一种公钥密码体制下指纹识别与数字水印的身份认证协议*

蔡龙飞[1]，赵慧民[2]，方艳梅[3]
（1. 广东工程职业技术学院计算机信息系，广东 广州 510520；
2. 广东技术师范学院电子与信息学院，广东 广州 510665；
3. 中山大学信息科学与技术学院，广东 广州 510006）

摘 要：在公钥密码 PKI（Public Key Infrastructure）安全体制下，结合指纹识别与数字水印的各自特性，提出了一种基于指纹特征作为数字水印的网络身份认证协议。由数据交互过程的分析可见，此协议方案能充分保证用户密钥和指纹信息的保密性和真实性，并能有效抵抗 Stolen-verifier 和 Replay Attack 两种攻击，可用于电子商务环境下高安全性的身份认证系统。

关键词：指纹识别；数字水印；身份认证；协议
中图分类号：TN918 **文献标志码：**A **文章编号：**0529－6579（2013）04－0051－07

An Identity Authentication Protocol of the Public Key Infrastructure Combining Fingerprint Identification with Digital Watermarking

CAI Longfei[1], *ZHAO Huimin*[2], *FANG Yanmei*[3]
（1. Department of Computer Information, College of Engineering Occupation Technology, Guangzhou 510520, China;
2. School of Electronic and Information, Guangdong Polytechnic Normal University, Guangzhou 510665, China;
3. School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510275, China）

Abstract: Combining characterics of fingerprint identification with digital watermarking technology, a novel identity authentication protocol is proposed in the framework of public key infrastructure (PKI) network security theory. By the interactive data process, the protocol can fully achieve important performances of the security and authenticity for user key and fingerprint information, and can safely resist for the two common attacks of Stolen-verifier and Replay, therefore the protocol can apply for identity authentication system in network e-commerce environment.

Key words: fingerprint identification; digital watermarking; identity authentication; protocol

认证（Authentication）又称鉴别，是指对用户身份的确定，它是防止非法人员对系统进行主动攻击的重要技术。认证技术主要包括信息认证与身份认证两个方面的内容，信息认证主要用于保证信息的完整性与抗否认性，身份认证则用于鉴别用户身份，限制非法用户和用户非法使用网络信息系统。

身份认证常常被用于通信双方相互确认身份，以保证通信的安全。其基本思想是通过验证被认证对象的某一特殊属性来达到确认被认证对象是否真实有效的目的。被认证对象的属性可以是口令、证

* **收稿日期：**2013－01－07
基金项目：国家自然科学基金资助项目（61272381）；广东省自然科学基金资助项目（S2012010008639）；广东省科技计划资助项目（2012B010100035）.
作者简介：蔡龙飞（1976 年生），男；E-mail：cailongfei2@126.com

书、数字签名或者像指纹、声音、视网膜这样的生理特征。当前，网络上流行的身份认证技术主要有基于口令的认证方法、基于智能卡认证、动态口令认证、生物特性认证、USB Key 认证等，这些认证技术并非孤立或单独使用，有很多认证过程同时使用了多种认证机制，结合各自特点达到更加安全可靠的目的[1-3]。

基于生物特征的用户身份认证技术是通过计算机利用人体所固有的生理特征或行为特征，如指纹、手形或视网膜等来进行的个人身份鉴别。目前生物认证技术已被广泛使用，其中，指纹技术具有通用性、唯一性、持久性和易采集性等优点，相对于其它生物认证技术更加便捷可靠。所以，它是目前基于生物特征的身份认证方式中应用最为广泛的一种方法[4]。

但是，在大多数生物特征身份认证技术中，认证信息几乎都是以明文的形式在网络上传播，很容易被篡改或受到重发攻击，没有达到认证数据机密性和完整性的要求。主要存在问题如下：

1）传输指纹信息的过程中并不能保证指纹信息不被截获，即使通过加密手段也很容易被非法第三方截获后解析出来，一旦用户指纹落入他人之手，后果则不堪设想。

2）存放生物特征数据的特征数据库本身并不具有安全保密性，而且对于需要客户端的系统来说，也不可能将整个数据库都存放在每个终端当中。指纹特征信息数据库的安全性问题成为了进一步提高系统安全性的瓶颈。

因此，要实现指纹识别认证系统在现实中安全有效的应用，必须引入一种针对于特定应用环境下的认证方案，既能完成录入、对比、认证等过程，又能有效的保护特征参考信息。

数字水印技术的发展为生物身份认证技术提供了一种安全可靠的信息隐藏通信技术途径。数字水印技术是将数字、序列号、文字、图像标志等版权信息嵌入到多媒体数据中，以起到版权保护、秘密通信、数据的真伪鉴别等作用[5-7]。然而，目前把数字水印应用于身份认证也存在有待解决的几个问题：

1）水印信号的唯一性问题一直没有得到很好地解决。Yeung 和 Pankanti 提出了在指纹图像模板中嵌入水印，以防止指纹模板被恶意篡改[8]。但是，水印以噪声的形式嵌入指纹模板，必然会影响指纹的特征提取，从而影响系统判断成功率，降低整个系统的可靠性。

2）水印信号与用户身份并不具有严格的相关性。水印信号标识并不能完全确定的代表与之相关的用户身份。

为此，从提高身份认证可靠性和安全性的角度出发，本文将数字水印与指纹识别技术相结合，利用二者各自特点，提出并建立了一种网络身份认证的新模型和新协议，用于实现对用户身份的双重认证，防止非法用户进入系统盗取重要机密，防止合法用户访问非授权文件和文件夹，进一步保障网络信息的安全应用[9-10]。

1 指纹特征作为数字水印的身份认证系统模型

为了解决身份认证的可靠性和安全性，将数字水印与指纹识别技术相结合，提出一种网络身份认证新模型，如下图 1 所示。。这种模型采用两个物理认证因素：一个是用户的 ID 和口令信息，另一个是用户的指纹特征信息。这样，就使得用户身份认证的确定性按指数级递增，是一种强身份认证方案。总体来说，由指纹信息提取、水印生成、嵌入水印、传输载体信息、水印提取和检测等部分组成。当然，在系统对用户身份认证之前，服务端应该具有用户的注册信息数据库。系统客户端一方在水印信息嵌入完成后将载有水印信息的载体信息通过某种协议方式安全传送至服务端，服务端通过解密提取出水印信息后，与服务端的原有信息库中信息比对，完成身份验证的功能。

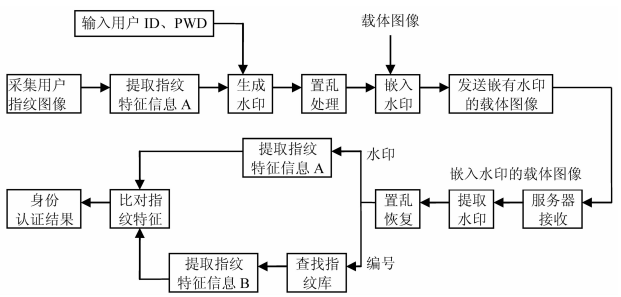

图 1 基于指纹作为水印的网络身份认证系统模型图

Fig. 1 System model of network identity authentication based on fingerprint watermarking

基于指纹水印的网络身份认证系统工作流程分如下几个步骤：

1）信息提取。此处所说的信息提取包括两部分：对用户 ID 和 Password 等基本信息的提取；同时，通过指纹采集设备和技术提取此用户指纹特征信息 A。将提取到的指纹信息与用户 ID 建立对应

关系。

2）水印生成。将上述两者信息结合来构成对应与此用户的数字水印。然后对其进行 ASCII 码形式连续排列。

3）嵌入水印。根据载体特征和水印特征使用扩频载波调制的嵌入算法将用户信息嵌入某个指定的载体[8]。

4）传输信息。用户端将已经嵌入水印的载体信息发送至服务器端。但此处的发送方式应该通过一种安全有效的通信协议方式来完成（这是为了防止非法第三方冒充客户端或服务端）。

5）水印提取和匹配。服务端接受到信息后，通过密钥对载体解密提取出用户身份信息。此时，需要对提取出来的信息进行双层验证。根据用户 ID 和 Password 到服务器用户数据库中查找并比对，如果信息比对一致，则进入第二层验证过程，即根据用户 ID 从服务器端的指纹特征库中查找出相对应的用户指纹信息 B，此时，根据某一特定匹配算法对特征信息 A 和 B 进行匹配。如果匹配成功，那么身份认证成功，此用户是具有合法性。

由系统实现步骤可见，所有的信息资源访问权限都在身份认证系统（服务器端）的管理之下，经过两个物理认证因素的验证，确保了信息资源访问的合法性。这样，将数字水印与指纹识别技术相结合，实现了用户合法身份的双重认证：

- 水印信息中的 ID、口令—第一重认证。服务器端从用户端发送来的载体图像中提取水印信息，该水印信息中包含用户的 ID、口令，若 ID 和口令不正确，则可判断为非法用户。
- 指纹特征信息—第二重认证。服务器端将从用户端发送来的载体图像中所提取的指纹特征信息 A 与存储在指纹特征库中的该用户的原始指纹特征信息 B 进行比较，再次验证用户的合法性。

在本文提出的基于数字水印和指纹识别技术的网络双重身份认证过程中，由于水印信息量较少，一般在几 kb 到十几 kb，通过控制嵌入强度和嵌入的水印信息量，可以将水印嵌入并且隐藏在载体图像中，使得水印的嵌入不会对载体图像构成影响，人们也很难用肉眼观察出载体图像的失真，不易引起攻击者的注意，起到了一定的保护作用。

2　基于指纹识别与数字水印的身份认证传输协议

2.1　协议的总体方案

本文采用的双重身份认证方案一方面运用公钥基础设施 PKI 提供的 CA 证书服务对系统交互双方分发数字证书并相互进行验证，另一方面，在此基础上将系统用户生物指纹与用户基本信息以及时间信息等一并作为数字水印嵌入到传输载体，通过安全的 SSL 协议通道发送至服务器端，实现对系统操作方的身份验证。本文所提及的协议主要是根据方案设计建立在系统应用层之上，它包括两个阶段：注册阶段和认证阶段。注册过程发生在一个新用户想要加入到认证系统时，每个用户只进行一次注册过程，实际注册的内容将在后面详细描述。而认证过程发生在用户每次登陆系统和进入系统后进行关键性操作时，认证的目的是数据交互双方互相验证对方身份合法性。

协议的参加方主要有：用户（客户端）A、服务器端 S、管理员 M、CA 认证中心。其中，CA 认证中心用于产生根证书，并根据用户、服务器和管理员的请求生成数字证书颁发给各请求方。管理员在参与方中充当可信第三方的角色。在实际应用系统中，一般 CA 认证中心还需有 RA 注册中心、KMC 密钥管理中心、证书库/CRL 发布与查询系统等部分配合完成，它们综合起来称之为 PKI 公钥基础设施[11]。整个 PKl 基础设施为用户提供身份注册，证书签发、撤销、查询，证书撤销列表 CRL 签发、查询，以及密钥对的产生与备份等服务。基本物理平台构架如图 2。

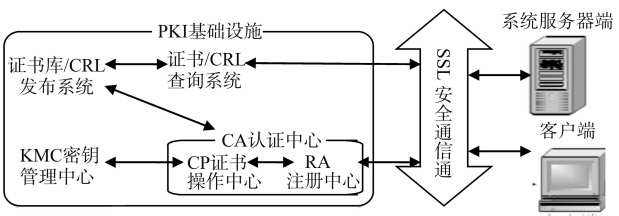

图 2　基本物理平台构架图

Fig. 2　The physical building platform of the identity authentication system

2.2　协议方案要求

协议要求从两方面来考虑：首先，应当保证系统提供服务的安全性，合法的数据交互不受非法攻击；其次，应当保证系统的高效性。从系统的安全性考虑，本系统认证服务应提供如下的功能：

1）参与方之间的相互认证：包括管理员和客户端用户之间的双向认证，管理员和服务器端之间的双向认证，客户端用户和服务器端之间的双向认证。以上认证都要防止非法攻击。

2）数据传输的机密性保证：数据以密文形式

在网络中传输并保存在数据库中，以保证数据不会泄漏，只有数据接收方使用正确密钥解密，才能获取明文数据。

3）防止 Stolen-verifier 攻击：防止攻击者使用从认证服务器中盗窃的用户身份和指纹信息冒充合法用户进行系统登录与访问。

4）防止重发攻击：防止非法用户利用合法实体以前使用过的信息进行非法的系统访问。从用户的角度来讲，指纹是无需用户记忆和携带的口令或密码，而从系统的角度看，指纹仅仅是一个比特串，与任何其它的密钥没有什么不同。在网络上，如果没有任何保护措施，基于指纹的身份认证系统同样对重发攻击没有免疫力，攻击者能够简单地绕过指纹采样设备，直接将一个比特串发往认证服务器，从而假冒合法用户的身份访问系统资源。因此，像密钥一样，指纹认证中的指纹数据必须得到保护。

2.3 协议注册过程描述

当用户注册其指纹特征时，需要到管理员处，由管理员审核其指纹特征数据的真实性，并由管理员负责将用户的指纹注册到服务器 S 。图 3 表示用户注册协议流程。注册过程具体描述如下。

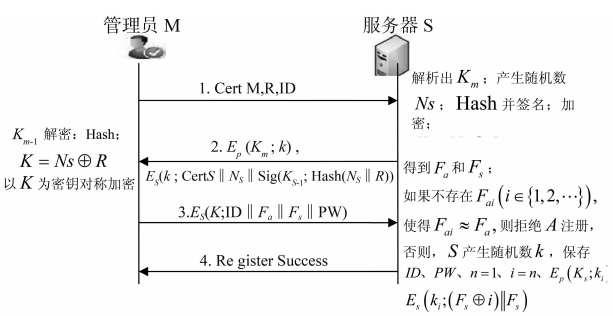

图 3 注册协议数据交互实现流程

Fig. 3 Implementation process of data interaction for user registration

消息 1：管理员 M⇒服务器 S：CertM，R，ID.

用户在管理员处申请注册用户 ID，由管理员 M 向服务器 S 发出一个注册请求消息，消息包括管理员证书 CertM 、注册请求随机数 R 和用户 ID。服务器收到管理员的注册请求后，首先通过 PKI 提供的证书验证服务对 CertM 进行验证。如果验证不正确，则认为此管理员不合法，协议中断；如果验证正确，则从证书 CertM 中解析出服务器的公钥 K_m，认为它是某个合法管理员的公钥。但此时，还不能确定 CertM 是否为被某一非法用户窃取后进行的冒名攻击。之后，服务器生成消息 2 并发送至管理员 M，管理员通过收到的消息 2 验证服务器身份。

消息 2：服务器 S⇒管理员 M：$E_p(K_m;k)$，$E_s(k;\text{CertS} \| N_s \| \text{Sig}(K_{s-1};\text{Hash}(N_s \| R)))$.

服务器生成一个随机数 N_s，将 N_s 与 M 发送来的 R 连接并对其哈希，而后用服务器私钥 K_{s-1} 对此哈希值签名，同时生成随机数 k 作为对称密钥，用解析出的管理员公钥 K_m 对 k 机密，用 k 对服务器证书、随机数 N_s 和签名值进行对称加密，并发送至管理员 M。

管理员 M 收到消息 2 后，用私钥 K_{m-1} 解密消息，获得服务器证书 CertS、随机数 N_s，此时，管理员通过 PKI 对 CertS 进行验证，如果不正确，则认为服务器是非法的，协议就此中断；如果正确，则从证书 CertS 中解析出服务器公钥 K_s，并认为 K_s 为合法服务器的公钥。而后，用服务器公钥 K_s 对签名值进行 $\text{Sig}(K_{s-1};\text{Hash}(N_s \| R))$ 解密，得到哈希值 Hash（$N_s \| R$）。此时，管理员 M 已经得到服务器发送来的随机数 N_s，且拥有自己产生的随机数 R，管理员 M 用相同算法对 $N_s \| R$ 进行哈希运算得到 Hash´（ $N_s \| R$）。这时，对比两个哈希值，如果相等，说明交互对方服务器确实为合法服务器。这样便有效防范了某个非法服务器在窃取合法服务器的证书 CertS 后进行假冒攻击。

消息 3：管理员 M⇒服务器 S：$E_s(K;\text{ID} \| F_a \| F_s \| PW)$.

管理员收到消息 2 并对服务器身份验证通过后，将解析所得的随机数 N_s 与随机数 R 异或得到新的对称交互密码 $K = N_s \oplus R$。由于 N_s 和 R 只有管理员 M 和服务器 S 知道，故对称密钥 K 是安全的。

而后，管理员 M 用对称密钥 K 对消息 $\text{ID} \| F_a \| F_s \| \text{PW}$ 进行加密，并发送至服务器端。其中，F_a 是管理员指纹特征数据，F_s 是新用户的指纹特征数据，PW 是用户设置的密码。

服务器收到消息 3 后，用本地计算出的 $K = N_s \oplus R$ 对其解密，获得 ID、PW、F_a 和 F_s，并验证此 ID 是否为消息 1 发送来的用户 ID ，如果不一致，则认为消息 3 为非法用户所发送，协议中断；如果一致，则说明消息 3 的发送方仍然是之前交互的合法新用户。

服务器端将解密出的 F_a 与服务器端存储的管理员指纹数据 F_{ai}（$i=1$，2，...）进行 1：N 匹配，判断是否有 $F_{ai} \approx F$。如果不存在任一 F_{ai} 与 F_a 匹配

成功，则管理员的身份验证不通过，用户指纹不能注册。否则，管理员指纹匹配成功，服务器端生成随机数 k 作为对称加密密钥，将用户 ID、PW、$E_p(K_s;k_i)$、$E_s(k_i;(F_s\oplus i)\parallel F_s)$ 和认证成功计数器 $n=1$ 保存在后台安全数据库，作为用户原始注册数据，其中 $i=n$。

消息 4：服务器 S⇒管理员 M：Register Success.

服务器 M 对用户信息成功注册后，向管理员 M 回送注册成功消息 4。

2.4　协议认证实现过程

当用户 A 申请登录系统时，触发认证协议。认证协议分 Time1 和 Time2 两个阶段。Time1 阶段用于对普通用户登录时的身份认证，认证通过用户进入系统，服务器根据用户身份对其设定访问权限，用户可以进行浏览、查看等普通操作；Time2 阶段用于用户要实施关键性操作时对用户身份再次进行验证，这次验证通过判断用户的指纹特征是否与用户声称身份相符的方式来实现。这样，有效的防止了非法用户在窃取合法用户 ID、密码和数字证书等私有信息后进入系统实施数据盗用和数据毁坏等非法操作。

协议描述　客户端与服务器连接成功后，用户 A 在客户端向服务器 S 发送登录请求，认证协议被触发，认证协议的第 i（$i\geqslant 1$）次认证过程如图 4 所示。

用户 A 的第 i（$i\geqslant 1$）次认证过程具体描述如下：

消息 1：用户 A⇒服务器 S：$R\parallel \mathrm{ID}\parallel E_s(\mathrm{PW};\mathrm{ID}\parallel \mathrm{CertA}\parallel R)$.

用户 A 请求登录系统时触发认证协议，用户端将用户登录口令 PW 作为对称加密密钥，对用户 ID、随机数 R 和用户数字证书进行加密，并将加密后的数据与明文 R 和 ID 发送至服务器端。

服务器端接收到消息 1，首先根据收到的 ID 查看数据库，判断该 ID 是否已注册。如果此用户并未注册，则拒绝访问，协议中断；如果此 ID 已注册，用数据库中此 ID 对应的口令 PW 对 $E_s(\mathrm{PW};\mathrm{ID}\parallel \mathrm{CertA}\parallel R)$ 解密，如果能正确解密出 ID 且与明文 ID 一致，且验证证书 CertA 合法，则服务器 S 认为消息 1 确是由拥有此 ID 和 PW 的用户所发送，并且消息完整未被改动。否则，服务器 S 认为用户 A 的 ID 被盗用或消息 1 在传输过程中被修改，协议中断。

消息 2：服务器 S⇒用户 A：$\mathrm{OK}\parallel E_s(\mathrm{PW};\mathrm{CertS}\parallel R)$.

服务器 S 同样用此 ID 的口令 PW 作为对称密钥，对证书 CertS 和 R 加密，连同认证通过消息 OK 发送给用户。

用户接收到消息 2，用 PW 解密获得服务器证书 CertS 和随机数 R，对证书验证合法并且 R 与消息 1 中生成的随机数 R 相同，则认为消息 2 是合法服务器发来的数据，用户 A 确信服务器发来的认证通过 OK 消息。

至此，用户身份的第一层认证结束，用户可以进入系统访问应用服务器，但用户的操作权限受限，只能实施诸如浏览、查询等普通操作。当用户要进行关键性操作时，触发以下第二层认证协议。

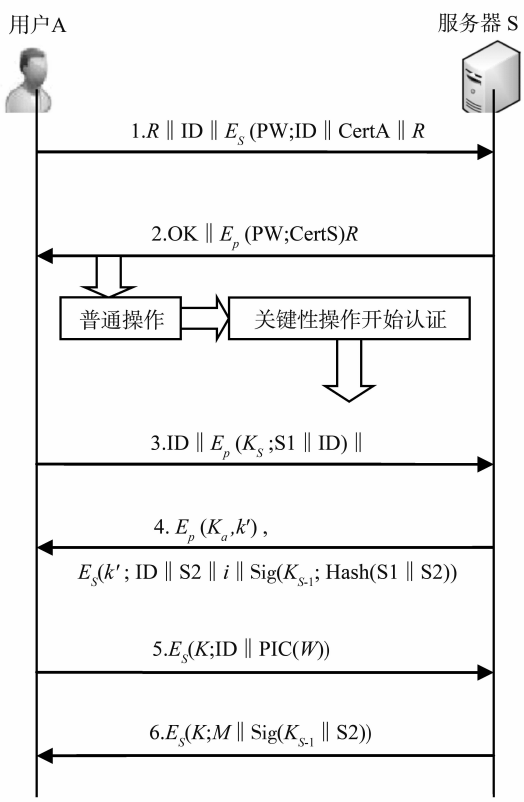

图 4　认证协议实现流程

Fig. 4　Implementation process of the authentication protocol

消息 3：用户 A⇒服务器 S：$\mathrm{ID}\parallel E_p(K_s;\mathrm{S1}\parallel \mathrm{ID})$.

用户触发第二层认证协议，客户端产生认证请求随机数 S1，并用解析证书 CertS 得到的公钥 Ks 对 CertS 和用户 ID 进行加密，连同明文 ID 发送给服务器 S 端。

服务器 S 收到消息 3 后，知道是具有此 ID 的用户请求认证，S 用私钥 $K_{s\text{-}1}$ 解密得到 S1 和用户 ID，判断解密所得 ID 是否与明文 ID 一致，如果不

一致，则协议中断；否则，服务器S认为消息3可靠。S产生一随机数S2，并对接收到的消息1中的CertA解析得到用户公钥K_a。

消息4：服务器S⇒用户A：$E_s(K_a;k')$,$E_s(k';\mathrm{ID}\parallel \mathrm{S2}\parallel i\parallel \mathrm{Sig}(K_{s-1};\mathrm{Hash}(\mathrm{S1}\parallel \mathrm{S2})))$.

服务器生成随机数k作为对称密钥，并用公钥Ka对其加密，同时S对两个随机数的连接哈希得到一个散列值Hash（S1‖S2），并用私钥K_{s-1}对其签名。然后，S将ID、S2、i和哈希值用k对称加密，并发送给用户A，其中i是后台安全数据库中所存的此用户的认证成功计数器n的值。

用户A接受到消息4后，用私钥K_{a-1}解密。然后，使用用户公钥K_s对签名值$\mathrm{Sig}(K_{s-1};\mathrm{Hash}(\mathrm{S1}\parallel \mathrm{S2}))$解密，得到哈希值Hash（S1‖S2）。此时，用户已经得到服务器S发送来的随机数S2，且拥有自己产生的随机数S1，用户A用相同算法对S1‖S2进行哈希运算得到Hash′（S1‖S2）。这时，对比两个哈希值，如果相等，说明交互对方服务器确实为合法服务器，这样便有效防范了某个非法服务器的假冒攻击。

消息5：用户A⇒服务器S：$E_s(K;\mathrm{ID}\parallel \mathrm{PIC}(W))$.

用户根据S1、S2产生新的会话对称密钥$K=\mathrm{S1}\oplus\mathrm{S2}$。同时，采集用户指纹特征信息$F_u$，计算$W=E_s(K;(F_u\oplus i)\parallel F_u)$，使用密钥$K$将$W$作为水印信息根据约定好的水印嵌入方法嵌入到载体图像PIC中得到PIC（W），将ID和PIC（W）用密钥K加密，将载体图像加密为密文发送至服务器S。

服务器S接收到消息5后，用密钥$K=\mathrm{S1}\oplus\mathrm{S2}$对消息解密得到ID和嵌有水印的图像PIC（$W$），服务器端使用密钥$K$提取出水印信息$W=E_s(K;(F_u\oplus i)\parallel F_u)$，同时判断W是否等于数据库中注册的ID对应的$E_s(k;(F_s\oplus i)\parallel F_s)$，如果不相等，协议中断；否则，S解密W，得到$F_u\oplus i$与$F_u$。此时，计算$F_u\oplus i_u\oplus n$是否等于$F_u$，如果不相等，协议中断；否则，对$F_u$和$F_s$进行相似度匹配，如果匹配成功，即$F_u\approx F_s$，则认为用户A的指纹特征数据合法，验证通过。同时，将数据库中的用户注册数据ID、PW、$E_p(K_s;k_i)E_s(k_i;(F_s\oplus i)\parallel F_s)$、$i=n$更新为ID、PW、$E_p(K_s;k_{i+1})$、$E_s(k_{i+1};(F_u\oplus(i+1))\parallel F_u)$、$i=n$，其中，$n=n+1$。如果匹配不成功，则用户认证失败，协议中断。

消息6：服务器S⇒用户A：$E_s(K;M\parallel \mathrm{Sig}(K_{s-1};M\parallel \mathrm{S2}))$.

服务器在确认了用户的合法身份后，产生指纹认证成功的消息M，用私钥对K_{s-1}M‖S2进行数字签名，然后用对称密钥K对M和签名加密形成消息6发送给用户。

用户收到消息6后，使用密钥K对消息解密，获得指纹认证成功消息M。但此时，用户并不能确认消息是否为某个非法认证服务器的假冒攻击，所以用服务器公钥K_s对服务器签名进行验证，若不正确，用户认为消息为假冒服务器所发；若正确，用户认为对方服务器是合法的，指纹认证成功消息可靠。

至此，用户A与服务器S相互认证完毕，用户可以在系统中进行关键性操作。

3 协议实现分析

下面从指纹数据的保密性和真实性、Stolen-verifier攻击以及重放攻击三方面对协议实现过程进行分析。

3.1 指纹数据的保密性和真实性

在协议中，公钥算法和对称密码算法分别使用RSA与AES，其中RSA的密钥长度为1 024 bit，AES算法的密钥长度为128 bit，我们知道密钥长度为1 024 bit的RSA算法和密钥长度为128bit的AES算法在计算上是安全的[11-12]。同时，对于用户指纹敏感信息做了特殊处理，将其作为数字水印嵌入到载体图像，对载体图像以密文的形式传送。这样，攻击者无法从中获取用户的指纹等敏感信息，保证了交互过程中数据的保密性。

由注册协议分析可知，攻击者假冒用户A进行指纹注册不可行，故服务器后台数据库存放的都是合法用户的真实指纹数据。在认证阶段，如用户冒名以他人身份登录系统并使用假冒指纹F_u'认证，服务器端获取F_u'与用户所声称的ID对应的数据库中的指纹模版匹配，结果失败，服务器可以认为F_u'是假冒用户所发，认证不通过，从而可以保证了用户指纹数据的真实性。

3.2 Stolen-verifier 问题

Stolen-verifier是指攻击者使用从认证服务器中盗窃的用户信息冒充合法用户。假设攻击者已经盗窃了S中A的数据，由于服务器存的指纹和用户信息都是经过双方协商好的密钥K加密的密文，攻击者无法得到F_s或F_u，也就是说只有认证服务器S才能知道F_s和F_u。如果攻击者用盗窃来的$E_s(k_i;(F_s\oplus i)\parallel F_s)$代替A发送到S的信息$E_s(K_n;(F_u\oplus n)\parallel F_u)$，然后将该消息发送到S

中。此时，用户发送来的消息与服务器端数据库保存的用户信息完全一致，即 $F_s = F_u$，但是实际上，用户每次发来的指纹信息和密钥 K 不可能与上次完全一致，故由协议流程可知，服务器决定拒绝该用户的登录请求；另一种情况，如果攻击者盗用的是服务器数据库早先存放的用户指纹数据，并非最近一次认证信息所存的指纹数据。那么，$F_s \neq F_u$，但此时也可以对攻击者控制。由于服务器端设置了用户每次认证通过的都修改的计数器 n，所以服务器计算 $F_u \oplus i \oplus n \neq F_s$，由协议可知，该用户的登录请求被拒绝。这样攻击者利用 Stolen-verifier 问题发动的攻击无法奏效。

3.3　重发攻击

假设攻击者在用户 A 进行第 i 次身份认证时，截获到用户 A 发送到服务器 S 的信息 $E_s(k_i;(F_u \oplus i) \parallel F_u)$。在第 i 次身份认证成功后，服务器 S 的认证计数 n 加 1，i 值随 n 值的同时变化，保存在数据库中的用户 A 的指纹相关信息更新为另一种形式 $E_s(k_{i+1};(F_u \oplus (i+1) \parallel F_s)$。之后，攻击者冒充 A 将截获到的 $E_s(k_i;(F_u \oplus i) \parallel F_u)$ 发送给服务器 S，S 收到后，得到 $F_u \oplus i \oplus n \neq F_u$。由协议可知，服务器 S 拒绝攻击者的登录请求，攻击者没有机会利用重发攻击假冒 A 的身份登录服务系统。所以，协议有效的防范了攻击者的重发攻击。

4　结　论

基于公钥密码体制，提出了一种基于指纹特征作为数字水印的双重网络身份安全认证协议。协议中使用公钥和对称密钥技术，将用户私有信息和指纹特征数据作为数字水印隐藏在载体图像中，将载体图像作为普通数据来进行加密解密处理。协议保证了注册阶段和认证阶段用户指纹信息的真实性和保密性，并对重发攻击和 Stolen-verifier 攻击具有较高的安全性。考虑到客户端的并发量对服务器的压力较大，协议设计过程中将服务器的计算量尽量让客户端分担，从而大大减轻了服务器的负担。

参考文献：

[1] 周琳娜,数字图像盲取证技术研究[D]. 北京:北京邮电大学,2007.

[2] 朱丽娟,须文波,刘渊. 基于指纹的网络身份认证技术的研究与实现[J]. 计算机工程与应用,2003,31:171-173.

[3] 张龙军, 黄继武. 一个安全电子商务身份验证协议[J]. 中山大学学报:自然科学版,2003,42(1): 20-23.

[4] 张喜青. 基于指纹特征的用户身份认证技术研究与开发[D]. 成都:电子科技大学,2002.

[5] 《中国商用密码认证体系结构研究》课题组. 数字证书应用技术指南[M]. 北京:电子工业出版社,2008:21-52.

[6] HE H J, ZHANG J S , CHEN F. Adjacent-block based statistical detection method for self-embedding watermarking techniques[J]. Signal Processing, 2009, 89(3): 1557-1566.

[7] WANG S S, TSAI S L. Automatic image authentication and recovery using fractal code embedding an image inpainting[J]. Pattern Recognition, 2008, 41(2):701-712.

[8] 蔡龙飞,赵慧民. 一种多级数据嵌入的信息隐藏方法研究[J]. 中山大学学报:自然科学版,2013,52(2):23-27.

[9] 陈宏武,赵慧民. 一种基于数字指纹在电子商务中的应用方法研究[J]. 中山大学学报:自然科学版,2009,48(3): 41-46.

[10] YANG H W, SHEN J J. Recover the tampered image based on VQ indexing[J]. Signal Processing, 2010, 90(1):331-343.

[11] NIST. Advanced eneryption standard(AES)[S]. Federal Information Processing Standards Publication, 2007.

[12] 张福泰,孙银霞,张磊,等. 无证书公钥密码体制研究[J]. 软件学报,2011,22(6):1316-1332.