基于 Diffie-Hellman 算法的身份认证模型研究

黄 俊[1],王行恒[2]
(1.公安部第三研究所 信息网络安全公安部重点实验室,上海 201204;
2.华东师范大学 计算中心,上海 200062)

摘 要:随着 Web 应用的安全问题的突出,Web 应用系统的身份认证越来越重要,一种通用的安全的身份认证模型的建立很有必要。提出了一种安全的、可互操作的、实用的身份认证模型,它基于 Diffie-Hellman 密钥交换算法,在认证的过程中用共享密钥加密防窃听,用随机数防重放攻击,用消息认证码防篡改,保证了应用服务器与认证服务器之间的通信安全。文章详细阐述了模型设计细节,通过对其安全性进行分析,认为该模型具有较高安全性,并计划将该模型用于公安 eID 系统中。

关键词:Diffie-Hellman 算法;身份认证;消息认证码;信息安全;信息交换
中图分类号:TP393 **文献标识码**:A **文章编号**:1673-629X(2011)05-0143-04

Diffie-Hellman Algorithm-Based Identity Authentication Model Research

HUANG Jun[1], WANG Xing-heng[2]
(1. Ministry of Public Security Key Laboratory of Information Network Security, Shanghai 201204, China;
2. Computer Center, East China Normal University, Shanghai 200062, China)

Abstract: As the security problems of the Web applications become more prominent, Web application authentication systems are more and more important, and it is necessary to build an available and security identity authentication model. Present a kind of safe, interoperable, practical identity authentication model, which is based on Diffie-Hellman key exchange algorithm. The model uses sharing encryption to prevent eavesdropping, uses random against replay attack, and uses MAC against tampering, which ensure the communication security between the application server and the authentication server. Also expound the details of the model design and analyzes its safety. Finally, believe that the model is of high security, and plan to use it in the eID system.

Key words: Diffie-Hellman algorithm; identity authentication; MAC; information security; information exchange

0 引 言

随着英特网上电子商务以及企业级 Web 系统应用范围不断扩展及应用程度的不断深入,相应的网络安全问题日益突出。其中,身份认证技术是安全技术的一个重要方面,它通过判断某用户是否是他所声称的身份,阻止非法用户使用资源,同时严格地限制不同用户的访问权限。身份认证是安全系统的第一道关卡,一旦身份认证系统被攻破,那么系统的所有安全措施如同虚设。在黑客实施攻击中,其首要目标往往就是身份认证系统,因此完善身份认证体系对维护网络安全起着十分重要的作用,这也体现了构建一个安全的身份认证模型的必要性。

传统身份认证形式可以归纳为如下几种[1]:

(1)简单密码认证形式[1]。这是一种基于用户自己注册用户名和口令的验证方式。每一个合法用户都分配一对用户名和密码,用户访问应用系统时只要用户名和密码匹配正确,则该用户通过身份验证。此种方法在应用通信过程中,都以明码方式传输,因此其缺点很明显,那就是密码易受猜测攻击。但这种身份认证方式在一些安全性要求不高的应用系统中经常使用,原因就是它认证方法简单,使用过程中通过辅以其它措施,以增强其安全性,如将认证方式建立在 ssl/tls 之上,传输以 ssl 加密等[2]。

(2)数字摘要认证形式[1]。数字摘要认证是一种通过 hash 函数把认证的信息加密成数字摘要,并将该摘要传送的认证方认证的方式。它解决了简单密码认

收稿日期:2010-09-10;**修回日期**:2010-12-23
基金项目:国家 863 计划(2008AA01E412);发改办高技([2008]1736 号)
作者简介:黄 俊(1981-),男,江西吉安人,研究实习员,硕士,研究方向为信息网络安全;王行恒,副教授,研究方向为计算机应用技术。

证所造成的基本问题，通过在服务器端把收到的数字摘要同之前得到的数字摘要进行比较，如果比较两者一致则认证通过，否则认证失败。这种方式在认证中比较安全，但是它仅仅是提供身份认证，而并不能保证数据传输过程的完整性和抗抵赖性。

以上两种认证方式存在着各自的缺陷，文中提出的认证模型是建立在基于数字证书的认证基础上之上，遵循 Kerberos 协议，同时需要可信赖的第三方 CA 的支持，并与传输层的 SSL/TSL 协议配合使用[2]。其认证过程可以简单描述为用户、应用服务器和认证服务器，首先从认证中心获得相应证书，用户在请求访问 Web 应用系统时提交个人证书，应用服务器请求认证服务器验证证书的有效性，并将认证结果返回给用户。在当前分布式系统中，这种认证因其安全性较高而被广泛采用，并且通过构造和 CA 之间的信任关系，可以实现跨域之间的认证。认证过程中涉及到密钥交换技术，可采用传统的 Diffie-Hellman 密钥交换算法，同时为了防止中间人攻击在数据交互过程中采用数字签名和双方互相认证方式[3]，以此来提高认证的安全性。

1 Diffie-Hellman 密钥交换算法

Diffie-Hellman 密钥交换算法是 1976 年由 Whitfield Diffie 和 Martin Hellman 提出来的。它的安全性是建立在下述事实上的[4]：求关于素数的模数幂运算相对容易，而计算离散对数却非常困难；对于大素数，求离散对数是不可行的。

Diffie-Hellman 密钥交换算法描述如下[5]：

(1) 首先定义一个素数 p 和本原根 a，这两个数为全局公开量。

(2) 用户 A 的密钥产生。用户 A 选择秘密的 $X_A < q$，并计算公开密钥 $Y_A = a^{X_A} \bmod q$。Y_A 对外公开。

(3) 用户 B 的密钥产生。用户 B 选择秘密的 $X_B < q$，计算公开密钥 $Y_B = a^{X_B} \bmod q$。Y_B 也对外公开。

(4) 用户 A 计算产生共享密钥。$K = (Y_B)^{X_A} \bmod q$。

(5) 用户 B 计算产生共享密钥。$K = (Y_A)^{X_B} \bmod q$。

(6) 因为 $(Y_B)^{X_A} \bmod q = (a^{X_B} \bmod q)^{X_A} \bmod q = (a^{X_B})^{X_A} \bmod q = a^{X_B X_A} \bmod q = (a^{X_A})^{X_B} \bmod q = (a^{X_A} \bmod q)^{X_B} \bmod q = (Y_A)^{X_B} \bmod q$。两者计算的结果相等。从某种意义上说双方已经悄然地交换了一个相同的秘密密钥。

Diffie - Hellman 密钥交换协议可以描述为图 1 所示。假定 A 希望与 B 建立连接，并使用密钥对该次连接中的消息加密。用户 A 产生一次私钥 X_A 计算 Y_A 发送给 B；用户 B 也产生私钥 X_B，计算 Y_B 并将 Y_B 发送给 A，这样 A 和 B 都可以计算出密钥。当然在通信前 A 和 B 都应已知公开的 q 和 a。

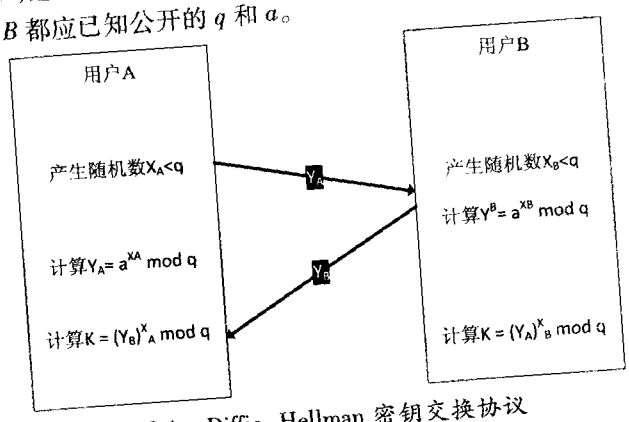

图 1 Diffie-Hellman 密钥交换协议

2 模型设计与安全分析

模型设计是在如下条件前提下完成的，每个用户都有 USB Key，装有数字证书。认证服务端 PKI 基础设施建设完成，能提供 LDAP/OCSP 的目录服务，应用服务器与认证服务器通过 Diffie-Hellman 密钥交换算法共享密钥，并且事先知道生成因子 a 和模数 q，每个应用使用认证服务时，必须先注册应用信息，包括 IP 地址、MAC 地址、域名、验证结果接受 URL 等等，认证服务器通过防火墙对 IP 和 MAC 过滤，确保只有注册过的应用才能访问认证服务的特定端口。下面给出所涉及到的符号及其描述[6]，如表 1 所示。

表 1 符号表

符号	描述
U	用户或用户代理
AS	认证服务器
WAS	应用服务器
K	密钥
C(K,M)	消息 M 的认证码，密钥为 K
E(K,X)	用密钥 K 和对称算法加密明文 X
D(K,Y)	用密钥 Y 和对称算法解密密文
x||y	级联 x 和 y
q	Diffie-Hellman 算法的模
a	Diffie-Hellman 算法的生成因子即 q 的原根
ID_X	X 标识
mode	认证请求类型，包括请求建立关联和请求认证两种类型
Expires_in	会话关联的有效时长
ADx	X 的 IP 地址
result	认证结果，包括肯定断言与否定断言
nonce	随机数
Association-handle	会话关联句柄
MacType	Mac 类型，用于生成 mac

本模型完整的身份认证过程如图 2 所示。

其步骤为[7]：

(1) U → WAS: IDu||IDwas 。用户插入 usb

通过浏览器访问应用服务器,应用服务器查看本地数据库是否有与认证服务器的会话关联,会话关联是以认证服务器的 IP 地址为 key 存储在本地数据库,它包括此关联的 macKey、关联生成时间,关联的有效时间和认证服务器的 IP 地址,也称该地址为关联句柄 Association_handle。应用服务器定时维护数据库,检查关联是否过期或更新,如果没有应用服务器则请求建立关联。

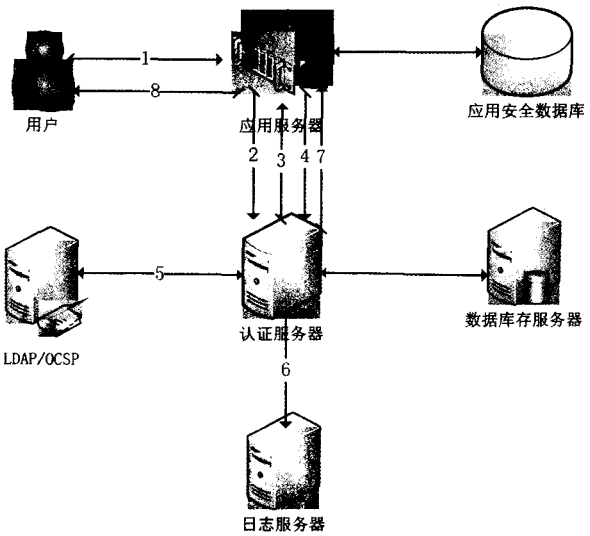

图 2　身份认证模型

(2) WAS → AS: MacType || DH-was-Public || mode。应用服务器向认证服务器以 https 形式发送建立关联的请求,关联请求的消息中包括产生消息认证码的类型和应用服务器 Diffie-Hellman 算法公钥以及请求的目的 mode,mode 有两种建立关联请求和认证请求。

(3) AS → WAS: E(k, MacKey) || Association_handle || Time || Expires_in || DH-Server-Public。其中 MacKey = G(MacType, size)。认证服务器根据 mac 类型,生成 *mac*,并用共享密钥加密,同时附上将关联的产生时间、有效时间、关联句柄和认证服务器 Diffie-Hellman 算法公钥。认证服务器将该关联的相关参数存入本地数据库中,同时将关联参数发送给应用服务器,应用服务器用关联中的 DH-Server-Public 和本地的私钥由 Diffie-Hellman 算法得到共享密钥 K,通过 D(k, E(MacKey)) 解密得到 mac,应用服务器将由 macKey、Association_handle、time、Expires_in 组成的关联存储在本地数据库。这样应用服务器和认证服务器间的关联就建立完成。

(4) WAS → AS: mode || E(k, IDu || IDwas || ADwas || Association-handle || C(k, IDu || IDwas || ADwas || Association-handle))。在认证的阶段应用服务器和认证服务器间的通信采用消息认证码和加密的方式以 https 传输,如图 3 所示[8]。应用服务器将认证请求发送给认证服务器。发送请求中包括明文传送的认证请求标识 mode,以及用共享密钥加密的用户标识(用户的证书信息)、应用服务器的标识、应用服务器的 IP 地址、关联句柄和相应的消息认证码等信息。

(5) AS → LDAP/OCSP: IDu。认证服务器收到认证请求,先用共享密钥解密消息,再用关联的 macKey 求解源消息的消息认证码,通过与解密的消息认证码比较,如果不一致则放弃此次认证。比较通过后,再看源消息中的应用服务器的 IP 地址 ADwas 与此时发送请求的应用服务器 IP 地址是否一致,不一致则放弃认证。都通过之后,提取证书信息,从缓存中查找该证书有效性,如未找到则查询 LDAP/OCSP,认证其有效性。

(6) AS → LS: LOGS。无论认证结果是成功还是失败,认证服务器请求日志服务器记录日志。

(7) AS → WAS: mode || E(k, IDu || IDwas || ADwas || Association-handle || result || *nonce* || C(k, IDu || IDwas || ADwas || Association-handle || ADas || result || nonce))。认证服务器将认证结果返回给应用服务器,传送方式与第 4 步一样,参数中增加了认证服务器的地址 ADas,验证结果 result(包括肯定断言和否定断言)和随机数 nonce。

(8) WAS → U: result。应用服务器收到认证服务器发来的响应,先用共享密钥解密,再比较消息认证码,都通过之后开始验证 nonce,如果 nonce 有重复,则认为该用户认证不通过。验证 IDwas,看是否与本应用服务器的 IDwas 一致,不一致则同样认为该用户认证不通过。验证 ADas,查看是否与请求认证的服务器地址一样,不一样则认证不通过。验证 result,如果 result 是肯定断言,则用户身份认证通过。如果 result 是否定断言,则用户身份认证不通过,将结果返回给用户。

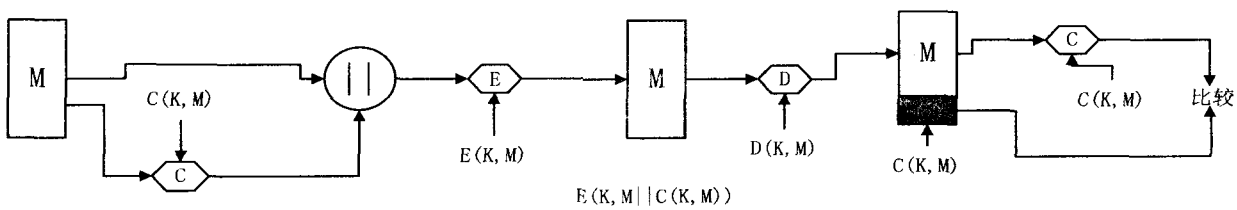

图 3　消息认证码

上述步骤描述了一个认证的基本过程，并使认证具有保密性、完整性、不可否认性、可控性、可审查性及可用性。

(1)保密性。用户在访问应用后，应用服务器组成的认证请求是通过对称加密算法(共享密钥)加密后发给认证服务，只有认证服务器能解密。认证服务把认证结果重定向给应用时，也是通过对称加密算法加密的，只有应用服务器才能解密。应用和认证服务的共享密钥通过安全管理、定期更新与其它保护措施确保不被泄漏。

(2)完整性。用户在访问应用后，应用服务器组成的认证请求和认证服务器返回给应用的认证结果都以消息认证码的形式来确保数据的完整性。

(3)不可否认性。应用服务器和认证服务器之间每次交互，通过消息认证码(数字签名)保障不可否认性。用户和应用之间交互，以用户的 IP 地址来保障用户访问的不可否认性。

(4)可控性。每个应用使用认证服务时，必须先注册应用信息，包括 IP 地址、MAC 地址、域名、验证结果接受 URL 等等。同时将应用证书和应用信息存入认证服务。通过严格执行以上管理措施，达到认证服务对应用认证请求接受来源和回复信息目的全面可控。即使应用或者认证服务 IP 地址被冒用，DNS 被欺骗，但应用和认证服务交互的数据都以各自证书作了加密。盗用者也无法利用这些数据。

(5)可审查性。认证服务器上有专门的日志服务器记录应用每次向认证服务的认证请求和处理结果，所有记录结果可供将来审计用。

(6)可用性。通过本模型的认证方式能保证拥有合法证书的用户，可通过认证服务器的认证，登入已注册的应用。

以下主要是从攻击者角度对上述认证模型做出安全性分析[9,10]：

(1)中间人攻击。中间人攻击易发生于应用服务器与认证服务器建立关联的时候，由于密钥交换算法的 a 和 q 在应用服务器和认证服务器间都是事先知道的，第三方截取到公钥时算不出共享密钥，故也就不能解密认证服务器生成的 macKey。

(2)网络钓鱼。认证过程中都会验证对方的地址，可以对网络钓鱼进行预防。

(3)网络窃听。认证过程中，所有消息都是经过加密处理，密钥只有应用服务器和认证服务器知道，因此窃听不大可能发生，除非密钥泄露。

(4)重放攻击。认证服务器在反加认证结果中，附加一个随机数，该随机数的生成是唯一的，应用服务器可以验证随机数来预防重复攻击。

(5)消息篡改。认证过程中，所有消息都附上了消息认证码，对源消息更改在比较认证码时会导致验证不通过。

(6)拒绝服务攻击。应用服务器和认证服务器间建立安全通道，认证服务器只对注册过的应用服务器开放端口，而且通过验证地址是否一致来进一步阻止拒绝服务攻击。

3 结束语

如今的 Web 应用系统一方面要满足用户的易操作性、跨平台性和分布式要求，另一方面也要为应对各种各样的安全威胁而做大量的工作。这其中重中之重就是把好身份认证关，身份认证做好了，各种用户的使用资源的权限也就相对明朗了，也即各种潜在威胁都在掌控之中[11]。因此可以说身份认证是 Web 服务安全中的核心问题之一[12]。文中提出了一种安全的、可互操作的、实用的身份认证模型解决方案，它具有灵活性、可扩展性强，易维护等特点，同时保证了认证过程的保密性、完整性、不可否认性、可控性、可审查性，能够解决典型 Web 服务应用模式下的通信安全问题。今后工作的重点将放在将模型应用于公安 eID 身份认证系统中，由于 eID 系统要求每秒认证人数达到百万级，因此模型的设计除了能保证其安全性外，是否会带来性能上的障碍还需在应用实践中有待验证。

参考文献：

[1] 沈海波. Web 服务中的关键安全技术研究[D]. 武汉：华中科技大学，2007.

[2] Tsalgatidou A, Pilioura T. An Overivew of Standards and Related Technology in Web Services[J]. Distributed and Parallel Databases, 2002, 12(2): 135-162.

[3] 岳 昆，王小玲，周傲英. Web 服务核心支撑技术[J]. 软件学报，2004，15(3)：428-442.

[4] Bhatti E, Shafiq B, Bertino E, et al. X-GTRBAC Admin: A Decentralized Administration Model for Enterprise-Wide Access Control[J]. ACM Transactions on Information and System Security(TISSEC), 2005, 8(4): 388-423.

[5] Stallings W. 密码编码学与网络安全[M]. 王丽娜，傅建明，译. 北京：电子工业出版社，2007.

[6] 丁振国，陈陆艳. 一种安全的身份认证模型的研究与实现[J]. 航空计算技术，2010，40(1)：131-134.

[7] 韩 涛，郭荷清. Web 服务安全模型研究与实现[J]. 计算机工程，2006，33(24)：130-132.

[8] Stallings W. Crytography and Network Security Principles and Practies[M]. Beijing: Publishing House of Electronics Industry, 2006.

[9] 张明西. OpenID 多级安全问题研究[D]. 上海：东华大学，

(下转第 154 页)

加密后的。DES 为对称加密算法，Hash 为非对称加密算法。对称加密算法 AES 比 DES 具有更好的保密性，但是运算量也更大。如果希望进一步提高安全性，也可以使用读写 tag。在第 3 步中，reader 随机生成一个新的 $ID^{'}$，并计算 $D(ID^{'} \| K)$ 发送给 tag。第 4 步 tag 通过认证后，将 ID 同步更新为 $ID^{'}$。

Hybrid-Encryption 模型的优势主要在于传送的信息是通过对称加密算法进行加密的，这样可以增强其安全性，减小被破译的风险。由于应用了非对称加密思想，可以适应物联网应用环境的开放性。

Hybrid-Encryption 模型的处理算法复杂度分析如下：首先是 tag 方，tag 收到 $D(R \| K)$ 后解密，DES 算法是一个分组加密算法，它以 64 位为分组对数据加密，一共进行 16 轮完全相同的运算。DES 算法加密解密的时间复杂度均为 $O(n)$，空间复杂度也为 $O(n)$。接着 tag 要计算 Hash($R \| ID$)，Hash 算法中的 Hash 函数是一个单向函数，因此 Hash 算法的复杂度就取决与该函数的复杂度，而这个函数可以是不固定的，例如可以利用 DES 来构造 Hash 函数，也可以利用 IDEA 来构造 Hash 函数，所以 Hash 算法的复杂度需要具体情况具体分析。认证过程的最后，tag 还会收到 $D(ID_j \| K)$，解密仍然是 DES 算法。所以 tag 的时间复杂度是：$O(n)$ +Hash 函数的时间复杂度，空间复杂度是：$O(n)$ +Hash 函数的空间复杂度。现在再考虑 reader 的复杂度。Reader 首先用 DES 加密随机数，在收到 tag 返回的 Hash 值后也要计算 Hash 并和返回的 Hash 匹配，之后同样是用 DES 对 ID_j 进行加密。Hash 的匹配一般是通过 Hashtable 进行，其复杂度为 $O(\log 2n)$。因此，reader 的复杂度为：$O(n)$ +Hash 函数的时间复杂度+ $O(\log 2n)$。通过分析，可以看到，该算法在功能上有较好的改进，但是对性能的影响却是有限的。不过由于 tag 上要同时集成 DES 算法和 Hash 算法，所以 tag 的制造成本会有所提高。

4 结束语

随着 RFID 系统的应用范围不断扩大，RFID 系统的安全性和保密性显得愈加重要。尽管在 RFID 的安全问题上已经做了不少研究，RFID 系统仍然具有各种各样的安全问题。文中分析了对 RFID 的安全问题，在研讨目前主要安全协议的基础上，给出了一种混合方式的安全模型，并对该模型的有效性和算法复杂度做出了说明。随着科技的不断进步，相信在不久的将来，RFID 的安全问题将得到进一步的提高。

参考文献：

[1] 王昭顺，张晓锋. RFID 安全隐患及其解决方案[C]// 2008 国际 RFID 技术高峰论坛论文集. 北京：出版者不详，2008：107-111.

[2] 杨海东，杨 春. RFID 安全问题研究[J]. 微计算机信息，2008(8)：238-240.

[3] 郎为民，刘德敏，李建军. RFID 安全机制研究[J]. 技术前沿，2007(9)：55-58.

[4] 落红卫，程 伟. RFID 安全威胁和防护措施[J]. 电信网技术，2010(4)：38-40.

[5] 周永彬，冯登国. RFID 安全协议的设计与分析[J]. 计算机学报，2006(4)：581-589.

[6] 李 莉，刘建伟. RFID 安全保密技术研究进展[J]. 信息安全与通信保密，2007(8)：165-167.

[7] 彭 昭，刘 威，马选斌，等. RFID 系统安全与隐私问题研究[J]. 微电子学与计算机，2007(8)：129-131.

[8] 祝胜林，杨 波，张明武. RFID 协议及其安全性研究[J]. 信息安全与通信保密，2007(8)：168-170.

[9] Weis S A. Security and Privacy in Radio-Frequency Identification Devices[D]. Massachusetts: Massachusetts Institute of Technology, 2003.

[10] 欧阳麒，蒋兴浩，孙锬锋. 一种基于相互认证的安全 RFID 系统[J]. 信息安全与通信保密，2006(12)：142-144.

[11] Weis S A, Sarma S E, Rivest R L, et al. Security and privacy aspects of low-cost frequency identification systems[C]// Proceedings of the 1st International Conference on Security in Pervasive Computing. Berlin: [s. n.], 2004: 201-212.

[12] Lee S M, Hwang Y J, Lee D H, et al. Efficient authentication for low-cost RFID systems[C]//Proceedings of the International Conference on Computational Science and Its Applications. Berlin: [s. n.], 2005: 619-627.

(上接第 146 页)

2010.

[10] 刘 敏，吕先竞，宋玉忠. 基于 OpenID 的分布式认证系统的设计与实现[J]. 现代情报，2008(6)：90-92.

[11] 张志明，徐建桥，严承华. 基于图像数字水印的身份认证研究[J]. 计算机与数字工程，2006，38(3)：95-97.

[12] 许 峰，林果园，黄 皓. Web Service 的访问控制研究综述[J]. 计算科学，2005，32(2)：1-4.

(上接第 150 页)

程与应用，2010(3)：4-8.

[11] Bayer B. An optimum method for two-level rendition of continuous-tone pictures[C]//IEEE International Conference on Communications. [s. l.]: IEEE, 1973: 11-15.

[12] 廖小涛，张晓林，刘荣科. SPIHT 算法容错性能的分析及改进[J]. 遥测遥控，2003(5)：44-48.