/X^a$, $\kappa_j\leftarrow F(U_i,X,W_i,K_i)$. 并使用 κ_j 解密第 i 个 MS 的加密密文 $C_{i=j}$ 即 $MS'=D_{\kappa_j}(C_j)$. 并且根据获得的信息判断 V_S 是否合法, 如果 V_S 不合法, 用户终止协议. 否则计算鉴别标志 $V_{U_i}\leftarrow H_2(U\|S\|TRANS\|MS')$ 和会话密钥 $SK\leftarrow H_3(U\|S\|TRANS\|MS')$. 并将 V_{U_i} 发送给服务器 S.

步骤 4: 如果服务器 S 验证收到的 V_{U_i} 不合法则终止协议, 否则服务器生成会话密钥 $SK\leftarrow H_3(U\|S\|TRANS\|MS)$.

2.2 SKI 机制

国际标准化组织 ISO/IEC JTC1/SC27 启动了 ISO/IEC20009 项目负责开展匿名实体鉴别的标准化工作, 本文的匿名口令鉴别构件系统基于的 SKI 机制即是 ISO/IEC20009 第 4 部分[1]收录的 3 个基于弱秘密的机制之一. SKI 机制在 VEAP 的基础上, 对协议流程进行了更加详细的描述并通过修改协议步骤细节增强了协议的安全和隐私性.

在 SKI 机制中, 将 VEAP 交互鉴别阶段步骤 2 的数据 $\{C_j\}_{1\le j\le n}$ 的计算和获取提前到协议准备阶段. 在 SKI 机制的协议准备阶段, 服务器为群组 U 选取一个随机主密钥 MS, 使用用户 U_i 的口令 pw_i 散列结果加密 MS 生成密文 C_j, 并将 C_j 与用户的假名 $\underline{U_j}$ 以<$\underline{U_j}$, C_j>$_{1\le j\le n}$ 的形式公布. 其中假名 $\underline{U_j}$ 只能由掌握 pw_i 的用户和服务器在由服务器选取的公开参数作用下计算得出. 当用户 U_i 从服务端获取了群组全体成员的<$\underline{U_j}$, C_j>$_{1\le j\le n}$ 后, 根据计算得出的假名定位对应的 C_j 进而使用 pw_i 的参数解密 C_j 得出 MS'. 同时, SKI 为增强协议性能, 对该公布的群组信息<$\underline{U_j}$, C_j>$_{1\le j\le n}$ 设置一段有效时间 T, 从信息公布起的 T 时间内, 服务器不重新计算该群组信息, 降低了服务端的计算开销.

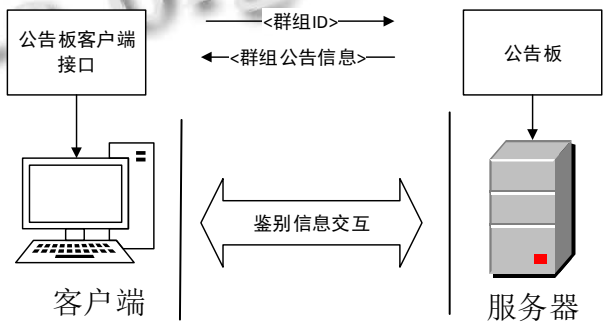

图 1 协议执行示意图

3 匿名口令鉴别构件系统设计

3.1 设计目标

匿名口令鉴别构件系统面向互联网应用旨在为其提供使用纯口令作为鉴别手段的匿名鉴别服务. 构件系统基于 ISO/IEC 20009-4 收录的标准协议[1]进行设计

和实现, 我们认为面向普通互联网用户的基于口令的匿名身份鉴别系统应当具备以下特性:

(1) 可用性. 作为匿名鉴别标准, SKI 等协议在进行应用时仍然面临着诸多问题. 我们的一个重要目标即是发现并解决这些问题, 填补协议标准理论与应用间的差距.

(2) 安全性和隐私性. 我们强调在系统的鉴别过程中, 除用户外的任何实体应不能够获取用户相关的身份信息.

(3) 用户体验尽量保持不变. 当下, 大多数的网站为其用户提供了使用浏览器的 Web 登录方式. 本文中的匿名口令鉴别构件系统应尽量维持用户的登录体验保持不变.

(4) 适用性强. 匿名口令鉴别上层应用可能应用在多种不同场景中, 其中所使用的通信模式多种多样. 因此在系统设计时需要考虑上层应用可能使用的通信方式.

(5) 低延迟. 在 SKI 机制中服务端和客户端将进行多次的大整数指数运算, 并进行多轮消息交互来完成匿名鉴别过程, 因此会产生较大的协议执行开销. 鉴别系统应当对系统性能进行优化以实现客户端登录过程低延迟的目标.

3.2 设计思想

本匿名口令鉴别构件系统实现 3.1 节中提出的 5 个目标. 可用性以及安全隐私性是我们首先要解决的问题. 结合现实应用的特点, 本文提出了“双重公告信息”、“Cache 机制”等多种方案对 SKI 协议在实现时遇到的问题进行了修补以增强系统安全强度、优化系统性能, 使其现实可用. 为了兼容多样性的上层通信模式, 我们对系统进行了分层设计, 将协议具体实现与上层应用接口隔离开来, 并引入独立的中间模块“消息适配器”定义两个构件间的互操作方式.

3.3 系统架构

匿名口令鉴别构件系统由公告板、协议包构件、消息适配器、上层应用接口四个相对独立的构件组成. 协议包构件负责匿名口令鉴别协议的实现. 消息适配器处于协议包构件和上层应用接口之间, 将从一方发出的消息转化成另一方可以接收的形式, 完成消息适配的功能. 公告板对应于 SKI 机制的协议准备阶段, 实现该阶段群组公告信息的发布和维护等功能. 上层应用接口面向网络应用, 为应用提供协议协商、通信支持、协议配置管理等功能. 系统架构图如图 2.

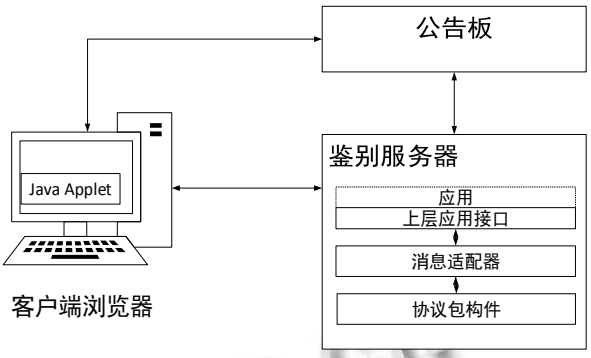

图 2　系统架构图

3.4 协议包构件

匿名口令鉴别协议包构件负责实现 SKI 协议交互鉴别阶段的鉴别逻辑处理以及定义协议包与外部的交互方式. 另外考虑到未来的扩展需求, 我们为容纳更多的鉴别协议预留了开发接口.

标准文献[1]收录的 3 种匿名口令鉴别协议的主要处理流程均是对接收消息的处理过程, 并且都会涉及到散列运算、随机数生成、大整数运算、编解码等操作. 另外, 对于服务端和客户端的交互而言, 各个协议都有其特有的消息格式. 基于以上分析, 我们将协议包构件分为消息子模块、鉴别逻辑处理子模块、公告板数据接口、基础操作子模块等子模块. 各个模块的详细功能如图 3.

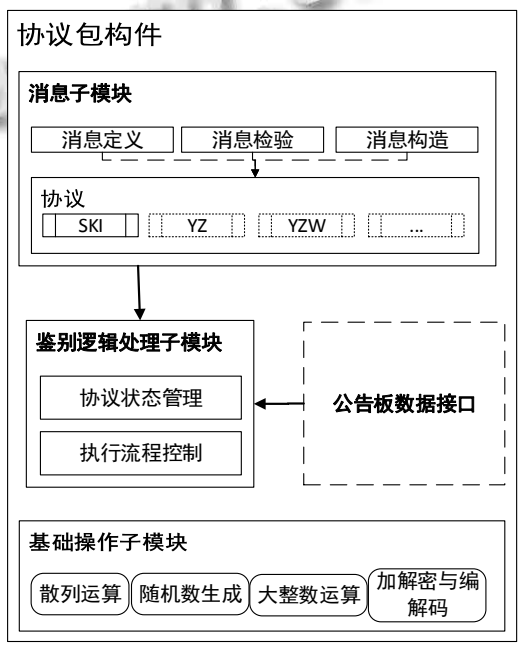

图 3　协议包构件示意图

3.4.1 消息子模块

SKI 鉴别协议的一次执行需要在服务器和客户端之间按序进行多轮的信息交互, 且各轮的消息格式不尽相同, 我们引入消息交互子模块负责定义协议包与外界的消息交互格式, 简化协议包的消息处理过程. 主要功能包括以下几个方面. 1)消息定义. 定义消息种类和数据内容; 2)消息验证. 消息验证指对消息的合法性进行验证. 消息合法性包括两个方面, 一是接收到的消息种类与协议当前的会话状态相符合, 二是消息数据部分包含的各个数据项符合该类消息定义的规范格式. 消息子模块仅检验第二种形式. 3)消息产生. 在外部使用协议包构件时, 消息子模块为其提供利用消息参数构造协议包消息格式的接口.

3.4.2 鉴别逻辑处理子模块

在协议逻辑处理子模块中实现鉴别协议的处理逻辑, 包括解析消息、构造响应、管理协议状态、控制协议执行流程等功能. 其中协议状态管理功能提供了 3.4.1 中描述的第一种消息合法性的保障机制, 即保证收到消息的种类与协议当前会话状态相符合.

协议状态管理: 一次完整的 SKI 协议执行流程会在服务端和客户端之间进行多次有序的消息交互, 对于正常鉴别流程中的每个状态都有其可迁移的后续状态, 鉴于 SKI 协议一问一答式的线性交互机制, 我们引入消息栈来管理会话状态. 首先在协议初始化时, 使用完整的接收消息序列初始化会话栈, 栈顶存放下一个待接收的消息类型. 在收到对方消息时检验其类型是否和栈顶消息类型一致, 若一致则正常处理, 否则拒绝该消息并终止协议. 通过协议状态管理可以实现第一种消息合法性的保障机制.

3.4.3 公告板数据接口

在协议交互鉴别阶段, 服务端和客户端都需要获取协议准备阶段计算出的群组公告信息, 该信息由 3.1 节描述的公告板模块负责维护和发布. 在协议包构件中使用“公告板数据接口”子模块负责处理和公告板的信息交互过程.

3.4.4 基础操作子模块

从系统兼容性和扩展性的角度出发, 我们考察了标准中收录的其他几个鉴别协议, 提取了其公共运算作为系统的基础操作. 主要包括以下几个方面:1)数学运算功能:包括大整数四则运算、模运算、逆元运算、有限域指数运算等; 2)随机数生成; 3)加解密与编解码; 4) 散列计算.

3.5 公告板

SKI 机制在协议准备阶段将计算匿名群组 U 的公开信息$<U_j, C_j>_{1\le j\le n}$,并连同群组标识 U、发布时间 *t0*、有效时长 *T*、 公开参数等一同公布. 在匿名鉴别系统中, 我们引入公告板模块负责以上信息的发布、维护和更新等操作.

客户端以及服务端的鉴别逻辑子模块均可以向公告板发出请求以获取目标群组 U 的公告板信息. 当公告板模块收到对某个群组的信息请求时, 首先检查已经发布的信息中是否存在该群组的公告信息, 并检查其有效期. 若存在 U 的有效公告信息则将其返回给请求者, 否则重新计算并公布 U 的公告信息, 并向请求者返回重新计算后的数据.

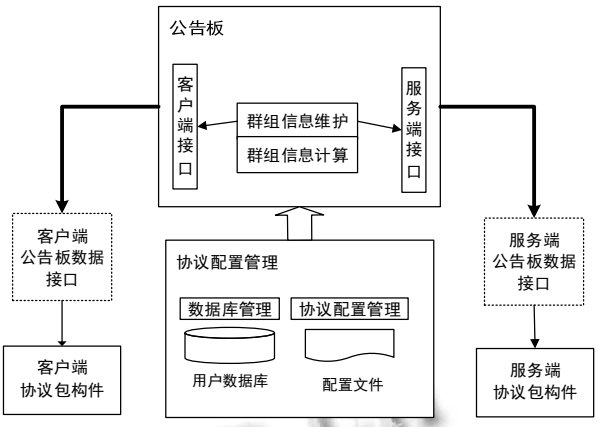

图 4 公告板结构示意图

3.6 消息适配器

消息适配器位于协议包构件和上层应用接口之间, 协议包和上层交互的消息首先通过消息适配器, 由其转化成对端可接受的格式后传递给对端.

上层应用根据实际需求选择合适的通信格式(例如: XML, JSON 等)在网络中传递消息. 而为简化处理, 协议包构件定义了自己独立的消息格式. 因此需要消息适配器在应用和协议包中间提供消息转化的功能. 对于使用不同的消息格式的上层应用, 只需在适配器中定义该格式的消息转化机制, 而协议包构件和上层应用无需改动, 从而提高了程序的扩展性、稳定性和适用性.

3.7 上层应用接口

上层应用位于协议包构件之上, 面向使用浏览器作为客户端的 Web 应用. 上层应用接口主要包括以下

几个模块: 协议协商、会话管理、配置管理等. 另外, 本节还给出了客户端的技术实现方案.

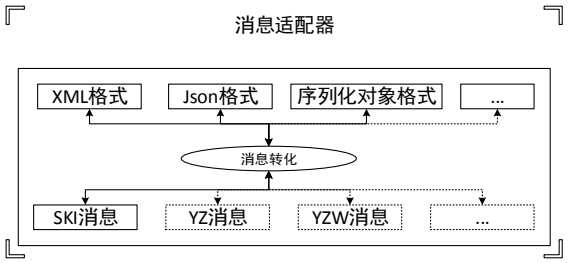

图 5　消息适配器示意图

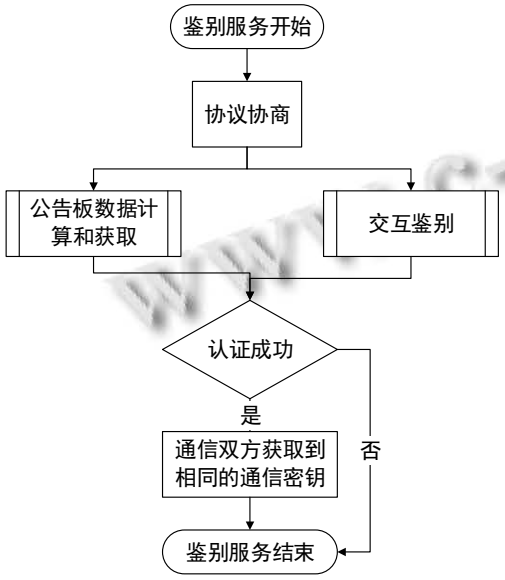

图 6　鉴别服务执行流程图

协议协商: 为了系统能够兼容更多的协议, 我们在鉴别流程开始前增加了协议协商环节用以确定鉴别协议类型以及相关参数. 协议协商模块定义了该环节的执行流程.

会话管理: 匿名口令鉴别协议的执行在客户端和服务端之间进行多次的消息交互, 而当使用 HTTP 等无状态协议作为其数据传输协议时, 将跨越多个 HTTP 会话. 本构件负责此类场景下的应用层会话状态的管理和维护, 保证系统在各种通信模式下的适用性.

配置管理: 配置管理提供协议参数配置管理、网站数据库接入等功能.

客户端实现: 我们考虑以浏览器为鉴别服务访问入口的用户如何执行匿名鉴别逻辑. 在浏览器端调用外部的程序逻辑可以通过以下几种方式实现: 1) 浏览器插件: 如 chrome 浏览器的扩展程序, IE 浏览器的 ActiveX 控件. 但不同内核的浏览器控件存在不兼容的问题. 2)Java Applet: 无需在客户端安装控件, 用户登陆时直接从服务器下载包含客户端程序执行逻辑的 Applet 并在浏览器本地执行. 目前所有的主流浏览器都提供了对 Applet 的支持, 不存在兼容性问题. 所以我们采用 Applet 作为客户端鉴别逻辑的实现方案.

3.8 多级化匿名身份管理

在现实中, 为了提供差别化服务和最大化企业利润, 网站往往对其用户提供会员制服务, 并根据客户价值对其身份进行多级划分, 以实现会员服务定制. 为了满足这层需求, 这就要求鉴别系统在提供匿名性的同时能够识别用户的会员身份等级信息.

SKI 协议的匿名群组天然地提供了对该需求的支持. 通过将同级会员身份的用户聚合到同一匿名群组之下, 并使用树状层级结构可以方便地管理用户的多级会员身份信息. 用户登录时, 网站服务方只能识别出用户所在的群组信息, 该信息即代表了用户的会员等级身份. 而对于用户的具体身份, 网站方则无法获得, 从而同时实现匿名鉴别和用户身份等级识别.

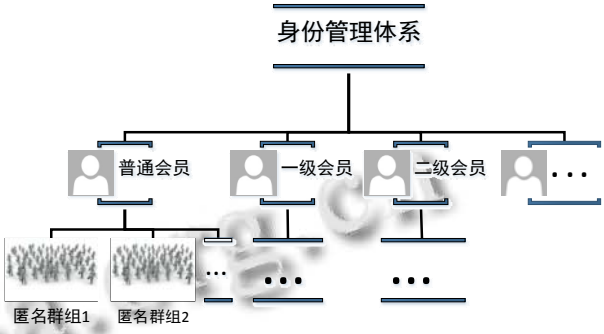

图 7　多级化匿名身份管理体系

4　系统评估

4.1 安全性分析

文献[3]证明了在 CDH 困难假设下, 在随机 oracle 模型中, VEAP 协议具有安全性. 并且文献[3]还证明了该协议对半诚实服务器具有无条件的匿名性. SKI 协议是对 VEAP 的标准化形式, 所以上述结论同样适用于 SKI 协议.

本文假设服务器可信并且用户输入环境安全可靠. 客户端本地输入环境的安全性需要用户实施额外的防护措施. 对于服务器恶意修改协议安全配置等行为不在本文讨论范围.

我们在系统的设计和实现过程中, 发现了协议在

实际应用中存在的一些问题, 并针对这些问题分别提出了解决方案.

4.1.1 公告板信息不一致

在协议的准备阶段, 用户 U_i 将从公告板获取其所属群组 U 的公布信息, 而负责处理 U_i 鉴别请求的服务器 S 则是在协议交互鉴别阶段收到 U_i 的鉴别请求后, 才向公告板查询群组 *U* 的公布信息. 因此, 在同一次鉴别协议执行过程中, 两次对群组 U 公告信息的请求会分别在不同的时间 *t1*, *t2*(*t1*<*t2*)先后到达公告板. 如果在 *t1*-*t2* 之间群组 U 的公告信息发生了更新将会造成该次协议执行过程中用户 U_i 和服务器 S 获取到的 U 的公告信息不一致, 从而导致合法用户认证失败的问题.

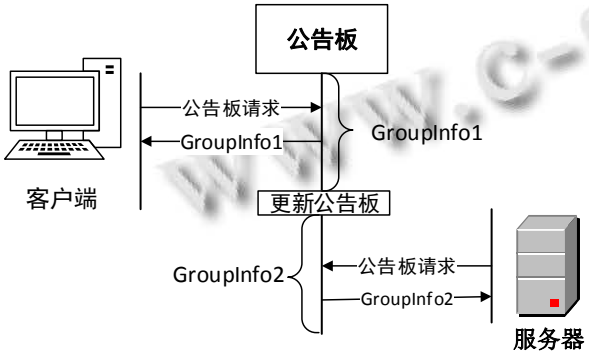

图 8 一次协议执行, 服务器和客户端获取到的 *GroupInfo* 不一致的情景

设$\Delta t=max\{t2-t1\}$为我们允许的正常情况下 *t2* 与 *t1* 的最大时间差. 对于串行执行协议准备阶段和交互阶段的协议方案, 用户 U_i 在获取到 U 的公告信息后才会进入交互鉴别阶段, 那么用户 U_i 向公告板发出的请求将先于服务器 S 到达公告板. 基于串行机制的上述特征, 我们提出“双重公告信息”方案解决用户和服务器获取到的群组公告信息不一致问题. 该方案允许在群组 U 的 *GroupInfo* 过期之后仍然存活Δt 时间, 但是不再作为过期后到达的用户请求的响应.

“双重公告信息”方案:

(1) U_i 从公告板获取到群组 U 的公告信息 *GroupInfo* 后, 提取 *GroupInfo* 的公布时间 *t0*,并在协议执行阶段的启动消息中包含 *t0*, 表明 U_i 所使用的群组公告信息版本.

(2) 服务器 S 收到 U_i 的启动消息, 从中获取参数 *t0* 并判断当前时间 *t* 与 *t0* 的差值是否小于(*T*+$\triangle t$). 若小于则说明公告板中存在群组 U 公布时间为 *t0* 的公告信息, 并将其返回给请求方. 否则重新计算公告信息并发布.

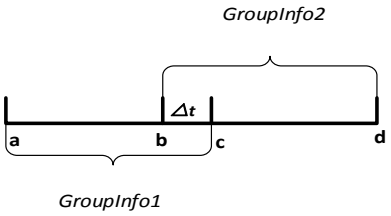

图 9 “双重公告信息”方案示意图. a,b,c,d 表示 4 个时间节点. 其中 *GroupInfo1* 在[a, c]内有效, *GroupInfo2* 在[b, d]内有效.

对于准备阶段和交互鉴别阶段并发执行的执行机制, 无法确定服务器 S 和用户 U_i 对公告板请求的到达次序, 因此无法直接上述方案. 针对并发情景, 我们提出了将请求与会话相绑定的解决方案.

“请求绑定会话”方案:

(1) 用户 U_i 在协议执行前随机生成一串长字符串 *SessionID* 作为其会话标识, 并在请求群组公告信息与交互鉴别阶段的启动消息中同时包含该 *SessionID*.

(2) 在交互鉴别阶段, 服务器收到客户端发来的启动消息后, 将其中包含的群组标志 *U* 和会话标志 *SessionID* 作为参数向公告板发起查询.

(3) 公告板维护 *SessionID* 到会话所使用的群组公告信息的双向映射关系. 当公告板收到 <*U*,*SessionID*>的公告信息请求时, 检查映射关系中是否存在该 *SessionID* 的相应记录, 若存在且其对应的 *GroupInfo* 处于存活期则将该 *GroupInfo* 作为响应返回给请求方, 同时将该映射的双向关系删除. 若不存在, 则返回群组 U 当前有效的公告信息 *GroupInfo* 并创建 *SessionID* 与 *GroupInfo* 的双向映射关系.

(4) 若在查询过程中发现公告信息 *GroupInfo* 已过期$\triangle t$ 时间, 则删除该 *GroupInfo* 所有的映射关系.

在上述两个解决方案中, 服务器 S 除了能获取用户 U_i 使用的群组公告信息版本(必需的)外, 并未获取任何用户相关的其他隐私信息, 不对用户的隐私造成危害. 因此可以有效地解决用户和服务器获取到的群组公告信息不一致的问题.

4.1.2 配置文件安全存储

服务器需要设置协议的安全参数, 包括选用的 hash 函数、有限域群生成元的位数等配置信息. 这些安全参数决定了匿名鉴别系统的安全强度, 如果被敌

手以非正常手段修改, 将对系统安全性造成危害. 因此有必要对系统参数配置文件进行特别保护处理. 鉴于本系统以构件形式向上层应用提供服务, 我们采用“两段式”对称加密存储来保护安全参数文件. 系统使用 DES 算法加密配置文件, 而加密密钥由上层应用负责保管. 并且上层应用可以根据实际情况定期更新加密密钥.

4.1.3 中间人攻击

中间人攻击是指将入侵者控制的计算机虚拟放置在网络中两台通信计算机之间, 在通信双方不知情的情况下对其中发生的网络通信数据进行嗅探和篡改. 当用户处在不安全的网络通信环境中, 恶意攻击者可能采用窃听攻击或者中间人攻击等方式对用户的通信数据进行窃听或者篡改. 由于系统执行鉴别协议过程中, 用户的任何敏感信息(口令、用户名等) 只在本地参与协议的计算过程并不在网络中传输. 并且即使协议执行中的交互消息被恶意篡改, 只会导致协议认证失败, 不存在泄露用户的隐私信息的问题. 因此该系统可以有效防范窃听攻击、中间人攻击等通信信道中发生的攻击方式.

4.2 性能分析

4.2.1 首次请求长延时

从 1.2 节的 SKI 协议准备阶段的描述中得知, SKI 的群组公告信息公布机制要求某个群组 U 的信息公布一定时间后失效. 此后, 对 U 的新请求到达时服务器必须重新计算 U 的公告信息, 而在 U 的信息计算过程中, 对每个成员 U_j 服务器 S 都将进行一次大整数有限域指数运算. 而对于尚未公布公告信息或者信息已过期的群组 U 而言, 首次请求鉴别服务的成员 U_i 需要在服务端进行群组公告信息计算时额外等待. 由实验得出的结果来看, 这一延时随着 U 的规模增大而变得不可忍受.

对此问题, 我们提出带“公告板 Cache”机制来解决这一问题. 公告板 Cache 表示公告板中的一块不对外公布的数据区, 在 Cache 中为每个群组 U 维护其最新尚未公布的公告信息 *CacheInfo*. 当某个请求到达时, 若请求的群组的公告信息已过期, 则使用该群组的 *CacheInfo* 代替原来的过期数据, 并修改当前时间为群组公告信息的发布时间. 该操作的开销仅为特定信息的查找时间, 对比大整数的有限域指数运算可忽略不计. 另外 Cache 中公告信息的计算可利用服务器的空闲时间进行, 因此不会影响用户使用体验. 具体方案如下.

“公告板 Cache”机制:

(1) 服务启动时, 对每个群组计算其公告信息并存储在 Cache 中.

(2) 当公告板中群组 U 的公告信息需要更新时, 在 Cache 中查询 U 的 *CacheInfo* 并将其作为更新后的数据公布. 同时将该 *CacheInfo* 从 Cache 中删除, 并将群组 U 加入到 Cache 的待更新队列中. 只有在 Cache 中不包含 U 的 *CacheInfo* 才需要重新计算 *GroupInfo*.

(3) 服务器空闲时, 若 Cache 待更新队列不为空则从中取出队列头的群组, 计算其公告信息并将结果存储到 Cache 中.

4.2.2 客户端性能优化

公告板的群组公告信息包含了群组所有成员的 $<\underline{U_j}, C_j>_{1\le j\le n}$, 数据量较大, 若每次协议执行都将 $<\underline{U_j}, C_j>_{1\le j\le n}$ 发送至客户端, 必将给服务器的通信带宽带来巨大压力. 由于根据群组公告信息中包含的发布时间和有效时长可以判断该信息是否过期, 我们提出下面的方案来优化客户端的通信性能.

“客户端 Cache”机制:

(1) 当用户 U_i 从服务端公告板获取到所属群组 U 信息后, 将该信息在本地保存.

(2) 在下次执行鉴别协议时, 优先查看本地保存的群组公告信息是否过期. 若不过期则向服务端发送本地信息副本的发布时间, 请求服务器对该信息的有效性进行确认.

(3) 若客户端得到肯定回复则使用本地保存的信息副本用于之后的鉴别过程. 否则, 重新发起对 U 的公告信息请求.

5 实验测试和分析

首先, 我们将从理论上分析 SKI 协议执行过程中服务器和用户各自所必需的开销, 之后分别设计实验测试了不同参数条件下系统执行一次鉴别服务的开销并对实验结果进行分析.

5.1 **SKI** 执行开销理论分析

SKI 协议执行过程中涉及的计算主要包括: 有限域指数运算、有限域乘法运算、加解密运算以及散列运算等. 对于协议执行的计算开销, 我们考虑由于协议选取的群的阶 q 以及群生成元 g 的位数都是

1024/2048bit 级别的大整数，与有限域指数运算相比，其他运算所消耗的时间很短，可以忽略不计. 因此我们将有限域的指数运算作为衡量方案计算效率的主要指标.

表 1 协议开销分析

有限域指数运算次数				通信开销
用户		服务器		
Total	T-P	Total	T-P	
5	1	n+5	3	N*|ε|+5*|q|+2*|H|

|.| 表示变量的比特长度. N 表示群组包含的成员数，“Total”表示协议执行所需的全部计算开销，“T-P”表示除去协议预计算后还需进行的运算. 通信开销栏目中的|ε|表示<$\underline{U_i}$, C_j>的长度，该长度与协议选用的散列函数相关，|q|表示群的阶 q 的位数,|H|表示 SKI 中生成鉴别标志 Vs 和 Vu 的散列结果的长度. 如果选用最低安全级别的参数，那么|ε|=128, |q|=1024, |H|=160.

5.2 实验设计

在构件系统之上，我们实现了基于 Web B/S 模式的简单匿名鉴别应用原型以测试系统性能. 原型服务端提供通信支持、数据库服务等功能. 浏览器采用 Java Applet 技术用以执行鉴别逻辑. 根据 4.1 小节的分析，通信双方在通信信道中不会传递敏感信息且协议具有良好的安全性. 鉴于以上因素，我们采用 HTTP 作为浏览器和服务器之间的通信手段.

基于实现的原型，我们对匿名鉴别系统的性能指标进行了测试. 首先我们分别构建包含[25,50,100,150,200,300,400]成员的群组来覆盖不同的群组规模大小. 对于各个规模不同的群组，测试在群的阶 q 分别为 1024bit 和 2048bit 时系统的性能表现. 每一种群组规模搭配一类群的阶为一组实验，在实验中测量系统在各个阶段时间开销，每组实验进行 100 次取测量数据的均值作为最终结果.

客户端和服务器分布在同一局域网内，我们基于 Cloudstack 云计算平台创建的虚拟机搭建匿名口令鉴别服务器，每个虚拟机的配置为：CPU 20*1.00GHz，内存 20GB. 鉴别服务运行在云端的各个虚拟机中. 客户端使用 chrome 浏览器，所搭载的主机配置为：CPU 2.93GHz，内存 4GB.

表 2 实验结果，群的阶 q 为 1024 位

群组 ID 数	群的阶 q 为 1024 位，执行时间开销(ms)				用户端延时总开销(ms)
	协议协商	公告信息预计算	交互鉴别阶段计算		
			用户	服务器	
25	262	131	146	95	855
50	268	186	140	102	905
100	262	308	140	96	884
150	262	442	170	98	854
200	252	568	142	94	842
300	276	809	142	94	868
400	257	1078	154	100	830

表单数据说明(同表 3)，“协议协商”表示用户和服务器对所使用的协议和协议配置参数进行协商的过程. “公告信息预计算”表示服务器对目标群组 U 的公告信息进行计算的过程，对应于 SKI 机制的准备阶段. “交互阶段计算开销”表示用户和服务器在交互鉴别阶段执行协议计算、处理交互消息所需的时间开销. “用户端延时总开销”表示用户从启动匿名口令鉴别协议开始到完成身份鉴别所需的总时间，反映用户的使用体验.

表 3 实验结果，群的阶 q 为 2048 位 25

群组 ID 数	群的阶 q 为 2048 位，执行时间开销(ms)				用户端延时总开销(ms)
	协议协商	公告信息预计算	交互鉴别阶段计算		
			用户	服务器	
25	255	677	1069	748	2906
50	270	1136	1058	733	2853
100	262	2062	1081	710	2837
150	267	3017	997	681	2831
200	254	4175	1027	761	2811
300	258	5805	1020	747	2842
400	269	7685	1140	735	2856

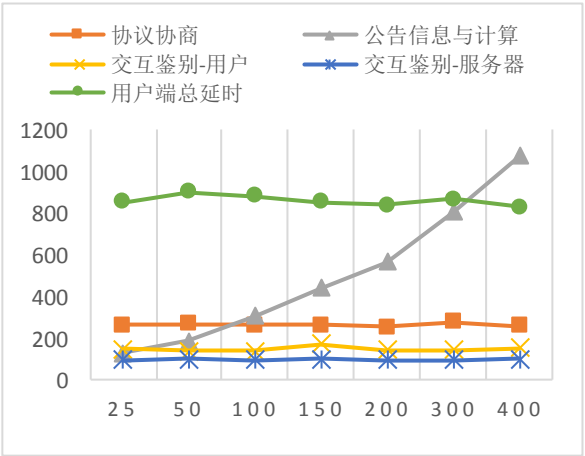

图 10 群的阶 q 为 1024 位时，系统时间开销

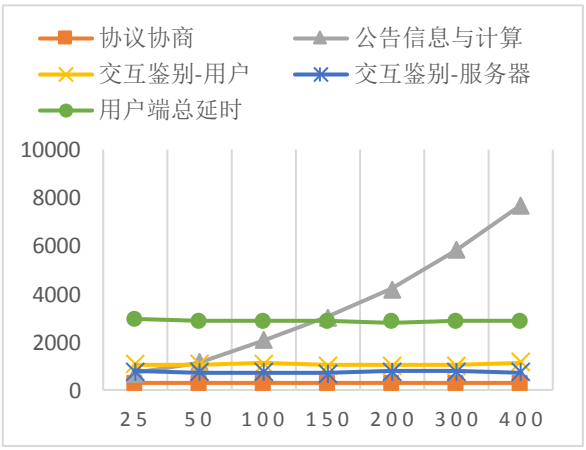

图 11 群的阶 q 为 2048 位时，系统时间开销

5.3 实验结果分析

从上面的实验结果可以看出，协议协商的时间开销基本为常量，分析可知其仅与用户所处的网络状况相关. 交互阶段的计算开销以及用户端延时总开销不随群组规模的增长而有明显变化，其主要受群的阶 q 的位数决定. 当 q 由 1024 升级为 2048 位时，除“协议协商”外的各项开销均有明显增长，因此在现实应用时可根据需求通过调整 q 的位数从而在系统安全性和性能之间进行平衡. 对于用户而言，鉴别过程的时延总计最长不超过 3 秒，在可接收范围之内. 虽然“公告信息预计算”的开销随群组 ID 数目增加而线性增长，但采用 3.1 节中提出的“Cache 机制”，可以在服务器空闲时对该信息进行预计算，并不影响用户体验.

6 结语

针对互联网领域的个人隐私保护问题，本文研究设计了一种匿名口令鉴别构件系统，系统基于 SKI 协议进行了实现，以构件形式为互联网应用提供匿名鉴别服务. 我们发现并解决了 SKI 协议在实际应用时产生的多个问题，并针对系统性能，分别对服务端和客户端提出了改进方案. 结合现实应用特点，我们提出“多级化匿名身份管理”体系，丰富匿名口令鉴别的应用场景. 最后设计实验测试了多种不同协议参数下系统的性能，验证了系统的具有较强的可用性.

参考文献

1 ISO/IEC DIS 20009-4 Information technology -- Security techniques – Anonymous entity authentication – Part 4: Mechanismsbased on week secrets. 2014.

2 SeongHan S, Kobara K. Anonymous password- authenticated key exchange: New construction and its extensions. IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences, 2010, 93(1): 102–115.

3 Yang J, Zhang Z. A new anonymous password-based authenticated key exchange protocol. Progress in Cryptology-INDOCRYPT 2008. 2008, Springer. 200–212.

4 Yang Y, et al. Towards practical anonymous password authentication. Proc. of the 26th Annual Computer Security Applications Conference. 2010. ACM.

5 Chaum D. Security without identification: Transaction systems to make big brother obsolete. Communications of the ACM, 1985, 28(10): 1030–1044.

6 Chaum D, Evertse JH. A secure and privacy-protecting protocol for transmitting personal information between organizations. Advances in Cryptology—CRYPTO’86. Springer. 1987.

7 Lysyanskaya A, et al. Pseudonym systems. Selected Areas in Cryptography. Springer. 1999.

8 Brands SA. Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy. Mit Press, 2000.

9 Camenisch J, Van Herreweghen E. Design and implementation of the idemix anonymous credential system. Proc. of the 9th ACM Conference on Computer and Communications Security. ACM. 2002.

10 Camenisch J, Lysyanskaya A. A signature scheme with efficient protocols. Security in Communication Networks. Springer. 2002. 268–289.

11 Camenisch J, Lysyanskaya A. Signature schemes and anonymous credentials from bilinear maps. Advances in Cryptology–CRYPTO 2004. Springer. 2004.

12 Tsang PP, et al. PEREA: Towards practical TTP-free revocation in anonymous authentication. Proc. of the 15th ACM Conference on Computer and Communications Security. ACM. 2008.