一种新的公钥基础设施——SPKI

张鹏程　陈克非
（上海交通大学计算机系　上海200030）

A New Public Key Infrastructure - SPKI

ZHANG Peng-Cheng　CHEN Ke-Fei
(Department of Computer Science, Shanghai Jiaotong Univ., Shanghai 200030, China)

Abstract Information security is a part of the core of computer field. This existing PKI system can not satisfy the more and more demand from people. Simple Public Key Infrastructure, SPKI, is new standard of public key certificate, which can bring a simple but precise naming and authorization system for distributed computing environment. The base theory of SPKI is presented, and the brief analysis about its study hotspots and shortcomings are also shown in the paper.

Keywords SPKI, Access control, Authorization, Certificate

1 引言

简单公钥基础设施(Simple Public Key Infrastructure, SPKI)是一种新的公钥证书标准。它能为分布式计算环境提供简单而精确的命名及授权体制。1999年,IETF 组织将 SPKI 标准化[3,4]。

SPKI 主要用于访问控制,它的突出特点有二:分布式安全管理、使用密钥来代替以往 X.509[6]体制使用的名字。此外,在 SPKI 的证书里可以定义不同形式的许可权限。应该说,SPKI 的全部思想都是基于公钥加密体制。所以,不管是证书的产生还是证书的使用,密钥都占主导地位。

2 SPKI 的历史探究

SPKI 的产生源于人们对现有 PKI 体制的不满意。人们有很多其它的需求:

1. 随意颁发,不依赖 CA 中心。分布式环境的发展使得集中式 PKI(比如:X.509)越来越无法满足当前需求。集中式 PKI 依赖于 CA 中心,电子证书的颁发和使用都不方便。人们需要有一种新的安全机制来方便快捷地颁发证书,而且可以赋予别人以一定的权力。比如:Alice 想让她的朋友来访问她的文件服务器,同时也希望有很好的、很灵活的安全机制来控制这种访问权限。

2. 权限可以传递。对于以往的 PKI 而言,权限需要详细定义在证书中。这种集中式的权限分配,造成证书格式复杂,内容庞大,不易使用。如果能让证书拥有者可以传递他拥有的全部或部分权限,就可以使权限的管理成为树形结构定义,既便于使用,又可将权限定义得非常精确。比如:Alice 也许会想允许她的一位朋友 Bob,可以将他获得的权限全部或部分地转授给他的朋友 Carol,然后让 Bob 控制 Carol 的访问权限。

3. 能够自由定义权限的内容。现实生活中,权限的表现形式是不一样的。有的可能就是读写文件这类很明显的访问控制需求,有的可以是"转账,查帐,透支"等等。如何来满足这各种各样的需求呢?

4. 更好的证书标示符。诸如 X.509这类 PKI 采用的是全局名字。为了保证证书名字的全局唯一性,使得证书名字非常不直观。真的只能用名字来标示证书吗?需要注意,公钥密码体系中的公钥本身就具有唯一性。

5. 更灵活的名字-密钥对应机制。同样,在以往的 PKI 中,名字与公钥的对应非常死板,不能表示一些复杂的关系。有没有更好的对应机制呢?

正是这些需求给我们带来了 SPKI 的产生与发展。

现在所说的 SPKI 其实是个混合体。1996年,Lampson 和 Rivest[7]提出"简单分布式安全基础设施(Simple Distributed Security Infrastructure, SDSI)"的概念。与此同时,Carl Ellison, Bill Frantz 等人设计出"简单公钥基础设施(Simple Public Key Infrastructure, SPKI)"。1997年,SDSI 和 SPKI 合并,成为"SPKI/SDSI"。为了简单起见,称它为"SPKI"或"SDSI"都是可以的。

3 SPKI 证书

SPKI 证书分为授权证书和名字证书,其中授权证书定义访问控制权限,而名字证书将公钥与名字或属性绑定在一起。需要注意的是,SPKI 完全可以在没有名字证书的情况正常工作。

3.1 授权证书

SPKI 授权证书是一个签名过的、包含5个字段的信息(见图1)。

这5个字段分别为:颁布者(issuer)、持有者(subject)、权限传递(delegation)、具体权限(authorization)和有效时间(validity dates)。它们通常表示为(I,S,D,A,V)。字段的具体内容说明如下:

颁布者:该字段说明谁颁布了这个证书,或是谁对这个证书签名。它可以是一个公钥或公钥的哈希值,该公钥对应的私钥对证书进行签名。由于证书是由密钥直接颁发的,故不涉及任何人名或计算机名。任何拥有公钥私钥对的实体都可以颁发证书。

持有者:该字段说明谁来持有这个证书。这个字段可以是一个公钥、一个公钥的哈希值或是一个名字。只有持有者字段中的公钥或名字可以使用该证书。

权限传递:该字段决定持有者是否可以将它拥有的权限

传递给其他人。该字段是个布尔变量。

具体权限：该字段定义了证书持有者拥有什么样的权限、拥有多大范围的权限等。颁布者可以自由定义这些信息。该字段的内容完全取决于具体的应用[5]。不同的应用有不同的控制权限，比如：文件服务器定义对某个目录的操作权限、银行定义对某账户金融活动的权限。

有效时间：该字段说明了证书在哪个时间段有效。

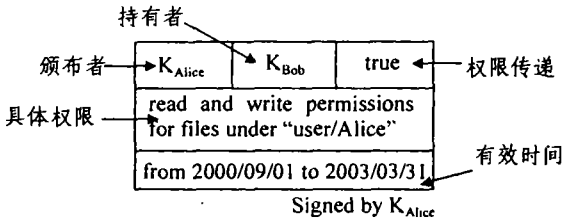

图1 证书结构——Alice 允许 Bob 对目录 user/Alice 下的文件进行读写，并且 Bob 可以传递权限，有效时间为2000/09/01至2003/03/31

如果某个持有者可以传递其权限，那么它可以将证书定义的权限全部或部分地转授给其他证书持有者。同样地，下一个持有者也可能可以传递权限。以此类推就形成了证书链(certificate chain)。某个持有者所拥有的权限是一个证书链来定义的。证书链中的每一环都有可能限制权限，这样证书链越长，持有者拥有的权限可能会越少。关于权限缩减的具体说明请见2.2节。

通过图1所示的例子来说明 SPKI 证书的使用。因为证书由 Alice 的私钥签名，所以其他人不能冒充 Alice 去伪造证书。Bob 如何使用他持有的证书？很简单，Bob 只需用他的私钥对他的请求签名，然后将签名后的请求与他的证书一起发送给 Alice 的文件服务器。文件服务器并不知道 Alice 的名字，它仅仅知道公钥 K_{Alice} 控制某些目录和文件。当文件服务器收到 Bob 的请求后，它会验证 Bob 的请求是否可以被允许。验证过程可以通过多种方法实现，但总体上由以下几步组成：

1. Alice 的文件服务器查看颁布者字段，判断是谁颁发该证书。服务器会发现是 K_{Alice}。通过查找本地的访问控制列表(Access Control List，ACL)，服务器能够知道 K_{Alice} 是否可以访问请求中提及的目录和文件。如果正确的话，服务器用 K_{Alice} 来判断证书是否真的是由 K_{Alice} 对应的私钥签名的。如果不是，请求则被拒绝。

2. Alice 的文件服务器查看持有者字段，判断对请求签名的私钥是否与持有者字段中的公钥对应。如果不是，请求则被拒绝。在这里可看到，服务器甚至不用知道密钥 K_{Bob} 属于 Bob。

3. Alice 的文件服务器比较 ACL、具体权限字段和请求所涉及的权限。如请求的权限越界，请求则被拒绝。此外，服务器还会查看有效时间字段，判断证书是否已失效。

4. 这时，Alice 的文件服务器可以确信 KBob 有资格访问它请求的目录。Bob 就可以进行他的操作了。

在使用证书的过程中，公钥是最重要的环节。名字可以不用涉及，这与传统的 PKI 有很大的区别。X.509是面向名字(Orient-Name)的，而 SPKI 是面向公钥(Orient-Key)的。

3.2 证书的归并(Reduction)

由于访问权限通常由一个证书链来确定，因此如何确定证书链的最终权限以及简化证书链是件非常重要的工作。当用户或一个公钥(以下我们都称之为 requester)试图获得某项服务，它必须向该服务的控制者出示它的授权证书。如果权限正确，请求则被批准。证书归并算法用于计算 requester 的权限，然后将 requester 的证书链简化为一个单独的权限证书。

证书归并：颁布者 I1给持有者 S1颁发一个授权证书，其权限为 A1，有效时间为 V1，并且允许 S1传递权限，故 D1为 true。故这个证书为(I1，S1，D1，A1，V1)。

S1可以传递权限，假设 S1将权限 A2转授给持有者 S2，有效时间为 V2。该证书为(I2，S2，D2，A2，V2)。这里，I2=S1，即第二个证书的颁发者就是第一个证书的持有者。

根据归并算法，这两个证书，或者说这个两级证书链所定义的权限等价于一个虚拟证书：(I1，S2，D2，Intersection〈A1，A2〉，Intersection〉〈V1，V2〉)。为了简单起见，颁布者 I1也可以直接给 S2颁发一个证书。它的内容与上面的虚拟证书一致。

我们再通过一个例子来看看如何使用证书归并算法来解决2.1节提到的证书链问题。

Alice 赋予 Bob 权限，而 Bob 将权限转授给 Carol，这样，Carol 的公钥 K_{Carol} 得到两个证书组成的证书链，如下图所示。

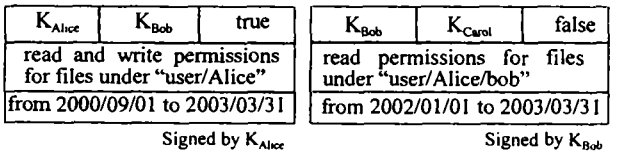

图2 K_{Carol} 的证书

根据前面提到的归并算法，可以得到 K_{Carol} 新的证书，见图3。

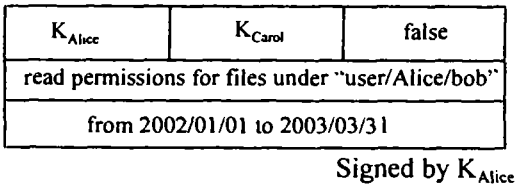

图3 K_{Carol} 的新证书

通常而言，当验证端接收到 requester 的请求和证书链后，将验证该请求和证书链，如果正确的话，请求被允许。如果为了以后访问方便，验证端可以为 K_{Carol} 签发新的证书。当然，也可以选择不颁发新证书。

3.3 名字证书

名字证书将公钥与名字或属性绑定在一起。名字证书的结构与授权证书类似，但它只有4个字段：颁布者(issuer)、持有者(subject)、名字(name)和有效时间(validity dates)。它们通常表示为(I，S，N，V)。除了名字字段外，其它字段的具体内容与授权证书中的一样，而名字字段定义被绑定的名字或属性。名字是任意字节的字符串，可以是人的名字，可以是任何实体的名字，也可以是一串引用，比如：Alice's boyfriend's sister，还可以是实体的布尔属性，比如："age-over-21"，"state-employee"。

SPKI 中没有全局名字，所有名字都是局部的，而这种局部的划分是由颁发者的公钥决定的。这些公钥拥有自己的名字空间。也就是说，对于名字证书来说，不同的颁发者可以使

用相同的名字。

在2.1节介绍证书结构中，我们提及到持有者字段也可以是一个名字。在这种情况下，名字证书就能发挥其作用。请看下面一个例子，图4所示的为一个授权证书和两个名字证书。

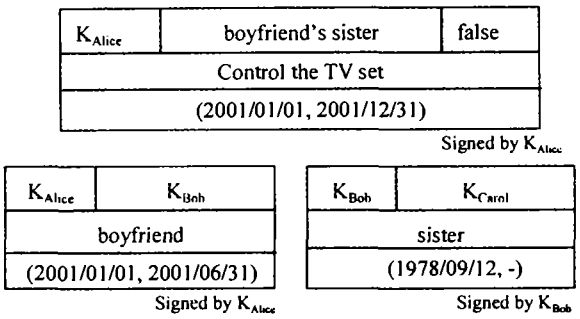

图4 名字证书

如上图，Alice 通过授权证书赋予"boyfriend's sister"以控制电视机的权力，通过名字证书定义她的男朋友是密钥 K_{Bob} 的持有者 Bob。同样，Bob 可以定义其姐姐是 Carol。注意，Bob 颁发的这个名字证书与 Alice 是无关的，他可以在持有 Alice 的证书之前颁发，也可以在这之后颁发。而且这个名字证书可以用到其他需要验证的地方，并不仅仅限于 Alice 的验证。此外，Bob 的姐姐不用是唯一的，可以是多个。这样，Bob 的姐姐们将获得同样的权限。

Carol 如何使用这些证书？

1. 她需要用自己的私钥对她的请求签名，也就是用 K_{Carol} 来签名。

2. 她必须发送图4中的名字证书来证明 K_{Carol} 就是授权证书中的"boyfriend's sister"。

3. 当验证端收到 Carol 发送的整个包后，开始验证 K_{Carol} 的权限。这里将使用名字证书的归并算法。通过图4左下角的名字证书，可以将"K_{Alice}'s boyfriend's sister"缩减为"K_{Bob}'s sister"；同样通过图4右下角的名字证书，就可得出"K_{Bob}'s sister"就是 K_{Carol}。

通过验证后，Carol 获得了控制 Alice 的电视机的权限。

4 SPKI 的研究动态与问题

IETF 的 SPKI 工作组成立于1996年，直至现在仍然在对 SPKI 体系进行不断研究[1]。通过一系列的 RFC 文档和草案[3~6]，我们可以对 SPKI 有个整体上的了解。这里有两个网址[2,9]给出了很多关于 SPKI 的研究工作。

SPKI 提倡者之一 Rivest 所在的麻省理工学院里，有很多人致力于关于 SPKI 各种算法和实现的研究。包括：名字证书路经搜索算法[10~12]，基于 Web 的应用[13,14]，SPKI 的具体实现[15,16]。

关于国内，目前研究的人很少，这方面的研究文章似乎没有。

由于 SPKI 仍是个新兴事物，围绕它的应用研究还处于起步阶段：

1. 移动代理安全

我们离"无所不在的计算技术"(ubiquitous computing)[20,21]越来越近了。智能化的设备将充斥我们生活的环境，Internet 会无所不在。然而，高连通情况下所涉及的安全和隐私问题也比以往更加重要。这些智能化的设备能够互相通讯，甚至可以为远端代码的执行而建立一个运行环境，而这个运行环境正是为智能软件代理和移动代码(mobile code)提供了终端平台。所以，这些设备必须只允许拥有权限的用户访问，而且当传送个人或秘密信息时必须保证通信安全。

SPKI 不仅编码方式简单，所需计算资源少。更为重要的是，它可以不依赖 CA 中心，这种分布式的 PKI 比集中式的 PKI(如：X.509)更适合分布式环境、更适合访问控制需求。如何具体实现还需努力研究。

2. 用 XML 编码 SPKI

XML 基于标准通用标记语言(SGML)。许多现有的无线设备(如 PDA、移动电话等)都已有内嵌的 XML 转译器，无线应用协议(Wireless Application Protocol，WAP)所用无线标记语言(Wireless Markup Language，WML)正是 XML 的子集。目前，IETF 组织正在起草 XML 编码 SPKI 的标准。若获通过，用 XML 编码的 SPKI 将具有更大的通用性，更佳的性能。

3. 与 Jini 相结合

SPKI 与 Jini 技术[23]的结合非常具有吸引力。Jini 技术是在 Java 所建立的、独立于平台并以网络为中心的计算模式基础上开发的，其目标是彻底冲破网络复杂连接方式所造成的壁垒，使网络连接极大的简化，要做到"即插即用"。可在任何时间、任何地点将任何智能器具接入 Internet 网络，实现网络与一切智能器具(也称含有 Jini 技术的器具)的无缝连结。

但 Jini 技术本身的安全性比较薄弱，缺乏有效的认证及授权手段。SPKI 的出现正可以弥补 Jini 的缺陷。

此外，SPKI 理论并没有完善，还存在一些问题。主要有以下几个：

1. 证书链路径搜索问题，特别是对于名字证书链。由于在持有者字段可能会是很长的一串引用，如何高效地、正确地找到名字引用的正确含义是一个需要解决的问题。此外，对于某个局部的名字空间，其名字的数量与计算名字引用的代价之间的关系，也是个重要的问题。对此，MIT 有人做了一定的研究[10~12]。但仍有很多问题没有很好解决。

2. 多条证书链问题。随着颁发的证书增多，有可能出现某两个证书之间有数条证书链，如何处理它们之间关系？是取最小权限，还是取最大权限，抑或是取权限的交集？在不同的应用背景有什么样的不同考虑？等等。

3. 验证出错的处理。如果验证端在验证不通过时，报错的信息是不给任何提示，还是给出在何处能取得所需的证书？如果是后者，如何定位相应的证书颁发者？

4. 权限传递的控制。一旦证书颁发者允许持有者传递权限的话，他将无法精确控制权限的传递。比方说，Alice 也许允许 Bob 将其权限传递给 Bob 的工作小组，但不希望 Bob 传递给 Carol。这时，Alice 很难精确控制。

结论 我们介绍了 SPKI 这类新的公钥基础设施，并讨论了它的一些相关问题。SPKI 主要用于访问控制，控制权限可以自由定义。证书的存储分散在各个用户和计算机，而不是集中在某几个 CA 手中。SPKI 强调分布性，任何拥有公私钥对的实体都可以给其他人颁发证书。用公钥代替名字来作为证书的标识符，用局部名字空间代替全局名字空间。

SPKI 本身的特点非常适合分布式环境，虽然现在支持 SPKI 的厂商不多，但从当前网络发展的趋势来看，SPKI 的应用前景非常广阔。

参 考 文 献

1 Internet Engineering Task Force. Simple Public Key Infrastructure (spki). Available at: http://www.ietf.org/html.charters/spki-charter.html., 1997

2 Rivest R L. Cryptography and Information Security Group Research Project: A Simple Distributed Security Infrastructure (SDSI). Available at: http://theory.lcs.mit.edu/~cis/sdsi.html., 1996

3 Ellison C, Frantz B, Lampson B, Rivest R, Thomas B, Ylonen T. SPKI Certificate Theory, Request For Comments 2693, Internet Engineer-ing Task Force, Sep. 1999

4 Ellison C. SPKI Requirements, Request For Comments 2692, Internet Engineering Task Force, Sep. 1999

5 Ellison C M, Frantz B, Lampson B, et al. SPKI Examples. The Internet Society, March 1998. Available at: http://world.std.com/~cme/examples.txt; This is draft-ietf-spki-cert-examples-01.txt

6 Ellison C M, Frantz B, Lampson B, et al. SPKI Structure. The Internet Society, March 1998. Available at: http://world.std.com/~cme/spki.txt; This is draft-ietf-spki-cert-structure-05.txt

7 Rivest R L, Lampson B. SDSI-a simple distributed security infrastructure. Available at: http://theroy.lcs.mit.edu/~rivest/sdsi10.ps., Aug. 1996

8 Recommendation X.509, The Directory - Authentication Framework, volume VIII of CCITT Blue Book, CCITT, 1988. 48~81

9 Ellison C M. SPKI/SDSI certificate documentation. Available at: http://world.std.com/~cme/html/spki.html., 2001

10 Elien J-E. Certificate Discovery Using SPKI/SDSI 2.0 Certificates. Master's thesis, EECS Dept., Massachusetts Institute of Technology, May, 1998. Available at: http://theory.lcs.mit.edu/~cis/theses/elien-masters.pdf

11 Clarke D, et al. Certificate Chain Discovery in SPKI/SDSI. To appear in Journal of Computer Security. Available at: http://theory.lcs.mit.edu/~rivest/ClarkeElElFrMoRi-CertficateChainDiscoveryInSPKISDSI2.ps

12 Burnside M, Clarke D, Devadas S, Rivest R. Integrating Resource Discovery and Communication with SPKI/SDSI Security. In prepara-tion for submission to a conference

13 Maywah A. An Implementation of a Secure Web Client Using SPKI/SDSI Certificates. Master's thesis, EECS Dept., Massachusetts Institute of Technology, June 2000. Available at: http://theory.lcs.mit.edu/~cis/theses/maywah-masters.ps

14 Elcock G. Web-based user interface for a Simple Distributed Security Infrastructure (SDSI). Master's thesis, EECS Dept., Massachusetts Institute of Technology, June 1997. Available at: http://theory.lcs.mit.edu/~cis/theses/elcock-masters.ps

15 Fredette M H. An implementation of SDSI - the Simple Distributed Security Infrastructure. Master's thesis, EECS Dept., Massachusetts Institute of Technology, May 1997. Available at: http://theory.lcs.mit.edu/~cis/theses/fredette-masters.ps

16 Morcos A. A Java implementation of Simple Distributed Security Infrastructure. Master's thesis, EECS Dept., Massachusetts Institute of Technology, May 1998. Available at: http://theory.lcs.mit.edu/~cis/theses/morcos-masters.ps

17 Hewlett-Packard. e-Speak. Available at: http://www.e-speak.hp.com

18 Intel. Intel Common Data Security Architecture. Available at: http://developer.intel.com/ial/security

19 UC Berkeley. The OceanStore Project: Providing Global-Scale Persistent Data. Available at: http://oceanstore.cs.berkeley.edu

20 Banavar G, et al. Challenges: An Application Model for Pervasive Computing. In: Proc. ACM MOBICOM, Aug. 2000

21 Dertouzos M. The Future of Computing. Scientific American, Aug. 1999

22 Oaks S. Java(tm) Security, First edition, O'Reilly Press, May 1998

23 Arnold K, et al. The Jini Specification. Addison - Wesley, June 1999

(上接第160页)

3 Bae J-S, Jeong S-C, Seo Y, et al. Integration of Workflow Management and Simulation [J]. Computers & Industrial Engineering, 1999, 37: 203~206

4 张莉, 王雷. 过程模拟技术及其支持环境 PMSE[J]. 软件学报, 1997, 6(增刊): 565~575

5 林慧苹, 范玉顺, 吴澄. 支持企业经营过程重组的工作流仿真技术研究[J]. 信息与控制, 2001, 30(1): 11~15

6 刘敬军, 张申生, 全春来, 陈加栋. 面向并行工程的集成多视图产品开发模型及管理研究[J]. 中国机械工程, 1998, 9(9): 33~35

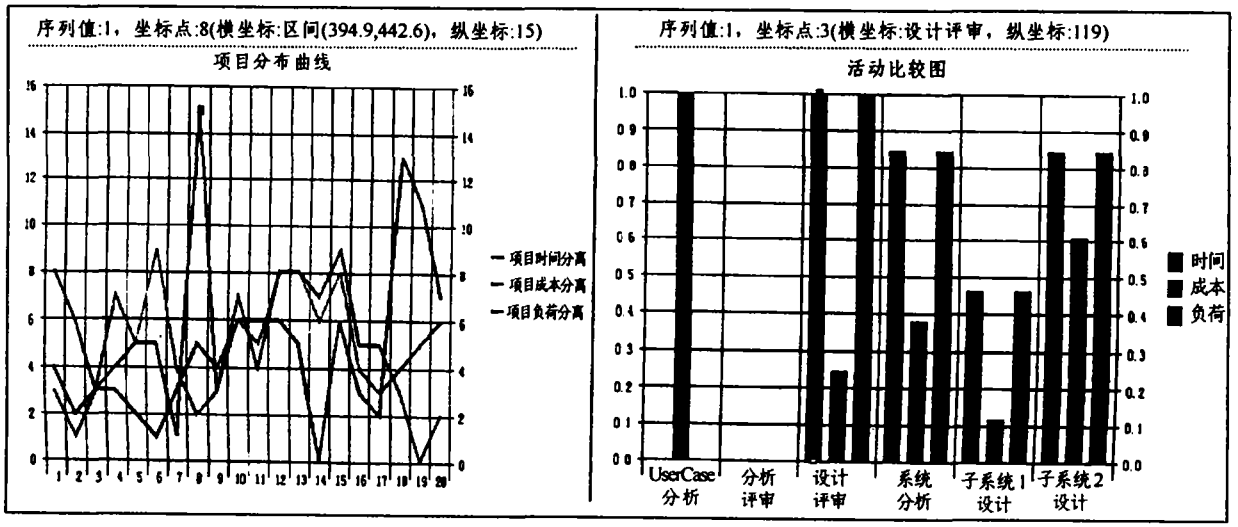

图5 仿真结果曲线图和直方图