[19] 中华人民共和国国家知识产权局

[51] Int. Cl[7]
G06F 1/00
H04L 29/06

[12] 发明专利说明书

专利号 ZL 00807108.X

[45] 授权公告日 2005 年 7 月 20 日

[11] 授权公告号 CN 1211719C

[22] 申请日 2000.2.24 [21] 申请号 00807108.X
[30] 优先权
[32] 1999.3.2 [33] US [31] 09/260249
[86] 国际申请 PCT/GB2000/000661 2000.2.24
[87] 国际公布 WO2000/052557 英 2000.9.8
[85] 进入国家阶段日期 2001.11.2
[71] 专利权人 国际商业机器公司
地址 美国纽约州
[72] 发明人 K·E·西蒙斯 W·H·温斯巴勒
审查员 李 熙

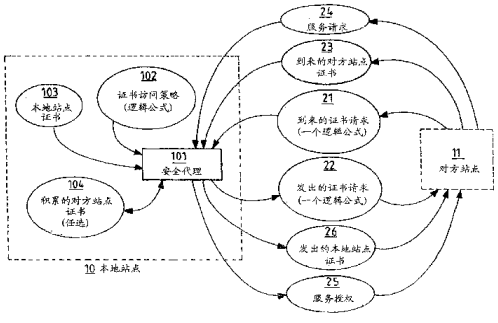
[74] 专利代理机构 中国专利代理(香港)有限公司
代理人 吴立明 梁 永

权利要求书 3 页 说明书 16 页 附图 3 页

[54] 发明名称 数据网络中使用自动增量证书公开的相互认证

[57] 摘要

在客户/服务器计算中，特别在电子商务领域中，在客户和服务器之间传递数字签名的证书以产生各方之间的信任。然而，这需要在公开方知道关于接收方的任何信息前一方对另一方公开(某一方必须先行)它的证书(这可能被认为是敏感的)。为解决这一问题，本发明实现了一种证书公开协商，称为自动增量证书公开。在一个本地站点保存的每一个证书都与一个基于对方站点证书的访问策略相关联。到来的证书请求与该访问策略逻辑组合以导出进一步的协商响应。



ISSN1008-4274

1. 一种用于客户/服务器网络的数据处理装置，其特征在于，一个客户数据处理装置给服务器数据处理装置发送一个数据处理请求，服务器数据处理装置根据该请求执行数据处理，并把回答返回到客户数据处理装置，该数据处理装置包括:

用于存储多个本地站点证书的存储设备;

用于从对方站点数据处理装置接收第一证书请求的设备，由第一证书请求请求的证书是存储在存储设备中的本地站点证书，它满足用第一证书请求提供的第一逻辑表达式；和

给对方站点数据处理装置发送一个第二证书请求的设备，该第二证书请求依赖于第一证书请求的内容，由第二证书请求请求的证书是对方站点证书，它满足用第二证书请求提供的第二逻辑表达式。

2. 权利要求 1 的装置，其特征在于，存储设备还存储多个证书访问策略，每一策略根据对方站点证书支配对相应本地站点证书的访问。

3. 权利要求 2 的装置，另外包括:

决定设备，用于决定用第一证书请求提供的第一逻辑表达式是否由在存储设备中存储的本地站点证书的组合满足，和当确定用第一证书请求提供的第一逻辑表达式由在存储设备中存储的本地站点证书的组合满足时，决定在存储设备中存储的证书访问策略和支配满足由第一证书请求提供的第一逻辑表达式的所述本地站点证书的组合是否由本地可用的对方站点证书满足;

发送设备，用于当决定设备确定证书访问策略由本地可用的对方站点证书满足时给对方站点数据处理装置发送本地站点证书的组合;

逻辑组合设备，用于当决定设备确定本地可用的对方站点证书不满足证书访问策略时，逻辑组合（a）接收到的第一证书请求，（b）存储的本地站点证书和（c）存储的证书访问策略以导出在为对方站点证书的第二证书请求中的第二逻辑表达式，它与本地可用的对方站点证书一起满足支配本地站点证书的组合的本地证书访问策略，所述本地站点证书满足以为本地站点证书的第一请求提供的第一逻辑表达式。

4. 权利要求 1 的数据处理装置，其特征在于，对方站点数据处理装置是一个客户数据处理装置。

5. 权利要求 1 的数据处理装置，其特征在于，对方站点数据处理装置是一个服务器数据处理装置。

6. 权利要求 1 的数据处理装置，其特征在于，对方站点证书高速缓冲存储在本地存储器中。

7. 权利要求 3 的数据处理装置，其特征在于，决定设备在发现这种组合不存在时给对方站点数据处理装置发送一个转达该发现的消息。

8. 权利要求 1 的数据处理装置，其特征在于，客户/服务器网络是因特网。

9. 一种操作用于客户/服务器网络的数据处理装置的方法，这里，一个客户数据处理装置给服务器数据处理装置发送一个数据处理请求，服务器数据处理装置根据该请求执行数据处理，并把回答返回到客户数据处理装置，该数据处理装置包括用于存储多个本地站点证书的存储设备；该方法包括步骤：

从对方站点数据处理装置接收一个第一证书请求，由第一证书请求请求的证书是存储在存储设备中的本地站点证书，它满足用第一证书请求提供的第一逻辑表达式；

给对方站点数据处理装置发送一个第二证书请求，它依赖于第一证书请求的内容，由第二证书请求请求的证书是对方站点证书，其满足用第二证书请求提供的第二逻辑表达式。

10. 权利要求 9 的方法，其特征在于，存储设备还存储多个证书访问策略，每一策略根据对方站点证书支配对相应本地站点证书的访问。

11. 权利要求 10 的方法，另外包括步骤：

决定用第一证书请求提供的第一逻辑表达式是否由在存储设备中存储的本地站点证书的组合满足，和当确定用第一证书请求提供的第一逻辑表达式由在存储设备中存储的本地站点证书的组合满足时，决定在存储设备中存储的证书访问策略和支配满足由第一证书请求提供的第一逻辑表达式的所述本地站点证书的组合是否由本地可用的对方站点证书满足；

当决定步骤确定证书访问策略由本地可用的对方站点证书满足时给对方站点数据处理装置发送本地站点证书的组合;

当决定步骤确定本地可用的对方站点证书不满足证书访问策略时，逻辑组合（a）接收到的第一证书请求，（b）存储的本地站点证书和（c）存储的证书访问策略以导出在为对方站点证书的第二证书请求中的第二逻辑表达式，它与本地可用的对方站点证书一起满足支配本地站点证书的组合的本地证书访问策略，所述本地站点证书满足用为本地站点证书的第一请求提供的第一逻辑表达式。

12. 权利要求 9 的方法，其特征在于，对方站点数据处理装置是一个客户数据处理装置。

13. 权利要求 9 的方法，其特征在于，对方站点数据处理装置是一个服务器数据处理装置。

14. 权利要求 9 的方法，其特征在于，对方站点证书高速缓冲存储在本地存储器中。

15. 权利要求 9 的方法，其特征在于，客户/服务器网络是因特网。

数据网络中使用自动增量证书公开的相互认证

发明领域

本发明涉及客户/服务器（也称为“分布式”）计算领域，其中一个计算设备（“客户”）请求另一个计算设备（“服务器”）执行客户的部分工作。

背景技术

在过去几年间在信息技术世界客户/服务器计算变得越来越重要。这一类型分布式计算允许运行在一个机器上的软件过程（例如客户）委派它的一些工作给运行在另一个机器上的软件过程（例如服务器），后者可能更适合执行该项工作。客户和服务器也可以是运行在同一机器上的单独的软件过程。

在客户/服务器系统中，非常重要的是客户和服务器在它们进行有意义的相互反应之前要形成足够的彼此信任级，因为在客户请求服务器处理期间可能交换的信息和/或服务器返回到客户的处理结果可能是高度敏感的信息。经常是，客户和服务器先前彼此没有关系，因此它们必须进入某些类型的初始对话，以便在它们公开任何可能敏感的信息之前决定它们是否可以彼此信任。说明这一点特别有用的一个很好的例子是当客户是一个通过因特网给一个万维网服务器应用发送电子商务请求的万维网浏览器应用时。在这些参与方之间进行初始反应时万维网客户和万维网服务器先前没有任何关系，万维网客户例如可能非常不愿意通过因特网给万维网服务器提供信用卡号码。

在现有技术中公知在客户和服务器之间交换证书（亦即由证书发布者给证书拥有者数字签名的确认）以便在它们之间发展信任。通过使用发布者的私钥(private key)签名证书并可以使用发布者的公钥验证。证书集合拥有者的一个或多个属性，每一个属性由名字/值对组成，并说明由发布者声称的拥有者的一些特性。每一证书还包括证书拥有者的公钥。拥有者可以使用相应的私钥回答盘问或者演示证书的所有者身份。拥有者也可以使用私钥签名由第三实体拥有的另一个证书。

这样，如在现有技术中众所周知，证书可以结合到一个链中，这

里一个证书的拥有者是在该链中下一个证书的发布者。这些链可以被认为是从一个已知实体（在该链中第一个证书的发布者）追踪到需要建立信任的提交实体的信任网。提交实体是在该链中最后证书的拥有者。提交实体可以通过演示拥有证书中包含的公钥的私钥配对而演示该证书的所有者身份。其它的支持证书由与提交实体具有直接或间接关系的实体拥有，并且虽然它们不由该提交实体拥有，但是提交实体确实保存和提交它们的复制件。每一支持证书包含公钥，使用其私钥配对签名链中的下一个证书。

所有提交的证书都与演示（可能是间接的）提交实体和发布链中第一证书的已知实体之间的关系有关。通过检查该链中证书的属性可以推断关系的性质。可以提交多重信任链以建立更高的信任度或演示提交实体另外的属性和它与已知实体之间的关系。

使用证书建立相互信任的现有技术可以分为两种基本方法。第一种方法在下面的文章和书籍中说明：由 A. Frier, P. Kocher 所著，网景通信公司 1996 年 11 月 18 日出版的“SSL 3.0 协议”；由 T. Dierks, C. Allen 所著“TLS 协议版本 1.0”，1998 年 11 月 13 日的 draft-ietf-tls-protocol-0.6.txt；S. Farrell 所著“用于基于属性证书的授权的 TLS 扩展”，1998 年 8 月 20 日的 draft-ietf-tls-attr-cert-0.1.txt。这一方法将称为 SSL 方法，因为其由 SSL、TLS 和 TLS 扩展为基于属性-证书的授权使用。在 SSL 方法中，客户和服务器可以如下交换证书。服务器通过单方公开一个预先选择的证书启动协商。它可以包括对客户证书的请求，包括服务器可以接受的证书类型和在属性证书的场合一个指示所需要的属性的模板。

第二种方法由下面的文章和书籍说明：N. Ching, V. Jones 和 M. Winslett 所著“数字图书馆中的授权：安全访问跨越企业边界的服务”，关于数字图书馆中研究和技术进步论坛的 ADL 96 会议录，1996 年在华盛顿特区召开，可在http://drl.cs.uiuc.edu/security/pubs.html看到；M. Winslett, Nching, V. Jones 和 I. Slepchin 所著“在万维网上使用数字证书”，计算机安全杂志，1997 年 5 月，255-267 页，可在http://drl.cs.uiuc.edu/security/pubs.html看到。我们将称该第二种方法为数字证书方法。在这一方法中，当由一个客户对服务器进行服务请求而没有附加合适的证书时，服务器给客户发送一个支配

该服务的策略。策略是一个证书公式，也就是说，是所请求的证书和对证书包含的属性的表述的限制的逻辑组合。可以使用策略表征提交实体所需要的特性及其与已知实体的关系。通过接受这一策略作为为证书的请求，客户有机会选择私人证书以提交授权服务。通过给客户发送策略，服务器卸载证书选择。该实践也允许不同的服务器具有非常不同的策略，要求不同的客户属性和接受由不同授权机构发布的证书。

这两种现有技术方法都支持服务器给客户发送证书请求，包括对该服务器可接受的证书特征。然而，本发明注意到了这一现有技术中如下的缺点。

在 SSL 方法中，对服务器来说在公开服务器的证书之前没有机会检验任何关于客户的信息。服务器可以作为高度秘密对待自己的证书，从而，如果客户和服务器没有能够建立相互信任，那么服务器已经交付给客户一则高度敏感的（秘密的）信息。此外，如果由服务器公开的证书不满足客户，则该客户没有机会从服务器请求附加证书。这当客户和服务器先前没有关系时可能是一个严重的问题。在这一场合，不太可能任何单一证书发布者接受对所有客户就感兴趣的所有服务器属性的授权。

基于数字证书方法的现有系统的缺点随客户希望只对已经建立某种信任程度的服务器公开证书而产生。使用数字证书方法开发的现有系统具有支持的客户-证书提交策略，它把服务分成等价的类，然后为每一等价类，给两个类目中的一个分配每一客户证书。在第一类目中的证书可以以在等价类中的任何服务请求提交。在第二类目中的那些只可以在为授权交互咨询用户后才可以提交。这些咨询允许用户从第二类目移动证书到第一类目，允许随后的自动提交。然而，该机构不是完全自动的，它需要用户在接触新服务类时能够进行信任决策。

在数字证书方法的环境内，一种备选技术由 Winslett 等人简要说明（上面引用的），从而客户可以请求服务器证书打开它自己的证书。可以使用该技术来实现协商，其中由每一参加者有单一的证书请求。每一服务与一个策略相关联，该策略由服务器在客户请求服务时发送给客户。当方案逆转时不清楚由服务器给客户提交证书的目的。一种可能是为与服务器一般交互反应的目的建立客户信任。另一种可能是

建立为专门鼓励客户公开他们的证书的信任。在后一种场合，可以使用该方法使客户在给服务器公开任何它自己的证书前从服务器请求证书。然而，对服务器来说然后不可能在公开它自己的证书前请求客户证书。这样做将引入循环依赖，使协商进入死锁，因为在这一模型中所有客户证书由同一策略支配，因此，任何后继的服务器请求将导向从客户来的同样的请求。

发明内容

没有一个现有解决方案明确建议使用证书作为控制证书公开的基础。没有提到在证书之间的相互依赖的问题和为不同的策略需要不同证书访问策略以避免某种死锁的需要。这些是在陌生方之间自动信任建立的方面，他们要把在过去被检查的证书保持为不公开的。

先前也没有提到在建立信任期间动态综合证书请求。现有解决方案从预先存在的策略中选择证书请求内容。

本发明的一个目的是提供在保护其证书的陌生的数据处理装置之间完全自动的信任协商。可以立即应用简单的协商策略。也可以考虑平衡成功协商的关心和避免非故意公开关于保存的证书的信息的更复杂的技术。

根据本发明的第一方面，提供一个用于客户/服务器网络的数据处理装置，其中，一个客户数据处理装置给服务器数据处理装置发送一个数据处理请求，服务器数据处理装置根据该请求执行数据处理，并把回答返回到客户数据处理装置，该数据处理装置包括：为存储多个本地站点证书的存储设备；用于从对方站点数据处理装置接收第一证书请求的设备，由第一证书请求请求的证书是存储在存储设备中的本地站点证书，它满足用第一证书请求提供的第一逻辑表达式；而给对方站点数据处理装置发送一个第二证书请求的设备，该第二证书请求依赖于第一证书请求的内容，由第二证书请求请求的证书是对方站点证书，它满足用第二证书请求提供的第二逻辑表达式。

根据第二方面，本发明提供一种操作第一方面的数据处理装置的方法，包括以下步骤：从对方站点数据处理装置接收一个第一证书请求，由第一证书请求请求的证书是存储在存储设备中的本地站点证书，它满足用第一证书请求提供的第一逻辑表达式；给对方站点数据处理装置发送一个第二证书请求，它依赖于第一证书请求的内容，由第二

证书请求请求的证书是对方站点证书，其满足用第二证书请求提供的第二逻辑表达式。

根据第三方面，本发明提供存储在计算机可读存储介质上的计算机程序产品，用于当运行在计算机上时执行第二方面的方法步骤。

根据第四方面，本发明提供一种在载波中实施的计算机数据信号，该信号具有程序元素，用于指示计算机执行第二方面的方法步骤。

这样，本发明扩展现有技术的数字证书方法以支持为证书公开的互相依赖的请求序列。为允许为证书公开的相互依赖的请求序列，不同的证书必须由不同的策略控制。客户从服务器接收到的证书请求不是用于单个的证书，或者甚至不用于一个证书的特定组合。相反，它是为满足一个逻辑表达式的任意证书。在本发明中，到来的证书请求与由客户实际拥有的证书逻辑组合，连同与这些证书的每一个关联的访问控制策略，导出为对方站点证书的一个新请求。这样，本发明包括任何从本地证书访问策略和一个到来的证书请求导出一个应答的证书请求，除了在应答请求独立于到来的请求的场合。并且在后者的情况下，即例外的情况，因为循环依赖，增量证书公开序列是不可能的，如上所述。

由本发明提供的一个重要的优点是它能使信任自动建立，甚至当所涉及的各方在公开他们的证书给对方前需要他们的对手某些知识时。在现有解决方案中，每一参加者只有一次机会在每一次协商中提交证书，并且一个参加者必须首先进行。不像现有技术解决方案，本发明不需要协商参加者一次公开它所有的证书，也不需要关于其他参加者的任何知识。为得到敏感的服务，客户可能必须提交高度敏感的证书，它只在首先得到一个中度敏感的服务器证书后才公开。为这一点，服务器依次需要某些较不敏感的证书。

本发明使得能够协商任意长度的从属证书交换序列。在一些场合，这样的序列能够协商比单一交换更高度的信任。这使得这一新解决方案在电子商务的意义上在陌生方之间可能非常重要，在电子商务中自动商务协商将需要高度信任，参加者相当忠实地讨价还价和适当处理公开的信息。

本发明提供为增量证书公开的自动协商的基础。它通过根据对方站点证书给每一个在一个本地站点保存的证书关联一个访问策略和通

过提供该策略与到来的证书请求的逻辑组合来导出协商响应。

附图说明

通过下面提供的本发明的一个优选实施例的详细说明，结合附图可以更好地理解本发明：

图 1 是一个方框图，表示根据本发明的一个优选实施例的软件部件；

图 2 是一个流程图，表示根据本发明的一个优选实施例由一个本地站点执行的处理步骤；

图 3 是一个示例定时示意图，表示根据本发明的一个优选实施例的请求和回答序列。

发明详述

在本发明的优选实施例中，多个数据处理单元通过一个数据通信网络彼此通信。在图 1 中，表示出一个本地站点 10 通过网络（未示出）与一个对方站点 11 通信，这两个站点都是本优选实施例中的数据处理单元（在另一个实施例中，它们可以是在同一数据处理单元上运行的单独的过程）。遵照上面讨论的第二现有技术方法（Ching，等人，Winslett，等人），一个数据处理单元（亦即协商参加者）由一个安全代理 101 在信任协商中代表，如图 1 所示。每一个协商参加者可以接收一个证书请求（到来的证书请求 21，图 1）。（每一证书请求取证书公式的形式，如在上述第二现有技术方法中一样）。这一请求是为公开一个本地站点证书给对方站点。这一公开的目的可以是解锁服务，或解锁为进一步进行信任协商所需要的对方站点证书的公开。在进一步协商中的一个立即的问题是决定本地站点 10 是否对对方站点 11 具有足够的信任而公开所请求的证书，如果没有，则建立一个对对方站点证书的请求（发出证书请求 22），使得能够建立信任。

如图 1 所示，每一站点以它自己的每一个证书 103 与一个证书访问策略 102 相关。该访问策略识别解锁本地站点证书公开的对方站点证书。当接收到一个证书请求 21 时，安全代理 101 决定什么行动是合适的。该决定在下面的段落中结合图 1 中的结构图和在图 2 中的流程图中的步骤（31-37）说明。

安全代理的行动在步骤 31 开始，此时它从对方站点接收一个形式为逻辑表达式的证书请求 21。如果本地站点在步骤 32 发现它不能处

理满足该请求的证书（以及如果需要某种响应的话，如同当安全代理属于一个服务器的话），则可以在步骤 33 发送拒绝。否则，安全代理必须在步骤 34 决定是否在对方站点已经建立足够的信任以证明提供满足该请求的证书的组合的合法性。具体说，伴随请求 21 或者已经在本地高速缓冲存储 104 的对方站点证书 23 可以满足支配本地证书 103 的访问策略 102，其依次满足到来的请求 21。在这种情况下，以这一方式解锁的并满足到来的请求 21 的本地证书的组合可以立即被送往 26 对方站点 11，如步骤 35 所示。当作为对先前由本地站点的请求的响应接收到当前到来的请求时先前的请求可以结合证书发送在步骤 35 重复，假定这些证书现在将产生解锁履行早先请求的信任。

另一方面，步骤 34 可以确定本地可用的对方站点证书（23 和/或 104）不足以解锁满足到来的请求 21 的本地拥有的证书 103 的组合。在该种情况下，在步骤 36，安全代理 101 通过逻辑组合到来的证书请求 21 与本地站点证书访问策略 102，从对方站点 11 导出发出的为进一步的证书请求 22。这些对方站点证书为解锁由对方站点请求的本地站点证书的目的而请求。于是，为避免不必要地请求证书，导出处理简化了该请求，通过考虑本地站点实际拥有哪一个请求的证书 103 和此外通过不请求附加对方站点证书来解锁已经由积累的 104 和/或到来的 23 对方站点证书解锁的本地证书。

在步骤 37，当安全代理 101 发送进一步的对方站点证书请求以解锁本地站点时，可以在同一时间选择提供一些已经解锁的本地站点。例如，可以提供某些在到来的证书请求 21 中提到的本地站点证书。虽然有不必要地证书公开的风险，但是这一协商策略决策可以增加协商进行的可能性和速度。通过增加对方站点立即提供它请求的证书的机会和通过减少对方站点推断在两个站点结合的证书访问策略中因而关于该协商存在有循环依赖的机会，而实现这一点。

如图 1 所示，当本地站点是一个服务器，而对方站点是一个客户时，从客户来到服务器的消息的内容可以包括服务请求 24。这种服务请求是证书公开在客户和服务器之间协商的典型的启动消息。在接收到服务请求 24 后，服务器的安全代理 101 应用服务支配策略（图 1 中未示出）以决定伴随该请求的对方站点证书 23 是否足以满足该服务支配策略（这对上面讨论的第二现有技术方法也是真的）。虽然一些服

务器是无态的，因此在每一客户请求后不保持客户证书，但是另外的像客户，可能高速缓冲存储对方站点的证书 104。这种服务器使用高速缓冲存储的对方站点证书 104，以及伴随该请求的那些证书 23，试图满足它们的服务支配策略。不管是否使用高速缓冲存储的对方站点证书 104，当满足该服务支配策略时，服务被授权 25。否则，安全代理 101 以发出的证书请求 22 的形式返回服务支配策略。然后在客户和服务器之间协商证书公开，并且如果成功，则客户可以用所附的足够的证书重复该服务请求以授权服务。这种交换的一个例子示于图 3。

在图 3 中，在阶段 1，客户给服务器站点发送一个特殊服务请求，请求服务器代表该客户执行一个特别的处理任务（例如读访问一个数据库）。在该请求上不附任何证书，假定因为该客户不知道支配该服务的证书策略。在阶段 2，服务器安全代理给客户安全代理发送服务支配策略，它通知客户安全代理关于证书的选项以提交产生对服务器安全代理一方足够的信任使所请求的服务被执行。这一策略组成一个证书请求，并当它由客户安全代理在阶段 3 接收到时被这样处理。客户安全代理根据上面讨论的步骤结合图 3 的流程图响应。

也就是说，客户安全代理接收证书请求（步骤 31）（授权在阶段 1 请求的服务）。决定该客户拥有满足该请求的证书的至少一个组合（步骤 32）。决定本地没有足够的服务器证书可用，以满足任何那些满意的组合的构成证书的访问控制策略（步骤 34）。因此导出一个新的请求（步骤 36），设计用以解锁它自己的证书的这种满意的组合，它将依次解锁希望的服务。最后，它发送该请求给服务器不附任何证书。（在该例子的一个变体中，客户可以在这一点在发出的请求上附一些由服务器在阶段 2 请求的证书，例如，如果它们的访问控制策略允许它们被公开而不需要先前的服务器知识的话。）

阶段 4 在服务器的安全代理接收到由客户在阶段 3 的末尾发送的证书请求时开始。该请求再次如上述讨论结合图 2 的流程图处理。服务器安全代理决定，有满足客户安全代理的请求的证书（步骤 32）和它们完全不为给客户安全代理公开而解锁（步骤 34）（服务器安全代理尚未从客户安全代理接收到任何证书）。导出一个客户证书请求（步骤 36），意图解锁由客户安全代理请求的证书，并将其发送（步骤 37）给客户安全代理，连同客户安全代理请求的一些证书，其访问控制策

略允许它们被公开而无需首先看任何客户证书。

阶段 5 在客户安全代理接收到请求和由服务器安全代理在阶段 4 的末尾发送的证书时开始。客户安全代理决定它有满足该请求的证书（步骤 32），但是没有由其访问控制策略由迄今接收到的服务器证书解锁（步骤 34）的证书组成的满意的组合。在解锁更多它自己的证书的努力中，客户安全代理然后导出一个新服务器证书请求如下（步骤 36）。它通过置换对不拥有固定错误的证书的参考修改到来的证书请求。然后它用为该证书的访问策略置换确实拥有的证书的每一剩余的出现。结果公式，像那些访问策略，根据对方站点证书表达。下一个相连接的是与由客户先前在阶段 3 的末尾给服务器发送的证书请求，其可能尚未完全由从服务器接收的证书满足。安全代理简化了结果的结合，以避免不必要地请求证书。通过从公式中消除客户安全代理已经接收到的和满足在公式中表达的属性限制的证书的每一次出现而实现简化。结果公式被简化，作为常数 true 处理消除的证书，并相应简化逻辑连接。（在一个连接中的 true 的出现被简单消除；一个空连接被 true 置换；一个包含 true 的拆卸连接由 true 置换。）最后，客户安全代理发送导出的并以这一方式简化的公式，也发送任何在到来的请求中请求的证书，其访问控制策略由在该阶段开始时接收到的服务器证书解锁。

阶段 6 在服务器的安全代理接收到请求和在阶段 5 的末尾由客户安全代理给它发送的证书时开始。服务器安全代理决定它有满足该请求的证书（步骤 32），但是它们并非所有都被解锁（步骤 34）。然后通过和在阶段 5 的客户侧说明的许多相同的步骤导出一个新的客户证书请求（步骤 36）。在服务器侧与在客户侧导出的主要差别在于，在该例中说明的协商策略中，服务器不重使用它先前的客户证书请求。

阶段 7 在客户的安全代理接收到请求和在阶段 6 的末尾由服务器安全代理发送的证书时开始。客户安全代理决定，它有满足该请求的证书的至少一个组合（步骤 32）。该客户现在接收到足够的服务器证书以解锁这种证书组合（步骤 34）。于是它在阶段 7 的末尾给服务器发送一个这样的组合，连同在阶段 5 的末尾发送给服务器的同样的证书请求。

阶段 8 在服务器的安全代理接收到该重复请求和在阶段 7 的末尾

由客户发送给它的证书时开始。服务器安全代理决定它有满足该请求的证书（步骤 32）和它们未被解锁（步骤 34），因为它们的访问控制策略由客户在阶段 7 的末尾发送的客户证书满足。然后它发送这些证书给该客户。

阶段 9 在客户的安全代理接收到由服务器在阶段 8 的末尾发送的证书时开始。这些证书，连同由客户安全代理在阶段 5 和 7 的开始时接收到的并由该客户安全代理从那时高速缓冲存储的证书满足客户证书的访问控制策略，所述客户证书满足由客户安全代理在阶段 3 接收的服务支配策略。客户安全代理然后发送解锁的证书的一个组合，这些证书一起满足服务支配策略。还重复原始的服务请求。

阶段 10 在服务器的安全代理接收到服务请求和在阶段 9 的末尾由客户发送的证书时开始。这些证书满足服务支配策略，所以该服务被授权，执行，并返回它的结果。客户在阶段 11 接收到它，以此结束该例。

协商示例

这里介绍的例子说明在图 3 中原理表示的假设协商的 11 个步骤。意在说明公式的操作，其由本发明的优选实施例规定。非正式表示出这些公式。本例仅为说明，不打算精确表征任何实际的协商、证书、或策略。

假设的证书

每一证书的条目以用于为该证书在该例的其它部分使用的缩写开始。

安全-实践证书 – 由服务器和客户两者保存的

我们假定安全-实践-标准顾问给他们鉴定其安全基础设施等级的实体发布安全-实践证书。等级可以由第三方使用来评估给该实体提供的信息由该实体非故意公开的可能性。为使分等级的实体演示它们满足一定等级的需求同时保持他们不能满足的等级不被人知，为每一等级发布一个单独的证书，使用一个称为“passed（通过）”的属性，它在为该等级的需求被满足时为 *true*（真），否则为 *false*（假）。

在我们的例子中，有 4 个安全实践等级，低、中等、高、和非常高。服务器满足的最大等级是“高”，由客户满足的最大等级是“中等”。每一实体保护为它不能达到的等级具有的证书。如果为它通过

的等级的证书不保护的话，则在保护中的差别使哪一个等级它不能达到变得很明显。所以为所有等级（除了最低的）的证书都被保护，而不管拥有者通过还是没通过。

Very High（非常高）安全-实践等级。高敏感性。

High（高）安全-实践等级。中等敏感性。

Med（中等）安全-实践等级。低敏感性。

Low（低）安全-实践等级。无敏感性。

客户证书

Contract（合同）目的地合同。由期望交付商品的一方发布。极端敏感。需要保证该信息不能泄露给竞争者。

Credit（信用）信用证。由信用者给拥有者发布。中-高度敏感性。

Dock（码头）在发起码头处的仓库协定。由码头管理发布。中等敏感性。需要避免散布给竞争者。

S-Receipt（收据）先前运输收据。由过去运输商品的运输者发布。在该例中的客户没有任何先前的运输收据。

Account（帐户）以服务者/运输者建立的帐户。由运输者发布。在该例中客户没有以服务者/运输者建立的帐户。

B-Org（组织）商务组织成员关系证书。由一个商务组织发布，诸如国际商会。不敏感。

服务器证书

Receipt（收据）先前交付收据。由在过去运送的商品的拥有者发布。不敏感。

Bond（债券）债券证书。由债券代理机构发布。不敏感。

Ref（参考）来自制造商的参考。由希望根据先前的商务经验建议运输者的制造商发布。低敏感性。在该例中的服务器具有至少两个不同的制造商。

B-Org（组织）商务组织成员关系证书。由一个商务组织发布，诸如国际商会。不敏感。

假设策略

支配一个客户或者服务器证书 X 的策略分别用 X_{client} 或 X_{server} 指定。这里表达的策略不完全。特别是，它们不表达支持证书的作为基本的需求。虽然不完全完整或者现实，但是由“when”引入的子句说

明证书属性的使用或限制。例如，客户为它的目的地合同证书的访问控制策略需要从两个不同的证书发布者来的参考。

客户的证书-支配策略

$Contract_{client}$ = High AND Ref_1 AND Ref_2 AND (Bond OR ($Receipt_1$ AND $Receipt_2$)) 这里 High.passed = *true* AND Ref_1.issuer ≠ Ref_2.issuer

AND $Receipt_1$.issuer ≠ $Receipt_2$.issuer

$Credit_{client}$ = Med AND Ref_1 AND Ref_2 这里 Med.passed = *true* AND Ref_1.issuer ≠ Ref_2.issuer

$Dock_{client}$ = Med AND (Bond OR ($Receipt_1$ AND $Receipt_2$)) 这里 Med.passed =

true AND $Receipt_1$.issuer ≠ $Receipt_2$.issuer

$Very\ High_{client}$ = High AND B-Org 这里 High.passed = *true*

$High_{client}$ = Med AND B-Org 这里 Med.passed = *true*

Med_{client} = Low AND B-Org 这里 Low.passed = *true*

Low_{client} = No Credential Required

服务器的证书支配策略

$Receipt_{server}$ = No Credential Required

$Bond_{server}$ = No Credential Required

Ref_{server} = Low 这里 Low.passed = *true*

$Very\ High_{server}$ =High AND B-Org 这里 High.passed = *true*

$High_{server}$ = Med AND B-Org 这里 Med.passed = *true*

Med_{server} = Low AND B-Org 这里 Low.passed = *true*

Low_{server} = No Credential Required

服务器为安排运输日程的服务-支配策略

Account OR (Dock AND ((S-$Receipt_1$ AND S-$receipt_2$) OR Contract) AND

Credit) 这里 S-$Receipt_1$.issuer ≠ S-$Receipt_2$.issuer

协商步骤

该例是一个成功的协商。证书请求在阶段 2 到 7 的末尾发送。需要由服务器在阶段 2 的末尾请求的客户证书来授权服务。协商的剩余

用以为客户对服务器公开那些证书建立足够的信任。需要在阶段 3 到 7 请求的证书解锁对依次为成功协商需要的证书的访问。

在一个站点接收一个证书请求的每一阶段中，该站点拥有满足该请求的证书，并且图 2 的步骤 32 成功。在每一这种阶段的末尾，有效的安全代理给对方站点发送在到来的请求中请求的所有证书，其访问策略通过可本地应用的对方站点证书解锁。这些在下面加以标签“解锁的，请求的证书”。在由这一例子说明的协商策略中，客户的安全代理高速缓冲存储和重复先前的请求，而服务器的安全代理不这样做。

阶段 1 – 客户发送服务请求：安排运输日程

阶段 2 – 服务器接收请求，返回支配服务的策略，服务器需要建立信任，即客户真的在为运输服务的市场上，并可以为他们付费。发送为安排上面表示的运输的日程的支配服务的策略。

阶段 3 – 客户接收证书请求以授权服务

服务器证书本地可用：没有

解锁的，请求的证书：没有（步骤 34 失败）

到来的请求，通过消除客户不拥有的证书简化：

Dock AND Contract AND Credit

这一公式，每一证书由其（括起来的）访问策略代替：

(Med AND (Bond OR ($Receipt_1$ AND $Receipt_2$)) 式中 Med.Passed = *true* AND

$Receipt_1$.issuer ≠ $Receipt_2$.issuer) AND

(High AND Ref_1 AND Ref_2 AND (Bond OR ($Receipt_1$ AND $Receipt_2$)) 式中

High.passed = *true* AND Ref_1.issuer ≠ Ref_2.issuer AND $Receipt_1$.issuer

≠ $Receipt_2$.issuer) AND

(Med AND Ref_1 AND Ref_2 式中 Med.Passed = *true* AND Ref_1.issuer ≠

Ref_2.issuer)

该公式，简化后，作为证书请求在阶段 3 的末尾发送到服务器：

High AND Ref_1 AND Ref_2 AND (Bond OR ($Receipt_1$ AND $Receipt_2$)) AND Med

式中 High.passed = *true* AND Ref_1.issuer ≠ Ref_2.issuer AND $Receipt_1$.issuer ≠ $Receipt_2$.issuer AND Med.Passed = *true*

阶段 4 – 服务接收证书请求

本地可用客户证书: 无

解锁的，请求的证书: Bond（步骤 34 失败）

到来的请求，每一证书由其（括起来的）访问策略代替:

(Med AND B-Org 式中 Med.Passed = *true*) AND

(Low 这里 Low.passed = *true*) AND

(Low 这里 Low.passed = *true*) AND

((No Credential Required) OR ((No Credential Required) AND (No Credential Required))) AND

(Low AND B-Org 这里 Low.passed = *true*)

这一公式，简化后，在阶段 4 的末尾发送给客户:

Med AND Low AND B-Org 这里 Med.passed = *true* AND Low.passed = *true*

阶段 5 – 客户接收证书请求和一个服务器证书

服务器证书本地可用: Bond

解锁的，请求的证书: Low，B-Org （步骤 34 失败）

到来的请求，每一证书由其（括起来的）访问策略代替:

(Low AND B-Org 这里 Low.passed = *true*) AND

(No Credential Required) AND (No Credential Required)

这一公式，连同请求在阶段 3 的末尾发送给服务器:

((Low AND B-Org 这里 Low.passed = *true*) AND (No Credential Required)

AND (No Credential Required)) AND

(High AND Ref_1 AND Ref_2 AND (Bond OR ($Receipt_1$ AND $Receipt_2$)) AND Med

式中 High.passed = *true* AND Ref_1.issuer ≠ Ref_2.issuer AND $Receipt_1$.issuer ≠ $Receipt_2$.issuer AND Med.Passed = *true*)

这一请求，通过消除本地可用的服务器证书简化，在阶段 5 的末尾发送给服务器:

Low AND B-Org AND High AND Ref_1 AND Ref_2 AND Med 式中 Low.passed =

true AND High.passed = *true* AND Ref_1.issuer ≠ Ref_2.issuer AND

Med.Passed = *true*

阶段 6 – 服务器接收证书请求和两个客户证书

客户证书本地可用: Low, B-Org

解锁的，请求的证书: Low, B-Org, Med (步骤 34 失败)

到来的请求，每一证书由其（括起来的）访问策略代替:

(No Credential Required) AND

(No Credential Required) AND

(Med AND B=ORG 这里 Med.passed = *true*) AND

(Low 这里 Low.passed = *true*) AND

(Low 这里 Low.passed = *true*) AND

(Low AND B-Org 这里 Low .passed = *true*)

这一请求，通过消除本地可用的客户证书简化，在阶段 6 的末尾发送给客户: Med 这里 Med.passed = true

阶段 7 – 客户接收证书请求和 3 个服务器证书

服务器证书本地可用: Bond (从阶段 5 高速缓冲存储), Low, B-Org, Med

解锁的，请求的证书: Med (步骤 34 成功)

请求在阶段 5 的末尾发送给服务器（见步骤 35）:

Low AND B-Org AND High AND Ref_1 AND Ref_2 AND Med 这里 Low.passed =

true AND High.passed = true AND Ref_1.issuer ≠ Ref_2.issuer AND

Med.passed = *true*

该请求，通过消除本地可用的服务器证书简化，在阶段 7 的末尾发送给服务器:

High AND Ref_1 AND Ref_2 这里 High.passed = *true* AND Ref_1.issuer ≠

Ref_2.issuer

阶段 8 – 服务器接收证书请求和一个客户证书

客户证书本地可用：Low，B-Org，（两者都从阶段 6 高速缓冲存储），Med

解锁的，请求的证书：High，Ref_1，Ref_2（步骤 34 成功，证书在步骤 35 发送）

阶段 9 – 客户接收 3 个服务器证书，它们结束解锁将授权服务的客户证书

因为没有接收到另外的证书请求，所以相关的请求（步骤 31）再次成为在阶段 3 的开始接收到的支配服务的策略。

服务器证书本地可用：Bond （在阶段 5 接收到），Low，B-Org，Med（在阶段 7 接收到），High，Ref_1，Ref_2（在阶段 9 接收到）

解锁的，请求的证书：Dock，Contract，Credit（步骤 34 成功）

服务请求，运输日程安排，在阶段 1 首先发送，现在以所附请求的证书重复。

阶段 10 – 服务器接收服务请求和 3 个所附的证书，授权服务

服务器证实所附证书满足服务支配策略并授权请求的服务。该服务的结果在阶段 10 的末尾返回。

阶段 11 – 客户接收他请求的服务。

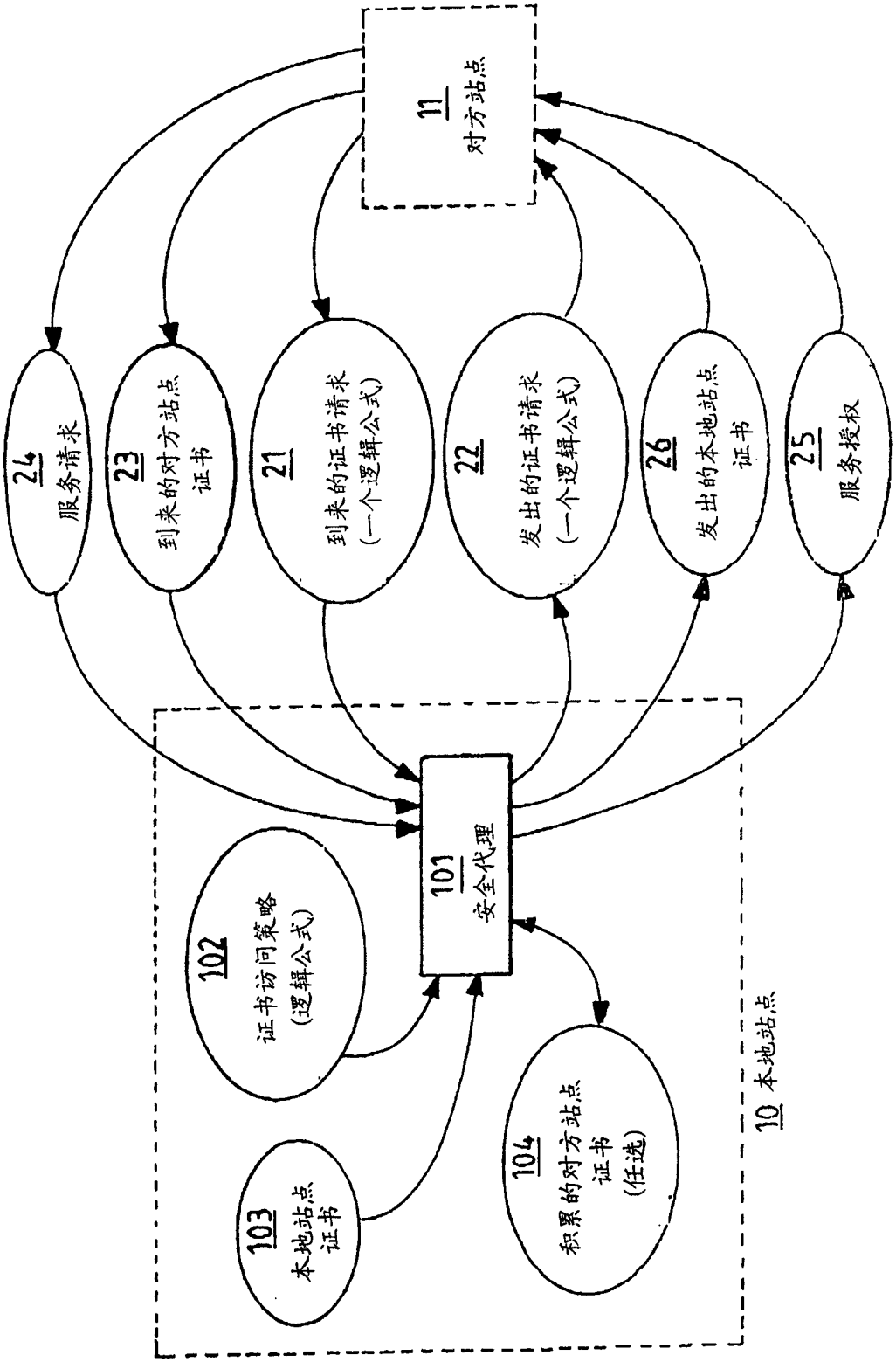

11 对方站点
24 服务请求
23 到来的对方站点证书
21 到来的证书请求(一个逻辑公式)
22 发出的证书请求(一个逻辑公式)
26 发出的本地站点证书
25 服务授权
101 安全代理
102 证书访问策略(逻辑公式)
103 本地站点证书
104 积累的对方站点证书(任选)
10 本地站点

图 1

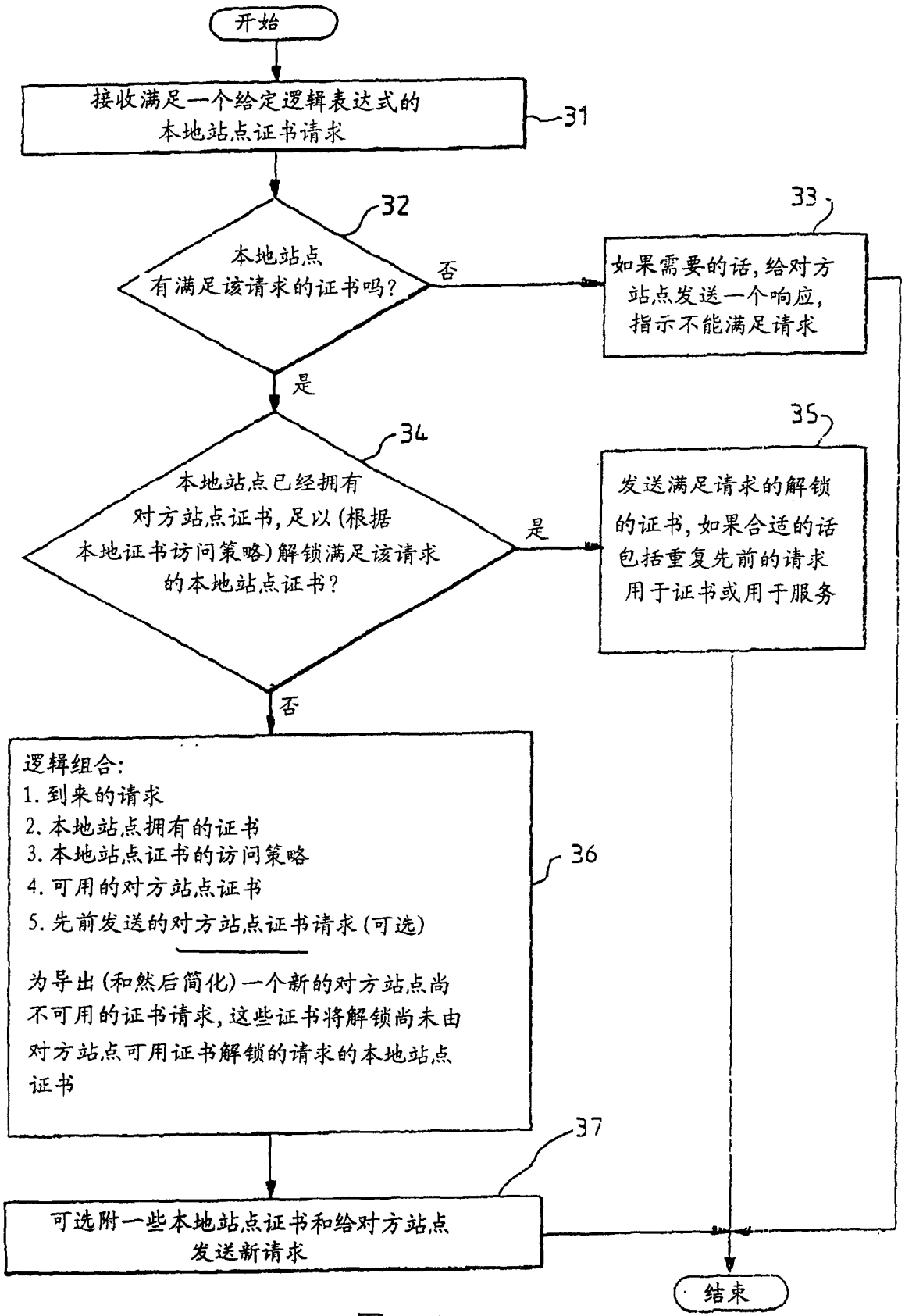

开始
接收满足一个给定逻辑表达式的本地站点证书请求
31
32
本地站点有满足该请求的证书吗?
否
33
如果需要的话,给对方站点发送一个响应,指示不能满足请求
是
34
本地站点已经拥有对方站点证书,足以(根据本地证书访问策略)解锁满足该请求的本地站点证书?
是
35
发送满足请求的解锁的证书,如果合适的话包括重复先前的请求用于证书或用于服务
否
逻辑组合:
1. 到来的请求
2. 本地站点拥有的证书
3. 本地站点证书的访问策略
4. 可用的对方站点证书
5. 先前发送的对方站点证书请求(可选)
为导出(和然后简化)一个新的对方站点尚不可用的证书请求,这些证书将解锁尚未由对方站点可用证书解锁的请求的本地站点证书
36
37
可选附一些本地站点证书和给对方站点发送新请求
结束

图 2

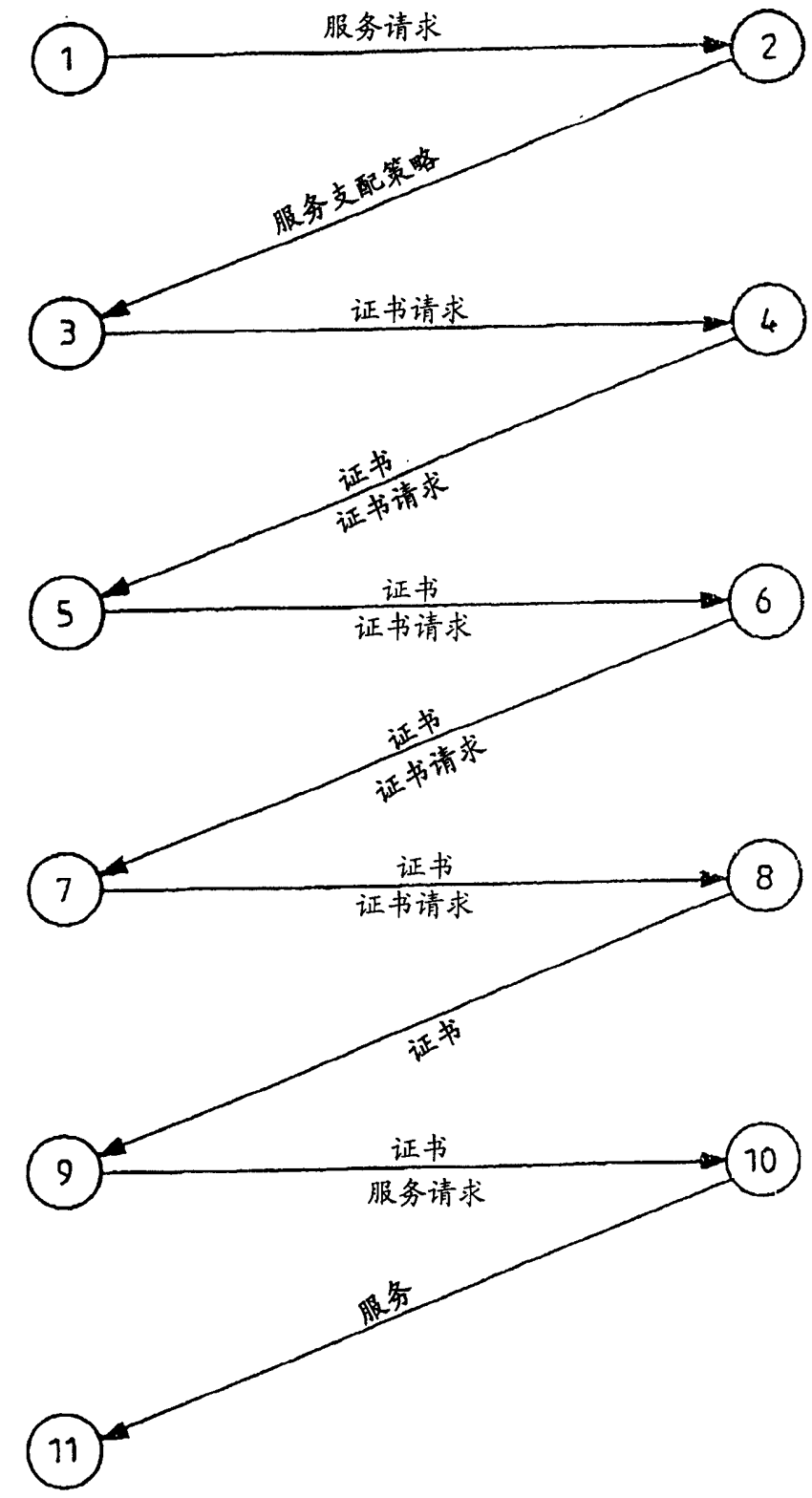

1
服务请求
2
服务支配策略
3
证书请求
4
证书
证书请求
5
证书
证书请求
6
证书
证书请求
7
证书
证书请求
8
证书
9
证书
服务请求
10
服务
11
客户站点
服务器站点

图 3