认证字典及其在 PKI 中的应用研究

卿斯汉[1,2], 周永彬[1,2], 张振峰[1,3], 刘 娟[4]
(1. 中国科学院软件研究所, 北京 100080; 2. 中国科学院信息安全技术工程研究中心, 北京 100080;
3. 中国科学院软件研究所信息安全国家重点实验室, 北京 100080; 4. 云南民族大学历史系, 云南昆明 650031)

摘 要: 认证字典是一类重要的数据结构, 它在众多研究领域都具有重要的理论和应用价值. 文中介绍了认证字典的基本概念和原理, 引入了时间约束, 给出了一种新的认证字典分类方法. 从认证字典在 PKI 中的应用出发, 分析了其实现技术; 并简单地分析和比较了基于不同认证字典实现的证书撤销方案.

关键词: 认证字典; 证书撤销; 公钥基础设施; 信息安全
中图分类号: TP309.08 文献标识码: A 文章编号: 0372-2112 (2004) 08-1356-04

Study on Authenticated Dictionary and Its Applications in PKI

QING Si-han[1,2], ZHOU Yong-bin[1,2], ZHANG Zhen-feng[1,3], LIU Juan[4]
(1. Institute of Software, Chinese Academy of Sciences, Beijing 100080, China;
2. Engineering Research Center for Information Security Technology, CAS, Beijing 100080, China;
3. State Key Laboratory of Information Security, Institute of Software, CAS, Beijing 100080, China;
4. Dept. of History, Yunan Nationalities University, Kunming, Yunnan 650031, China)

Abstract: Authenticated Dictionary (AD) is one of the important data structures, and it is of great theoretic and applicable value in many research fields. Basic concept and principle of AD were examined. Time constraint was introduced into the model of AD, based on which a new taxonomy of AD was presented. Afterwards, different implementation technologies of AD were given on the base of typical application of ADs in PKI fields. A brief performance analysis and comparison among these certificate revocation schemes based on different ADs were carried out.

Key words: authenticated dictionary; certificate revocation; PKI; information security

1 引言

认证字典 (Authenticated dictionary, 简记为 AD) 是由 Naor 和 Nissim 提出的一类重要的数据结构[1], 它在众多研究领域都具有重要的理论和应用价值, 诸如科学数据挖掘、地理数据服务器、Internet 上的第三方数据发布以及 PKI 中的证书撤销等[2,3].

认证字典所要解决的基本问题描述如下: 设计一个协议, 用于在一个不可信的示证者 P 和一个验证者 V 之间来对集合 S 进行隶属关系查询. 这里, S 是由某个可信实体定义的一个有限集合, 该集合对于 V 是未知的. 信息源控制着用 S 来表示的认证字典所必需的信息. 给定一个输入 x, P 可以证明 $x \in S$ 或者 $x \notin S$; 同时, 可信实体可以动态地改变集合 S 的元素. 对于 AD, 一般有如下两个假设: (1) P 与 V 交互作用时, S 固定不变; (2) 信息源是可信的. 对于不可信的信息源, 认证字典所要处理的实际情况将更复杂, 本文暂不讨论.

从参与模型的各主体与信息流的角度出发, 认证字典的基本模型如图 1 所示. 在图 1 所示的模型中, 信息源就是问题描述中所指的可信实体, 目录则为示证者 P, 而用户就是验证者 V. 需要特别说明的是: 示证者 P 可以是一个 (也通常是一个) 非可信的实体; 这一重要特性大大简化了 AD 的设计和实施.

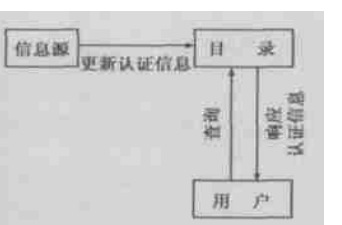

图 1 认证字典基本模型

在 PKI 中, 信息源即 CA (Certification authority) 中心, 它向实体签发数字证书, 用以证明实体公钥和其身份信息及其他信息之间的绑定, 从而来保证这些信息的真实性和有效性. 每一个证书都有一定的有效期, 有效期过后该证书便不再有效. 然而, 可能会因为某种原因 (例如, 私钥的泄漏或者丢失等) 需要在证书的有效期到达之前撤销证书. 因此, 证书的使用者就需要一种手段来查询给定的证书是否已被撤销. 证书撤销和证书状态验证服务是 PKI 中一个极为重要的基础性问题, 认证字典则是实现这类服务的天然数据结构.

2 认证字典的定义

记 U 为一全集, S 为其子集, 设 D_S 为表示集合 s 的一个数据结构. 定义如下操作:

隶属关系查询 其形式为 3 e4; 对该查询的响应是一个

收稿日期: 2003-09-13; 修回日期: 2004-03-02
基金项目: 国家自然科学基金 (No. 60373039/60083007); 国家重点基础研究发展规划 (No. G1999035802)

字符串3 a4, 满足 a I {YES, NO}, 分别对应于 e I S 和 e| S.

认证的隶属关系查询　其形式为3 e4; 对该查询的响应是一个字符串3 a, p4, 满足 a I {YES, NO}, p 是由示证者 P 给出的用于证明a 的真实性的一个证据.

更新操作　其形式有如下两种:

¹ 插入操作3 insert, e4, 这里 e I (U\ S). 设对 D_S 进行插入操作后所生成的新的数据结构为 D_{Sc}, 则 Sc= SG{e}.

º 删除操作3 delete, e4, 这里 e I S. 设对 D_S 进行删除操作后所生成的新的数据结构为 D_{Sc}, 则 Sc= S \ {e}.

基于上述直观描述, 下面给出认证字典的定义.

211　字典定义

/ 字典0 是一个表示支持隶属关系查询和更新操作的集合 S 的数据结构 D_S, 以及与所有这些操作相关的各种协议之总称.

212　认证字典定义

/ 认证字典0 是一个表示支持认证的隶属关系查询和更新操作的集合 S 的数据结构 $D_S^{\Delta U}$, 以及与所有这些操作相关的各种协议之总称.

认证字典是动态的, 它需要有一种机制来证明某个证明已经是更新过了的. 否则, 非可信的目录就可能重放旧证明. 这里假定: 要么由信息源按照预定的时间或策略来对 AD 进行更新, 要么发起查询请求的用户知道最新的更新是何时发生的. 但是, 无论在上述哪种情况下, 验证者 V 都应该、也必须能检验证明 p 的新鲜度.

根据认证字典的定义及其基本模型, 可以得出在设计认证字典时应该考虑达到如下目标:

计算量小　认证字典模型内的任何一个实体内部所要完成的计算都应尽可能简单、快速; 同样, 支持这些计算的数据结构所必需的内存空间应尽可能小.

通信载荷小　信息源到目录(更新认证信息) 之间的通信量以及目录到用户(查询认证信息) 之间的通信量应尽可能小.

安全性高　用户能够以很高的可靠性来验证目录所提供的数据的真实性.

时间约束　如果可能, 目录能为用户提供在某个特定时刻(t 时) 的信息真实性证明.

也就是说: 构造认证字典时, 在保证安全性的前提下, 要使得计算和验证 P 应尽可能高效, P 要尽可能短; 维护认证字典的操作所涉及的计算量和通信量应尽可能小; 如果可能, 应能够提供具有时间约束的认证查询.

3　认证字典的分类页

依据 AD 所支持查询种类与时间约束的不同, 本文将其分为三大类:

瞬时型认证字典(Ephemeral authenticated dictionary, 简记为 EAD)

持续型认证字典(Persistent authenticated dictionary, 简记为 PAD)

扩展型认证字典(eXtended authenticated dictionary, 简记为 XAD)

这三种不同的认证字典有不尽相同的应用场合, 下面逐一介绍.

311　瞬时型认证字典(EAD)

一般所说的认证字典就是指 EAD, 其基本形式为: / (当前) , 元素 e(不) 在集合 S 中吗?0. EAD 通常无需提及时间约束, 问题隐含着查询是针对当前时间而言的. 不难看出: 这类认证字典一般用于对操作数据的当前状态而言.

312　持续型认证字典(PAD)

PAD 是 EAD 在功能上的增强型, 其基本形式为: / 在 t 时刻, 元素 e(不) 在集合 S 中吗?0. PAD 可以对过去某个时刻数据的真实性进行证明. 显然, 当 t= 0(表示当前时刻), 有 $PAD_{(t=0)}$= EAD. 不难看出: PAD 比 EAD 功能增强的一个主要方面就是它可以很方便地支持/ 历史状态查询0, 以便于查询给定信息在过去某一时刻的真实性. 显然, EAD 无法满足这种需求.

313　扩展型认证字典(XAD)

XAD 包含了 EAD 和 PAD 的所有功能, 但不是二者的简单叠加; 即有(EAD G PAD) < XAD, 但(EAD G PAD) X XAD. XAD 的基本模型(如图 2 所示) 也和上述两种类型的认证字典的基本模型不同. 这里, 信息源不再是一个单一的、固定的信息源, 而是由多个信息源(设其数目为 n) 组成; 这些信息源共享同一个目录进行信息发布. 任意的两个信息源之间相互独立, 用户可以向目录进行认证的隶属关系查询.

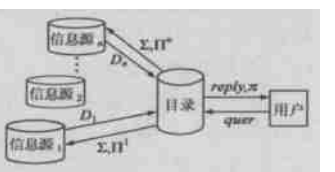

图 2　扩展型认证字典(XAD)

对于 XAD 而言, 问题的基本形式没有多大变化. 但是, 信息的发布和查询过程却与上述两种类型的 AD 有很大的不同, 具体如下:

信息发布:　设信息源为 O_i(1F iF n) 欲向目录 P 发布的数据子集为 D_i(1F iF n) . P 在某一个特定时刻收集所有来自各信息源 O_i 的数据子集 D_i, P 首先对这些子集的和集 D= $G_{i=1}^{n} D_i$ 计算摘要 E; 对于每一个 D_i, 都有一个唯一的证明 F^i, 用以证明 D_i I E. P 将 F^i 和 E 发送给每一个信息源 O_i, 同时 P 将 E 公开. 每一个信息源 O_i 都可以验证证明 F^i.

认证查询:　用户 User 向 P 发送查询 query. P 计算对应于查询 query 的响应 reply 以及其证明 P, 并将二者发送给用户 User. 用户检查 reply 和 P 的正确性, 并据此决定接受或拒绝响应 reply.

一般来说, 在基本的 XAD 中, 认证查询仍然是针对当前时刻而言的. 这种类型的 AD 在广义的基于 Internet 的真实信息发布中具有重要应用. 显然, 对 XAD 加上时间约束可将其扩展为更一般的基于时间约束的 XAD.

4　认证字典实现技术

实现认证字典有很多方法, 但是它们却几乎全部基于密

码学理论[4]. 实现 AD 所需要的基本密码学结构主要有数字签名、单向无碰撞杂凑函数、链接式杂凑认证、树形杂凑认证以及一些特殊的密码学结构, 例如单向累加器(One2way accu2mulator, 简记为 OWA) 等[4~7].

假设全集为 U, 欲为数据集合 S 构造一个认证字典 D_S^{AU}. 对于一个小的全集 U 而言, 构造 D_S^{AU} 的计算量可以和| U| 成正比. 在实现 AD 的过程中, 通常要用到一个单向无碰撞杂凑函数 h().

单向无碰撞杂凑函数 h() 是这样定义的: 在其定义域内, 找到一个 y X x 满足 h(x) = h(y) 在计算上是不可行的[4].

4l1 线性列表方法

如果 S 比较小, 信息源可以对包含所有 s I S 的线性列表的一个消息 M 进行签名. PKI 中最常用的证书撤销列表)))) CRL[3] 就是采用了这种方法.

这是最简单的认证字典构造方法. 用户进行查询时, 只需要向目录提出查询请求, 询问 x_i 是否在 S 中. 目录将这个完整的已签名过的列表发送给用户. 用户首先验证该签名列表的有效性; 然后, 通过查询该线性列表就可以确定待查询元素 x_i 是否在 S 中. 信息源更新 AD 时, 只需向该线性列表中添加元素, 或者从该线性列表中删除元素, 然后对更新后的线性列表重新签名即可.

4l2 单元素状态签名法

对于每一个元素 e I U, 信息源都对表示其相关状态 e I U 或者 e| U 的消息进行签名. 如果要更新 D_S, 就必须提供| U| 个签名, 其与 D_S 中变化元素的数目无关. CRS(证书撤销系统) 就是采用这种实现方法[8], 其基本思想是: 对于| U| 中的每一个元素 x_i, 信息源随机选取两个长度至少为 100 比特的数 X_0 和 Y_0, 计算两个参量 X 和 Y, 这里 $X = h(X_0)$, $Y = h^{365}(Y_0)$, h 是上面所描述的一个公开的高强度杂凑函数. 信息源将 X 和 Y 公开, 而将 X_0 和 Y_0 保密. 信息源在第 j 天(1 F j F 365) 定期向目录更新认证信息. 如果 x_i I S, 则信息源公开 $Y_i = h^{365-i}(Y_0)$; 如果 x_i| S, 则信息源公开 $X = h(X_0)$.

由上述描述可知: 基于这种技术实现的 AD 中, 信息源和目录之间的通信开销相当大. 为了减小信息源和目录之间的通信开销, William Aiello 等改进了 CRS, 提出了使用二叉树来作为基本数据结构的 HCRS(层次型 CRS)[9].

4l3 基于杂凑树的实现方法

该方案中, 信息源对/ 不在 S 内的元素的有序间隔0 进行签名, 并使用 Merkle 杂凑树作为基本的数据结构来组织数据. 有序间隔是一个对(s_1, s_2), 满足对所有的 s_1 F s F s_2, s| S. 例如, CRT(证书撤销树) / Skip2List/ 223 树等都是基于这种实现技术[1, 10, 11]. 下面给出一个基于杂凑树来构造 AD 的通用方法. 该构造方法中隶属关系查询和更新操作的计算量大致是 log| D_S| 的一个因子.

设 D_S 是一个表示集合 S 的字典, 其大小为| D_S| . 设 T_q 和 T_u 分别表示进行一个隶属关系查询操作和一个更新操作所需要的最坏时间. 设 h 是一个单向无碰撞杂凑函数, T_h 是对 U 的实例计算 h 所需要的时间. 考虑 S 的一个表示 D_S= {D_1, D_2, , }, 它可能是组成 D_S 的所有变量值的一个列表. 认证字典 D_S^{AU} 包含有 D_S 和一个杂凑树[4], 该杂凑数的节点代表着{D_1, D_2, , }; 由信息源签名的一个消息则包含有该树的根节点值及更新时间.

杂凑树的构造方法如下: 创建一个平衡二叉树, 其叶子节点的值分别为 D_1, D_2, , . 每一个内部节点 v 赋值为 h(x_1, x_2), 这里 x_1 和 x_2 分别表示赋给节点 v 的两个子节点的值. 最后, 信息源对该杂凑树的根节点信息以及更新时间进行签名.

#通过给 D_S 中的每一个元素 D_i 提供一个在计算过程中可以访问的证明, 就可以将一个隶属关系查询变换为一个认证的隶属关系查询. 该证明包含有位于从根节点到位置 i 的路径上所有节点的兄弟节点的值. 认证的隶属关系查询的复杂性为 O(T_q# T_h# log| D_S|).

#对认证字典进行更新操作后, 发生改变的与元素 D_i 相对应的杂凑树部分(也就是说, 自变化的元素 D_i 至根节点的所有路径) 就需要重新计算. 因此, 更新操作的复杂性为 O(T_u# T_h# log| D_S|).

4l4 基于 OWA 的实现方法

单向累加器(OWA) 是可用于实现 AD 的一类重要的密码结构. Goodrich 等给出了一种基于大整数环内模幂运算的 OWA[8]. 其基本原理如下: 信息源随机选择两个大素数 p 和 q(保密), 计算 N= pq. 同时, 信息源选择一个相当大的基 a, 满足 a 和N 互素, 即(a, N) = 1; 信息源公开 a 和N. 设 S= {x_1, x_2, , , x_n}, 其中 x_i 为素数, 信息源计算 $A = a^{x_1 x_2, \ldots x_n} \bmod N$; 又设 t 为信息源发布 D_S 的时间, 信息源对(A, t) 进行签名, 其签名为 sign; 则 D_S^{AU}= {S, (A, t), sign}.

进行查询操作时, 用户向目录询问 x_i 是否在 S 中. 目录计算 $A_i = a^{x_1 x_2, \ldots x_{i-1} x_{i+1}, \ldots x_n} \bmod N$, 并将{$A_i$, (A, t) , sign} 发送给用户. 用户首先验证 sign 是否为信息源对(A, t) 的签名, 然后计算 $Ac = A_i^{x_i} \bmod N$. 如果 Ac = A, 则必然有 x_i I S; 否则, x_i| S.

设要向集合 S 中插入新元素 x_{n+1}, 信息源只需重新计算 $A_{new} = A^{x_{n+1}} \bmod N$, 对(A_{new}, t_{new}) 重新计算签名 $sign_{new}$; 令 $S_{new} = SG\{x_{n+1}\}$, 信息源重新发布 $D_{S(new)}^{AU}$= {S_{new}, (A_{new}, t_{new}), $sign_{new}$} 即可. 相比而言, 删除操作则比较费时, 需要信息源重新计算 n 次大整数模幂运算. 设从集合 S 中删除元素 x_i, 则信息源需重新计算, $A_{new} = a^{x_1 x_2, \ldots x_{i-1} x_{i+1}, \ldots x_n} \bmod N$, 对(A_{new}, t_{new}) 重新计算签名 $sign_{new}$; 令 S_{new}= S \ {x_i}, 信息源重新发布 $D_{S(new)}^{AU}$= {S_{new}, (A_{new}, t_{new}) , $sign_{new}$} 即可.

由上面的描述可以看出: 基于这类 OWA 所构造出的 AD 具有先天的计算不对称性. 进行查询的用户仅需进行一次大整数模幂运算和一次数字签名验证运算; 而信息源则需要更多的计算.

4l5 小结

由上面的讨论可以看出: 无论实现认证字典基于何种数据结构, 也不管采用何种实现技术, 信息源都必须对所有的消息进行数字签名, 而且在被签名的消息中必须包含有认证信

息更新时间.

5　性能分析和比较

证书撤销是 PKI 中的一个十分具有挑战性的基础性问题.为此,研究人员提出了多种证书撤销方案[12];但从本质上讲,它们都是认证字典在 PKI 中的典型应用.下面将对当前 PKI 领域中几种主要的证书撤销方案作一下简单的性能分析和比较.本部分用到的基本记号如下: n 表示认证字典中元素的数目,即 n= | S| ; N 表示全集中元素的数目,即 N= | U| ; t 表示自创建一个新的待查询元素以来的更新次数.

为了对各种证书撤销方案进行性能分析和比较,首先就前面提出的认证字典的设计目标来简单介绍一下与认证字典性能有关的两个基本参数:

(1) 计算复杂性　与该参数有关的因素主要有三个,即构造一个认证字典所需要的时间和空间、完成一个认证的隶属关系查询所需要的时间和验证对一个认证的隶属关系查询的响应所需要的时间.

(2) 通信复杂性　与该参数相关的因素主要有两个,即更新认证字典所需要的通信量和认证的隶属关系查询证明 p 的长度.

因此,本文使用上面给出的评价参数来分析四种不同类型的证书撤销系统的性能和效率,它们分别是: CRL、CRS、CRT 和基于 OWA 的证书撤销方案.这些方案中,集合中的元素就是证书的序列号,而全集 U 则为某个 CA 签发的所有证书.由第 4 部分的描述容易得到其比较结果,如表 1 所示.欲了解详细的信息,请参见文献[12].

表 1　基于不同实现技术的认证字典的性能分析和比较

项目 / 方案	空间开销	更新时间	更新信息	查询时间	查询信息	验证时间	证明 p 的长度
CRL	O(n)	O(n)	O(n)	O(n)	O(n)	O(n)	O(n)
CRS	O(N)	O(N)	O(N)	O(N)	O(1)	O(t)	O(1)
CRT	O(n)	O(n)	O(1)	O(logn)	O(logn)	O(logn)	O(logn)
OWA	O(n)	O(n)	O(1)	O(n)	O(1)	O(1)	O(1)

由上面的比较分析可以看出:证书撤销方案依赖的认证字典所使用的数据结构不同,认证字典进行基本操作的难易程度以及操作的效率也会有很大的不同.

6　结论

认证字典是一类重要的数据结构,它在众多研究领域都具有重要的理论和应用价值,诸如科学数据挖掘、地理数据服务器、Internet 上的第三方数据发布以及 PKI 中的证书撤销等.介绍了认证字典的基本概念与原理,引入了时间约束,并据此给出了认证字典的一种新的分类方法,进而探讨了相关的实现技术.

认证字典在信息安全领域具有重要的应用价值,诸如 PKI 中的证书撤销和状态验证机制.文中对基于不同的密码技术实现的认证字典进行了简单的性能分析和比较.本文的研究表明:在应用这一重要的数据结构时,除了要使得构造出的认证字典具有安全、高效、简洁等特性,还必须要考虑到 AD 的具体应用背景因素,进行综合的分析比较.

参考文献:

[1] M Naor, K Nissim. Certificate revocation and certificate update[A]. Proceedings of the 7th USENIX Security Symposium (SECURITY298) [C]. San Antonio: IEEE Computer Society, 1998. 217- 228.

[2] P Devanbu, M Gertz, et al. Authentic third2party data publication[A]. Bhavani M Thuraisingham, et al. Fourteenth IFIP 11.3 Conference on Database Security[C]. Netherlands: Kluwer, 2000. 101- 112.

[3] R Housley, W Ford, et al. IETF RFC 2459, Internet X.509 public key infrastructure: certificate and CRL profile[S]. Jan. 1999.

[4] W Stallings. Cryptography and network security: principles and practice (Second Edition) [M]. New Jersey: Prentice Hall, 1998.

[5] R C Merkle. A certified digital signature[A]. Proceedings of Crypto. 89[C]. Germany: Springer2Verlag, 1989. 234- 246.

[6] J Benaloh, M de Mare. One2way accumulator: a decentralized alternative to digital signatures[A]. Advances in Cryptology2EuroCrypto93 [C]. Germany: Springer2Verlag, 1993. 274- 285.

[7] M T Goodrich, R Tamassia. An efficient dynamic and distributed cryptographic accumulator[A]. Proceedings of Information Security Conference (ISC 2002) [C]. Germany: Springer2Verlag, 2002. 372- 388.

[8] S Micali. Efficient certificate revocation[R]. Technical Memo MIT/LCS/TM2542b, 1996.

[9] W Aiello, S Lodha, et al. Fast digital identity revocation[A]. Advances in Cryptology2CRYPTO. 98[C]. Germany: Springer2Verlag, 1998. 137 - 152.

[10] M T Goodrich, R Tamassia, et al. Implementation of an authenticated dictionary with skip lists and commutative hashing[A]. DARPA Information Survivablity Conference and Exposition (DISCEX II), Volume 2 [C]. California: IEEE Computer Society, 2001. 1068- 1084.

[11] M T Goodrich, R Tamassia. Efficient authenticated dictionaries with skip lists and commutative hashing[R]. Baltimore: Johns Hopkins Information Security Institute, 2000.

[12] A Arnes. Public key certificate revocation schemes. ph. D thesis[DB/OL]. http://www.pvv.ntnu.no/~andrearn/certrev/t2hesis/CertRevThesis_29Feb2000.ps.gz, 2000- 02- 29.

作者简介:

卿斯汉　男,1939 年 10 月生于湖南隆回,研究员,教授,博士生导师,主要研究领域为信息安全理论与技术.

周永彬　男,1973 年 11 月生于山东阳信,博士,主要研究领域为应用密码学、网络与信息安全理论与技术.