第32卷 第19期 计 算 机 工 程 2006年10月
Vol.32 № 19 Computer Engineering October 2006
·开发研究与设计技术· 文章编号：1000—3428(2006)19—0264—03 文献标识码：A 中图分类号：TP 391

基于 XML 电子病历多重签名方案的设计与实现

孟 健[1,2]，曹立明[1]，王小平[1]，姚 亮[1]
（1. 同济大学计算机科学与工程系，上海 200092；2. 上海水产大学信息学院，上海 200090）

摘 要：根据 XML 的树型结构与病历的特点的吻合性，实现了 XML 格式的电子病历文档；对多重签名作了相应的改进，并应用于 XML 电子病历文档中的规则（rule），设计了 XML 电子病历多重签名方案；基于 XML 的签名规范，实现了 XML 电子病历多重签名，提高了签名的效率和灵活性，具有一定的可扩展性。

关键词：XML；电子病历；多重签名；XPath

Design and Realization on Multisignature Scheme of XML-based Electronic Medical Record

MENG Jian[1,2], CAO Liming[1], WANG Xiaoping[1], YAO Liang[1]
(1. Department of Computer Science and Engineering, Tongji University, Shanghai 200092;
2. School of Information, Shanghai Fisheries University, Shanghai 200090)

【Abstract】As the tree structure of XML agrees with the characteristics of medical records, an XML-based electronic medical record is realized. An improvement is made to multisignature and is used in rules of XML electronic medical records. A multisignature scheme is designed and realized. The efficiency and flexibility is improved through signing the rules of the record, and the expansibility can be obtained to some extent.

【Key words】 XML; Electronic medical record; Multisignature; XPath

随着医疗事业的发展，人们对医疗服务质量要求逐步提高，要求各医院之间做到病人、医疗设施等的信息共享。而目前的签名体系[5]还不完善，无法区分病历不同部分的相关责任人，缺乏实用价值。

本文针对这一情况，建立了基于XML格式[2,3]的电子病历文档针对XML文档的特点对目前的多重签名做了相应的改进，提出对文档规则进行分解、签名，建立了XML电子病历文档多重签名框架，描述了产生多重签名、验证多重签名的详细步骤，实现了一个完整的XML电子病历文档多重签名的过程。

1 电子病历及 XML 电子病历文档的设计与实现

1.1 电子病历

电子病历(Electronic Medical Record)是以电子化方式管理的有关个人终生健康状态和医疗保健行为的信息。它可在医疗中作为主要的信息源取代纸张病历，提供超越纸张病历的服务，满足所有的医疗、法律和管理需求。

1.2 XML 电子文档的设计与实现

我们采用XML格式，建立电子病历，依据XML的树型结构[1,4]，将每一位病人的病历作为一个结点，产生门诊病历等子结点，以每一位相关医生都能找到自己需要签名的子文档作为创建结点的依据，产生了一个XML电子病历文档的框架，如图 1 所示。

根据该 XML 电子病历文档框架，创建了如下所示的一个电子病历文档，记录了病人在门诊、住院和手术 3 个阶段的相应诊断与治疗。

```
<?xml version="1.0" encoding="GB2312"?>
<!DOCTYPE patient SYSTEM "mr. dtd">
<?xml 一 stylesheet type="text/xsl" href="MR.XSL"? >
<patient pid="213214780923031" vid="1" status="editing"nm ="Tom">
  <outrec>
<outhis>
  <natinf>出生日期为 1978-9-23，就诊日期 2004-7-15</natief>
  <depic>content</depic>
  <elediag>depiction</elediag>
</outhis>
<outcheck>
  <X-ray>result……</X-ray>
</outcheck>
  </outrec>
  <inrec>
<incase>
<firrec>record1</firrec>
  <dicrec>record2</dicrec>
</incase>
<incheck>
</incheck>
  </inrec>
  <operrec>
<beope><agreement>content</agreement><summery>content</su mmery></beorp>
```

作者简介：孟 健(1977－)，男，博士生，主研方向：分布系统；曹立明，博导、教授；王小平，博士、副教授；姚 亮，硕士生

收稿日期：2005-10-27 **E-mail：**michaelmeng1978@163.com

<onope><anaesthesia>content</anaesthesia><onnur>content</onnur></onope>

<afope><operator>sig</operator><ICU>sig2</ICU><general>sig3</general></afope>

</operrec>

</patient>

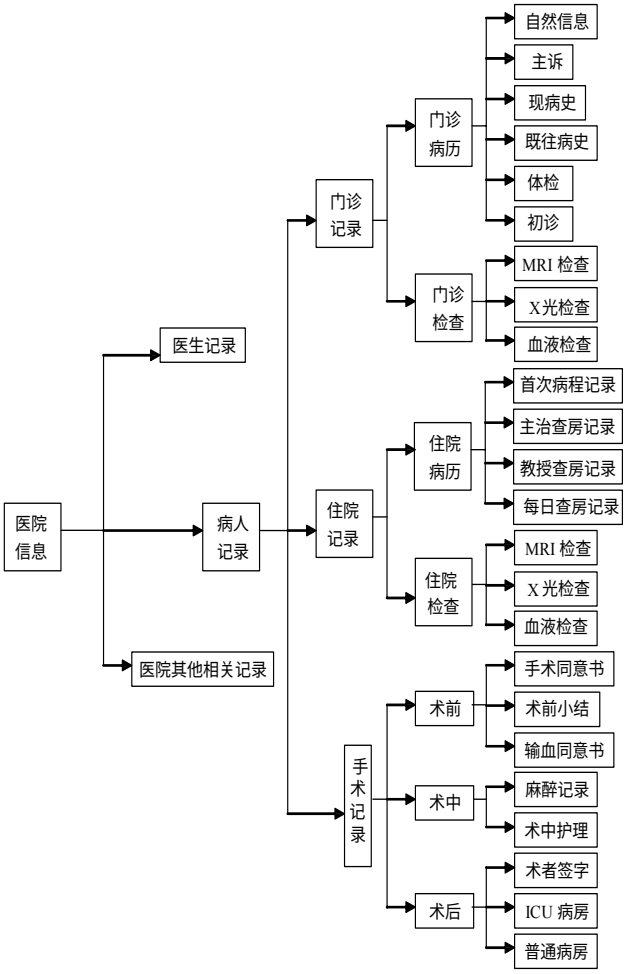

图 1 XML 电子病历文档框架

2 XML 电子病历文档多重签名的设计与实现

2.1 多重签名

原始的多重签名过程是将整个文档的拷贝发送至每个签名者、签名者利用签名算法进行签名、参与签名者的签名连接起来作为文档的多重签名。这种模式签名和验证的效率都比较低。2001 年，Tzong-Chen Wu[7]提出使用文档分解实现授权多重签名模式，允许签名者对其负责的子文档签名，提高了签名和通信的效率，无法保证分解成的子文档均有意义。

我们在Tzong-Chen Wu提出的算法基础上做了相应的改进，根据XPath产生相应的规则集，从形式上实现XML文档各个子部分与不同签名者的映射，并设计了基于改进型XML多重签名方案的框架，利用XML的签名规范[8]，并结合基于离散对数的广播多重数字签名，实现了XML形式的电子病历文档的签名方案，利用XPath变换将每个签名者与其需要签名的规则关联起来生成个人的签名，并最终生成XML多重签名。

2.2 多重签名方案的框架设置

我们设计的多重签名方案框架主要由 4 个模块组成：分别为一组签名者 U1、U2、U3... 它们共同构成了一个小组 G；系统授权机制（system authority，SA）；文档分配机制（document dispatcher，DD）和签名收集器（signature collector，SC)。电子病历 XML 文档通过 DD 被形式分解为一系列子文档，这些子文档通过一个映射关系（分配算法）分配给各签名者。SA 提供系统参数初始化，并为小组 G 整体和小组中的个体 U1、U2、U3...产生密钥和公钥。SC 通过与各个签名者交互、收集和验证由每个签名者产生的个人签名，产生多重签名。各部分之间的关系如图 2 所示。

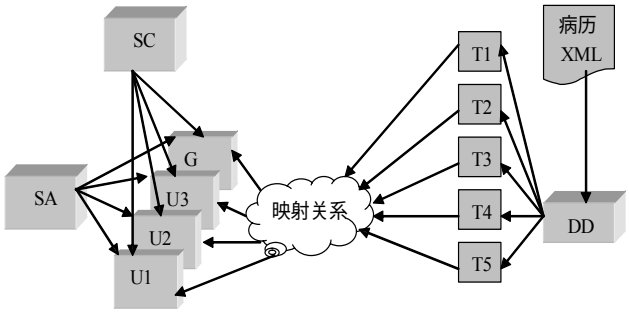

图2 多重签名框架

2.3 为每位负责医生和医院产生密钥/公钥对

初始化：SA首先选择两个大素数p、q，$p\geq 2^{512}$，$q\geq 2^{140}$，$q \mid p-1$，$q\geq 2^{140}$，α，α为q的阶；

选择一个单向 hash 函数 h，对于任何输入 x，h(x)均在GF(p)中；

发布 p、q、α 和 h；

SA 可接受注册，一旦注册完成，SA 即为每一位负责医生产生一个密钥/公钥对。

通过使用（x_j,y_j）（密钥/公钥对），SA为医院产生一个密钥/公钥对（X,Y）：

$$X=\sum_{\vec{u}j\in G} x_j \bmod q \qquad Y=\sum_{\vec{u}j\in G} y_j \bmod p$$

SA通过安全通道为医院的每位医生发送x_i（密钥），并向SC和医院的所有医生发布Y和y_j。

2.4 多重签名产生阶段

设M为需要进行多重签名的电子病历文档，医院为G。首先由DD将文档M分解为一组子文档$\Gamma=\{w_1,w_2,\ldots,w_m\}$，分解时遵循规则集$T=\{t_1,t_2,\ldots,t_n\}$。通过对XPath处理器应用规则$t_i$，即可获得一个子文档$w_i=C_{ti}(M)$。令$T_j$和$M_j$分别为T和M的子集，与用户$u_j$对应，为医院G产生电子病历文档M多重签名的过程描述如下：

步骤 1：DD向医生u_j发送{h(M)，Mj，T_j}，向SC发送{h(T),h(M)}；

步骤 2：G中每一个u_j从分配给他的M_j中分解出若干w_j，并共同检查M的完整性，完整性的检查可通过验证等式$h(M)=h(w_1 \| w_2 \| \ldots \| w_m)$；

步骤 3：每一个G的成员u_j从T_j中分解出若干个t_i，计算$w_i=C_{ti}(M)$，并验证计算获得的每一个w_i是否与接收到的每个w_i相同。若所有w_i均得到验证，则u_j随机选择一个整数$Z_j\in Z_q$，按下列式子计算获得r_j和R_j。

$$r_j=\alpha^{Zj} \bmod p$$

$$R_j=r_j^{h(Tj \| rj)} \bmod p$$

将R_j发送给G中的其他医生及SC。

步骤 4：G中的每一位医生u_j均根据获得的信息和下列公式计算R和S_j

$$R=\prod_{uk\in G} R_k \bmod p$$

$S_j=(Z_j h(T_j \| r_j)R+x_j h(M \| R))) \bmod q$

并将$\{T_j,r_j,s_j\}$发送给SC。(r_j,s_j)是u_j对M的个人签名。值得注意的是，产生S_j的时候，同时用到了T_j和M，这就确定了u_j使用T_j中的t_j对M_j进行签名。尽管u_j是对规则签名（T_j或M的结构），实际上也是对子文档M_j进行了签名。这与其他的多重签名方案不同，同时也降低了S_j的计算开销。

步骤 5：SC通过从T_j中分解出t_i，检查T的完整性，并验证等式

$H(t)=h(t_1 \| t_2 \| \dots \| t_m)$

步骤 6：SC通过公式$R=\prod_{uk\in G} R_k \bmod p$为组群中每个成员$u_j$验证$(r_j,s_j)$，验证等式

$r_j^{h(Tj \| rj)R}=(\alpha^{Sj})(y_j^{h(h(M) \| R)})(\bmod p)$

步骤 7：若以上的步骤产生的所有个人签名均正确，则SC 计算 S。

$$S=\sum_{uj\in G} S_j \bmod q$$

并将(R,S)作为 G 对电子病历文档 M 的签名发布出去。

2.5 多重签名的验证阶段

验证者可以通过计算R^R与$(\alpha^S)(Y^{h(h(M) \| R)})$是否相等来验证多重签名。即等式

$R^R=(\alpha^S)(Y^{h(h(M) \| R)})$

是否成立。

多重签名一旦产生，即可作为医院的整体签名，它是由医院的所有医生的个人签名共同产生的，体现了医院的整体性和医生的个体性的统一，当文档产生问题时，可以首先追究到医院，再由医院追究到具体负责医生，体现了签名和责任的层次性。

2.6 XML 电子病历文档多重签名的实现

我们采用Apache Xerces1-4-3 的XML处理器[6]，Apache Xalan-J-2-5-1 的XPath处理器，以及IBM's XML Security Suite的XML签名API[9]组成的开发环境来实现上述方案。其体系结构为一个基于Browse/Server/Database(B/S/D)的三层体系结构(图 3)：客户端使用XML浏览器访问Web服务器，通过Web服务器访问CA服务器或信息服务器。Web服务器提供文档中转，发布和访问CA服务器与信息服务器；CA服务器提供安全注册，安全信息发布服务(发布证书和撤销列表)；信息服务器中存储各种格式的数据客户通过客户端计算机向CA服务器进行注册获取自己的证书，客户对生成的XML文档进行签名后，通过Web服务器查询到相应的信息，从签名者的证书获取他的公钥，即可对已经签名的XML文档进行验证。

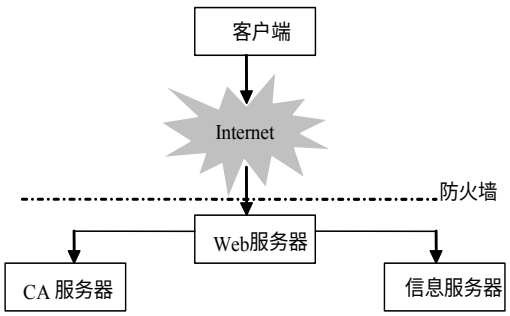

图3 多重签名方案的三层体系结构

签名生成过程：

(1)生成一个待填充的签名 DOM 树 TemplateGenerator siggen=new TemplateGenerator(doc, Xsignature. Hash1, Canonicalizer. W3C2, SignatureMethod. RSA)。

(2)对生成的 DOM 对象添加 reference 元素。

(3)构造 KeyInfo 元素。

(4)将 KeyInfo 信息插入签名元素 KeyInfo.insertTo (sigElement)。

(5)生成一个签名上下文实例：SignatureContext sigContect=new SignatureContext()。

(6)调用 sign 方法进行签名，sigContext.sign (sigElement,key)。

(7)将 XML 签名输出处理，可以输出到指定文件或保存到数据库中。

签名验证过程：

(1)创建 SignatureContext 实例和设置参数，相当于步骤(5)。

(2)准备公钥：可以从 KeyInfo 元素中获得，也可以从 KeyStore 中获取。

验证：Validity validity=sigContext.verify(sigElement,key)。

重复以上的步骤，即可完成多重签名的过程，实现对 XML 电子病历文档的多重签名。

3 结论

本文提出的构建基于 XML 的电子病历文档及多重签名方案，其安全性是基于离散对数求解的困难程度。充分利用了 XML 文档的树型结构特点，应用 XPath 变换规则，将文档按照不同医生的责任，分割为具有意义的子文档。医生只对自己负责的子文档的 XPath 变换的表达式签名，降低了签名计算量和通信开销，并有一定的可扩展性和灵活性，具有较高的实用意义。

参考文献

1 Bertina, Carminati B, Ferrari E. XML Security[R]. University of Insubria at Como, Italy, 2001: 44-58.

2 Boyd C. Multisignatures Based on Zero Knowledge Schemes[J]. Electronic Letters, 1991, 27(22): 2002-2004.

3 Camenisch J. Efficient and Generalized Group Signatures[C]. Proc. of Euro-Crypt'97, 1997, 1233: 465-479.

4 Erdmann M, Studer R. How to Structure and Access XML Documents with Ontologies[J]. Data and Knowledges Engineering, 2000, 36(3): 317-335.

5 Eastlake D, Reagle J, Solo D. XML Signature Syntax and Processing[Z]. http://www.w3g.org/TR/xmldsig-core/, 2002.

6 Eastlake D, Reagle J. XML Encryption Syntax and Processing[Z]. http: //www.w3.org/TR/xmlenc-core/, 2001.

7 Wu T C, Huang C C, Guan D J. Delegated Multi-signature Scheme with Document Decomposition[J]. Journal of Systems and Software, 2001, 55(3): 321-328.

8 W3C XML Signature Working Group. XML Signature Syntax and Processing[EB/OL]. http://www.w3.org/TR/xmldsigcore, 2002-02-12.

9 IBM's XML Security Suite. XML Signature Implementation [EB/OL]. http: //www.alphaworks.ibm.com/tech/xmlsecuritysuite, 2003-09-28.