(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 102647431 B
(45)授权公告日 2016.07.06

(21)申请号 201210144490.3
(22)申请日 2008.11.06
(30)优先权数据
11/935,783 2007.11.06 US
(62)分案原申请数据
200810170453.3 2008.11.06
(73)专利权人 英特尔公司
地址 美国加利福尼亚
(72)发明人 龙门 J·沃克 D·德拉姆
M·米利埃 K·格雷瓦尔 P·德万
U·萨瓦冈卡尔 S·D·威廉斯
(74)专利代理机构 永新专利商标代理有限公司 72002
代理人 刘瑜 王英
(51)Int.Cl.
H04L 29/06(2006.01)
H04L 9/32(2006.01)
(56)对比文件
CN 1697373 A,2005.11.16,
CN 101043335 A,2007.09.26,
CN 101335621 A,2008.12.31,
WO 2007/108651A1 A1,2007.09.27,
审查员 郭婧

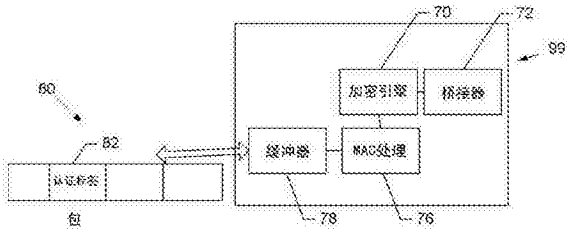
权利要求书2页 说明书5页 附图4页

(54)发明名称

具有业务流可见性的端到端的网络安全

(57)摘要

本发明公开了通过使用两个密钥的组合模式的单通加密和认证而获得客户端与服务器之间端到端的安全性以及业务流对中间网络设备的可见性。在各个实施例中,组合的加密-认证单元包括加密单元以及与所述加密单元并行耦合的认证单元;并且与使用加密密钥生成所述密文并行地使用认证密钥来生成认证标签,其中所述认证和加密密钥具有不同的密钥值。在各个实施例中,所述加密单元在AES计数器模式下工作,所述认证单元在AES-GMAC模式下并行地工作。使用两个密钥的、单通的组合模式的算法只使用有限数量的硬件门以保持网络的性能,同时允许中间设备访问所述加密密钥来对数据进行解密,而不需要向该设备提供使在端到端的设备之间保持的数据完整性受损的能力。



CN 102647431 B

1.一种用于检查包的装置,所述装置包括:

缓冲器,其用于中继在网络的客户端和服务器之间传输的多个网络包,每个包已由一个客户端或一个服务器使用对应于客户端-服务器对的加密密钥和认证密钥组单通地进行了加密并且关联了认证标签,所述加密密钥和认证密钥具有不同的密钥值;以及

处理单元,其耦合到所述缓冲器,用于使用相对应的加密密钥来对所述多个网络包中的一个或多个网络包进行解密和检查,所述处理单元就所述加密密钥和认证密钥方面而言只具有所述加密密钥。

2.根据权利要求1所述的装置,其中,所述处理单元还用于派生所述加密密钥。

3.根据权利要求1所述的装置,还包括加密引擎,所述加密引擎包括:

加密块,其用于对每个包进行加密以生成相应的加密包,其中,所述加密包每个均包括使用所述加密密钥连续生成的多个密文块;以及

认证块,其耦合到所述加密块,并用于基于所述认证密钥和连续生成的密文块来生成所述认证标签。

4.根据权利要求3所述的装置,其中:

所述加密块基于高级加密标准AES-计数器模式进行工作;以及

所述认证块并行地基于AES-伽罗瓦消息认证码GMAC模式进行工作。

5.根据权利要求4所述的装置,其中,所述认证块包括多个布尔函数块,其中,所述多个布尔函数块中的每一个用于对密文块和第一有限字段乘法单元的输出执行布尔运算,并生成第二有限字段乘法单元的输入,并且所述第一有限字段乘法单元和所述第二有限字段乘法单元采用被定义为AES(所述认证密钥,0^{128})的保密乘法器。

6.根据权利要求3所述的装置,其中,所述加密块和所述认证块共同位于所述客户端或所述服务器的一片集成电路上。

7.根据权利要求1所述的装置,其中,所述装置是芯片组。

8.一种用于检查包的系统,包括:

网络接口卡,其包括:

缓冲器,其用于中继在网络的客户端和服务器之间传输的多个网络包,每个包已由一个客户端或一个服务器使用对应于客户端-服务器对的加密密钥和认证密钥组单通地进行了加密并且关联了认证标签,所述加密密钥和认证密钥具有不同的密钥值;以及

处理单元,其耦合到所述缓冲器,用于使用相对应的加密密钥来对所述多个网络包中的一个或多个网络包进行解密和检查,所述处理单元就所述加密密钥和认证密钥方面而言只具有所述加密密钥。

9.根据权利要求8所述的系统,其中,所述处理单元还用于派生所述加密密钥。

10.根据权利要求8所述的系统,其中,所述网络接口卡还包括加密引擎,所述加密引擎包括:

加密块,其用于对每个包进行加密以生成相应的加密包,其中,所述加密包每个均包括使用所述加密密钥连续生成的多个密文块;以及

认证块,其耦合到所述加密块,并用于基于所述认证密钥和连续生成的密文块来生成所述认证标签。

11.根据权利要求10所述的系统,其中:

所述加密块基于高级加密标准AES-计数器模式进行工作;以及

所述认证块并行地基于AES-伽罗瓦消息认证码GMAC模式进行工作。

12.根据权利要求11所述的系统,其中,所述认证块包括多个布尔函数块,其中,所述多个布尔函数块中的每一个用于对密文块和第一有限字段乘法单元的输出执行布尔运算,并生成第二有限字段乘法单元的输入,并且所述第一有限字段乘法单元和所述第二有限字段乘法单元采用被定义为AES(所述认证密钥,0^{128})的保密乘法器。

13.根据权利要求10所述的系统,其中,所述加密块和所述认证块共同位于所述客户端或所述服务器的一片集成电路上。

具有业务流可见性的端到端的网络安全

[0001] 本申请是申请日为2008年11月6日、申请号为200810170453.3的同名专利申请的分案申请。

技术领域

[0002] 本申请涉及通信和网络领域，具体地涉及维护客户端与服务器之间的端到端的安全性并同时允许业务流(traffic)对中间网络设备具有可见性。

背景技术

[0003] 许多网络安全协议依赖于在客户端和服务器之间使用昂贵的非对称加密技术的协商会话密钥，并随后需要服务器记录大量的针对每个客户端会话协商的对称密钥。端到端的安全性是指在网络中从一端一直到另一端(例如从客户端到服务器以及从服务器到客户端)的通信的数据可靠性和/或数据机密性。业务流可见性是指中间的服务器以及信息技术(IT)监测设备可以查看受保护的业务流。在一定程度上，这两个目地是相互对立的，但是在可控制的环境中，出于网络安全的目的，这两者都很重要，在所述环境中，经授权的中间设备需要访问数据来执行有价值的网络功能，例如对病毒/蠕虫进行安全扫描。

[0004] 对于防止第三方对客户端和服务器之间的业务流进行篡改来说，端到端的安全对于客户端和服务器双方都是很重要，其中客户端更易于受到直接的操纵或篡改。因此，客户端保密(密码密钥)的独特性对于防止由于一个客户端的泄密而获得对于其他客户端的业务流的访问来说是极为重要的。业务流可见性对于IT管理来说是至关重要的，其要求IT管理设备观测业务流以检测异常现象。许多当前的主要安全协议只提供了端到端的安全性，而并未涉及业务流的可见性。

[0005] 近来，为了提升效率，业界已经转向对包加密和认证均采用单密钥组合模式加密(例如AES-GCM和AES-CCM)。因此，就端到端的安全性而言，潜在地具有单密钥的中间网络设备可能在网络业务流的可靠性方面会出现安全问题。换言之，成功地攻入中间网络设备的攻击者可以随意地篡改能够被客户端和服务器端点接受的任何合法的包。[AES=高级加密标准(Advanced Encryption Standard);GCM=伽罗瓦计数器模式(Galois Counter Mode);CCM=计数器CBC-MAC(Counter CBC-MAC);以及CBC-MAC=密码块链消息认证码(Cipher Block Chain Message Authentication Code)。]

附图说明

[0006] 下面将通过示例性实施例结合附图以非限制性的方式描述本发明的实施例，在附图中，相似的附图标记表示相似的元件，其中：

[0007] 图1是根据本发明各个实施例的企业网络安全图；

[0008] 图2是根据各个实施例的客户端平台上的序列的图示；

[0009] 图3是根据各个实施例的另一客户端平台序列；

[0010] 图4是根据各个实施例的服务器序列；

[0011] 图5是根据各个实施例的另一序列；

[0012] 图6是根据各个实施例的硬件图示；以及

[0013] 图7是根据各个实施例进一步详细地描述的图6的加密引擎。

具体实施方式

[0014] 本发明的示例性实施例包括但不限于，用于维持客户端和服务器之间的端到端的安全性、并同时允许业务流对中间网络设备具有可见性的方法和装置。中间网络设备不太可能能够伪造或者仿造消息来欺骗客户端和/或服务器，即使它们被入侵者成功攻入也是如此。

[0015] 将使用本领域技术人员通常使用的词句来来描述示例性实施例的各个方面，以将这些示例性实施例的作用的实质传递给本领域的其他技术人员。但是，对于本领域技术人员来说显而易见的是，可以只采用所描述的多个方面中的一些方面来实施替代性的实施例。出于说明的目的，阐述了具体的数量、材料和结构以提供对示例性实施例的透彻理解。但是对本领域技术人员来说显而易见的是，无需这些具体细节也能实施替代性的实施例。在其他情形下，略去或者简化了公知的特征，以免对示例性实施例造成模糊。

[0016] 此外，以最有助于理解示例性实施例的方式将各个操作描述为多个分离的操作；但是，描述的顺序不应该被解释成其意味着这些操作必然地是依赖于其顺序的。具体而言，这些操作无需以本文所呈现的顺序执行。

[0017] 在本文中反复使用了短语“在一个实施例中”。该短语一般并非指的是同一个实施例；但是它也可以这样指代。除非上下文有相反提示，否则术语“包含”、“具有”和“包括”是同义词。短语“A/B”表示“A或B”。短语“A和/或B”表示“(A)、(B)或者(A和B)”。短语“A、B和C中的至少一个”表示“(A)、(B)、(C)、(A和B)、(A和C)、(B和C)或者(A、B和C)”。短语“(A)B”表示“(B)或者(AB)”，即A是任选的。

[0018] 本发明的实施例提供了一种安全协议，其支持客户端和服务器之间的端到端的安全以及业务流对于中间网络设备的可见性，其采用了利用两个密钥的单通(single-pass)组合的加密认证，两个密钥为具有不同密钥值的加密密钥和认证密钥。可以逐帧或逐包地进行基于硬件的线速的端到端的加密和认证。在本申请中，除非上下文有明确的相反提示，术语“帧”和“包”是可以互换的。在各个实施例中，客户端和服务器与域控制器进行通信，该域控制器为每一客户端-服务器关系授予一组加密和认证密钥。一旦接收到加密和认证密钥，客户端和服务器对将所述密钥用于组合的加密-认证以及组合的认证-解密。为了业务流的可见性而同时不损害认证，域控制器还可以将解密密钥(而不是认证密钥)发送给授权的IT网络设备，例如IT监测设备/主机。当授权的IT网络设备具有解密密钥后，授权的IT网络设备便能够以全线速来对加密的经过的业务流进行解密，从而允许了业务流对授权的IT网络设备的可见性。但是，如果没有认证密钥，IT网络设备就无法替换认证，因此无法欺骗客户端和服务器。

[0019] 在各个实施例中，单通双密钥的组合加密-认证机制可以采用一种节省存储空间的派生(derived)密钥机制来实施。于2007年3月20日提交的申请号为11/731,562的题为“具有业务流可见性的端到端的网络安全(End-to-End Network Security with Traffic Visibility)”的美国申请中描述了派生密钥机制的实例。

[0020] 参见图1，企业网络14可以使多个客户端12与一个或多个服务器16进行通信。在所示的实施例中，企业域控制器20可以负责维持整个企业的端到端的安全以及维持业务流对于服务器16和IT监测设备18的可见性。域控制器20可以是例如认证、授权和审计(AAA)服务器、密钥分发服务器、或者策略服务器等。

[0021] 企业域控制器20将加密和认证密钥(如箭头22所示)分发给客户端12和服务器16。另外，企业域控制器20还可以将加密密钥(而非认证密钥)分发给IT网络监测主机18。本文所使用的术语“密钥”包括预派生(pre-derived)或者完全派生(fully derived)的密钥。换言之，如前面所提示，域控制器20可以将完全派生的加密和认证密钥进行分发，或者可以实施节省存储空间的“派生密钥”的机制，并将这些预派生的密钥分发给授权的设备，例如应用服务器和中间IT设备。在各个实施例中，域控制器总是将派生密钥分发给客户端，因为客户端易于遭受攻击而损害任何的预派生密钥。

[0022] 参考图2，描述了应用由域控制器20分发的加密和认证密钥的客户端的序列。使用两个接收到的密钥，即加密密钥和认证密钥，以单通方式对每个外发到企业服务器的帧进行加密和认证。初始，如24处所示，应用数据进入传输控制协议(TCP)/用户数据包协议(UDP)/互联网协议(IP)栈。互联网协议包由该栈分发给服务器。然后链路层驱动程序形成第二层的帧，如26处所示。在菱框28处进行检查，以确定目的地互联网协议地址是否输入企业服务器。如果不是，则如34处所示，通过网络接口卡传输该帧。如果是，则如30处所示，使用存储在硬件中的适当的加密和认证密钥以单通方式来对该帧进行加密和认证。然后，如32处所示，通过网络接口卡传输该加密后的帧。

[0023] 当客户端平台接收到帧时(其显示为包到达网络接口卡)，则在图3中的菱框36处进行检查，以确定该帧是否通过本文所述的协议进行了处理。如果不是，如38处所示，该帧被传输至更高的协议层；如果是，则使用认证密钥对该包进行认证。一旦成功认证，如方框40处所示，则使用存储在硬件中的适当的加密密钥对该包进行解密。然后如42处所示，该帧被传输至更高的协议层。

[0024] 接着，参考图4，当服务器16接收到帧时(其显示为包到达网络接口卡)，在菱框44处进行检查，以确定该帧是否由本文所述协议进行了处理。如果不是，如46处所示，该帧被传输至更高的协议层。如果是，则在48处，经适当认证的密钥(被派生并)被用来对接收到的帧进行认证。一旦通过认证，在方框50处使用加密密钥对该帧进行解密。最后，在52处，该帧被传输至更高的协议层。

[0025] 服务器可以使用图5中示出的序列来传输帧。在56处，在互联网协议栈中接收应用数据。包被分发给各个客户端。链路层驱动程序随后在58处接收帧。在方框60和62处，适当的加密和认证密钥被应用于单通双密钥算法中，来对该包进行加密，以及并行地生成认证标签。最后，在64处，该帧被传输到网络。

[0026] IT网络监测设备18(图1)与服务器12类似地工作。服务器12和监测主机直接或者派生地维持密钥来处理许多来自客户端的不同的安全的关联。因为不同客户端/会话的密钥是不同且无关联的，所以使一个客户端主机泄密的攻击者无法冒充为其他客户端。对于域控制器20、服务器16和监测设备18，由于采用派生密钥机制来实施时密钥的数量相对较小，因此这些密钥可以存储在硬件中，同时仍然提供对于防篡改的适当的保护。

[0027] 在一个实施例中，帧的格式可以捎带确认(piggyback)互联网协议安全(IPSEC)

帧。

[0028] 在一些实施例中，向企业网络提供了端到端的安全性以及业务流的可见性两者。在一些实施例中，该机制可以完全以硬件方式实现，在一些情形下，这样可以较低的成本实现全线速的性能。

[0029] 参见图6，硬件解决方案99包括标记为加密引擎70的组合的加密-认证块，所述加密引擎70耦合到桥接器72和媒体访问控制(MAC)处理单元76。桥接器72可以包括直接存储器存取模块(未示出)。处理单元76通过缓冲器78与到来和外发的包80通信。包80可以包括认证标签(T)82。对于客户端12和服务器16，为处理单元76提供加密和认证密钥，但对于中间网络设备18，为处理单元只提供加密密钥，允许这些设备具有对业务流的可见性，但即使中间设备泄密，其也不太可能伪造或仿造消息来欺骗客户端/服务器。

[0030] 参考图7，进一步详细地说明了单通组合的加密-认证引擎70的实施例。引擎70包括并行地相互耦合的加密块112和认证块114，使得认证块114能够使用认证密钥生成认证标签96，并且加密块112并行地使用加密密钥将明文包加密成具有多个连续生成的密文块的密文。

[0031] 在各个实施例中，加密块112在AES计数器模式下操作，而认证块114在AES-GMAC(伽罗瓦消息认证码)模式下操作。如所示出的，加密块112包括多个计数器92、递增器(incrementor)94、前向(forward)块96和多个布尔函数块98，它们如图所示相互耦合，其中认证块114包括多个有限字段乘法器(multiplier)104、前向块106和多个布尔函数块108。

[0032] 前向块106使用认证密钥操作，而前向块96使用加密密钥来操作。连续地生成密文块，每个密文块都是通过对明文块和相应的前向块96的输出执行布尔函数(XOR)操作而生成的。为了易于理解，只示出了两个计数器92、前向块96、布尔函数98的链。本领域技术人员在实施中将理解，通常地将提供多个计数器92、前向块96、布尔函数98的链。

[0033] 第一个有限字段乘法器104以认证数据作为输入。每个后续的有限字段乘法器104(除了最后一个)将相应的布尔函数块的输出作为输入，其中所述布尔函数块的输出是对先前的有限字段乘法器104的输出和相应的密文块执行布尔函数(XOR)操作的结果。最后一个有限字段乘法器104将相应的布尔函数块的输出作为输入，其中所述布尔函数块的输出是对先前的有限字段乘法器104的输出和认证标签的长度与密文的长度连接后的长度执行布尔函数(XOR)操作后的结果。对倒数第二个有限字段乘法器104的输出和认证标签的长度(len(A))与密文的长度(len(C))连接后的长度进行布尔运算，来生成附于包的密文的认证标签。在各个实施例中，根据AES(认证密钥，0^{128})来推导出每个有限字段乘法器104的被乘数H。

[0034] 因而，对于接收方设备，不论是客户端还是服务器，互补的组合解密器(未示出)可以使用认证密钥先计算密文的认证标签，并确定计算出的认证标签是否与附于密文的认证标签相匹配。如果计算出的认证标签与附于密文的认证标签不匹配，则该帧或包可以被丢弃。并且仅当计算出的认证标签与随附的认证标签相匹配时，才使用加密密钥对密文进行解密。

[0035] 对于中间网络设备，它可以对包进行解密来检查业务流。但是，如上所述，如果没有认证密钥，即使中间网络设备泄密，中间网络设备也不太可能伪造或仿造消息来欺骗客户端/服务器。

[0036] 再次参照图6，在各个实施例中，硬件解决方案99可以是网络接口卡的一部分，或者是处理器/芯片组内集成MAC的一部分。而且，可以从服务器16移除端到端的安全负担，使得在一些实施例中能够增加网络14的规模。这在一些情形下允许无缝地部署该解决方案，而不影响更高层的协议/应用。

[0037] 实施例可以被包括作为系统的一部分，例如具有磁盘存储的系统，如膝上型计算机、台式计算机、服务器、游戏机、机顶盒、媒体记录器等等。

[0038] 实施例可以由硬件、软件、固件、微代码或者上述的任何组合来实现。当以软件、固件或微代码来实现时，实施例的元素是用于执行必要任务的程序代码或者代码段。代码可以是执行操作的实际代码，或者是对操作进行仿真或模拟的代码。代码段可以表示过程、函数、子程序、程序、例程、子例程、模块、软件包、类，或者指令、数据结构或程序语句的任意组合。一个代码段可以通过传递和/或接收信息、数据、变量、参数或存储器内容来耦合到另一代码段或硬件电路。可以通过任何适当的方式传输、转发或者传输信息、变量、参数、数据等，所述方式包括内存共享、消息传递、令牌传递、网络传输等。程序或者代码段可以存储在处理器可读的介质中，或者通过传输介质由包含在在载波中的计算机数据信号或者由载波调制的信号来传输。“处理器可读或可访问的介质”或者“机器可读或可访问的介质”可以包括任何能够存储、传输或传送信息的介质。处理器/机器可读/可访问介质的例子包括电子电路、半导体存储器器件、只读存储器(ROM)、闪速存储器、可擦除只读存储器(EROM)、软盘、紧致盘(CD-ROM)、光盘、硬盘、光纤介质、射频(RF)链路等等。计算机数据信号可以包括能够通过传输介质传播的任何信号，所述传输介质例如是电子网络信道、光纤、空气电磁波、RF链路等等。代码段可以通过例如互连网、内联网之类的计算机网络下载。机器可访问介质可以体现为制品。机器可访问介质可以包括当被机器访问时使机器执行随后所述操作的数据。在这里术语“数据”指的是就机器可读的目的而编码的任何类型的信息。因此，其可以包括程序、代码、数据、文件等等。

[0039] 本文中提到的“一个实施例”或“实施例”表示结合该实施例描述的特定的特征、结构、特性被包括在本发明所涵盖的至少一个实施中。因此，多次出现的短语“一个实施例”或者“在实施例中”并不一定指的是同一个实施例。另外，可以以与所述的具体实施例不同的其他的适当形式实现特定的特征、结构或特点，并且所有这些变型都涵盖在本申请的权利要求范围中。

[0040] 尽管本文说明并描述了特定的实施例，但本领域技术人员将理解对所示出并描述的特定实施例可以替换为多种相当或等同的实现，而不超出本发明实施例的范围。本申请意在覆盖本文所论述实施例的任何修改或变型。因此，显而易见的是，仅由权利要求书及其等同体来限制本发明的实施例。

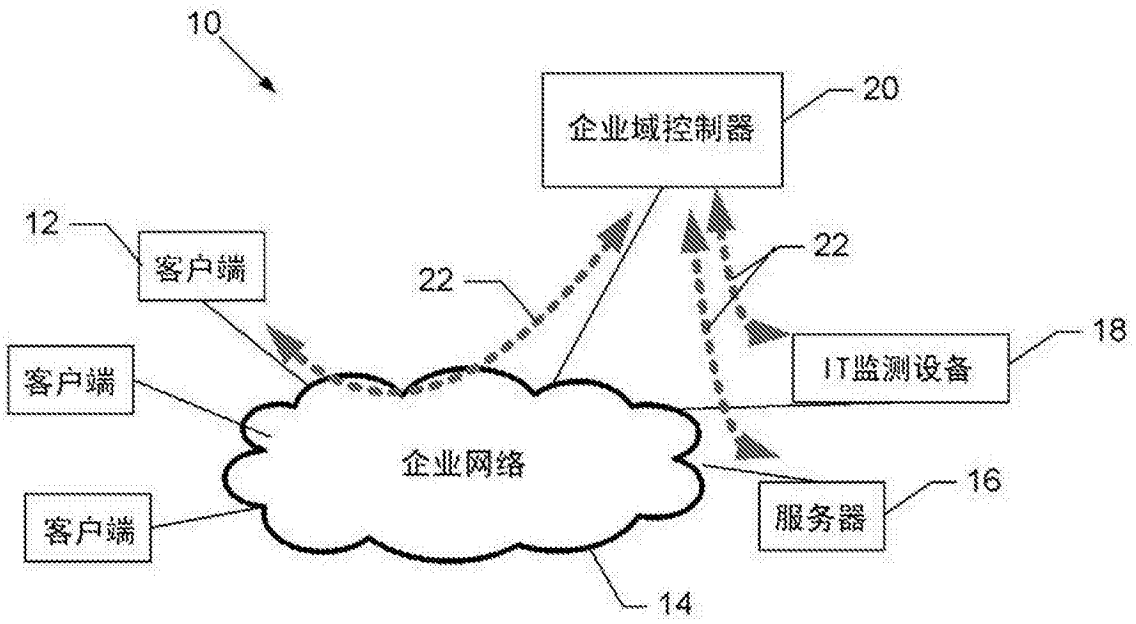

10
企业域控制器
20
12
客户端
22
22
18
IT监测设备
客户端
企业网络
16
服务器
客户端
14

图1

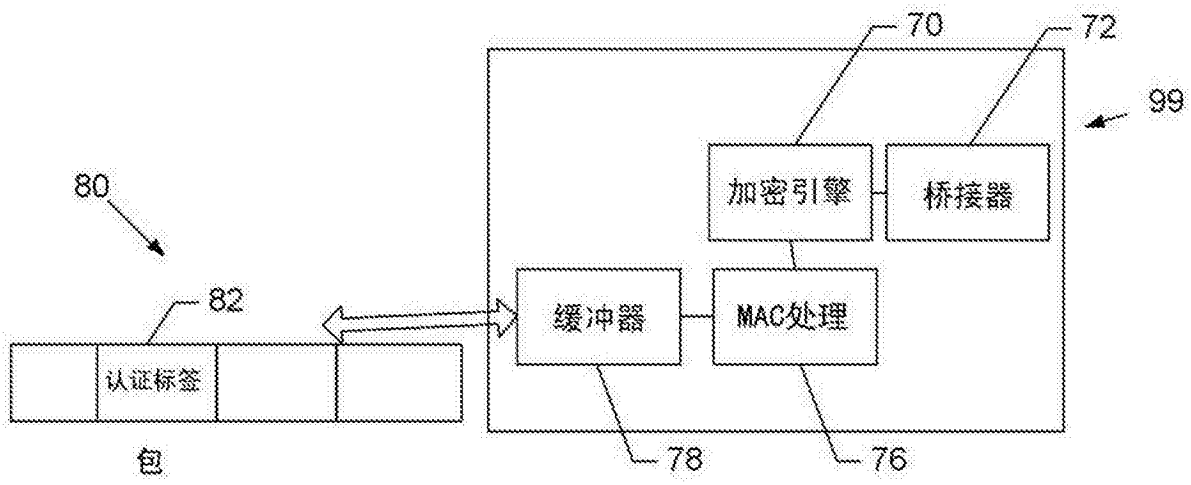

70
72
99
加密引擎
桥接器
80
82
缓冲器
MAC处理
认证标签
包
78
76

图6

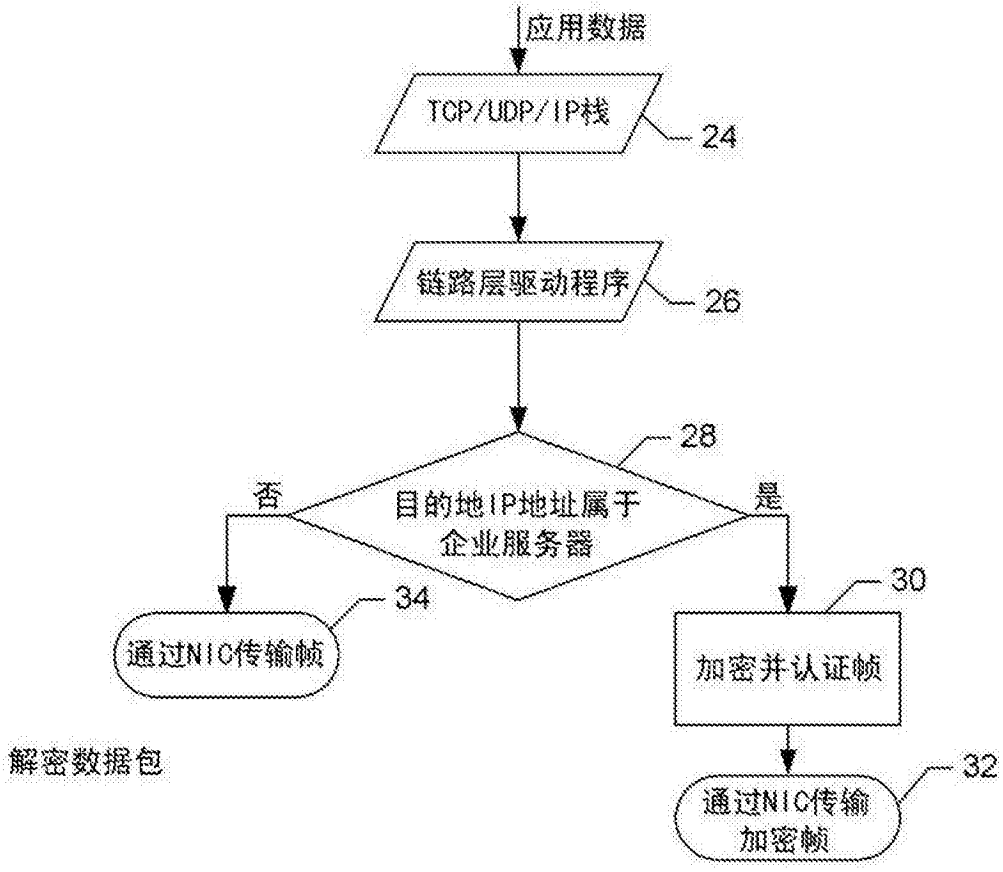

应用数据
TCP/UDP/IP栈
24
链路层驱动程序
26
28
目的地IP地址属于企业服务器
否
是
34
通过NIC传输帧
30
加密并认证帧
解密数据包
32
通过NIC传输加密帧

图2

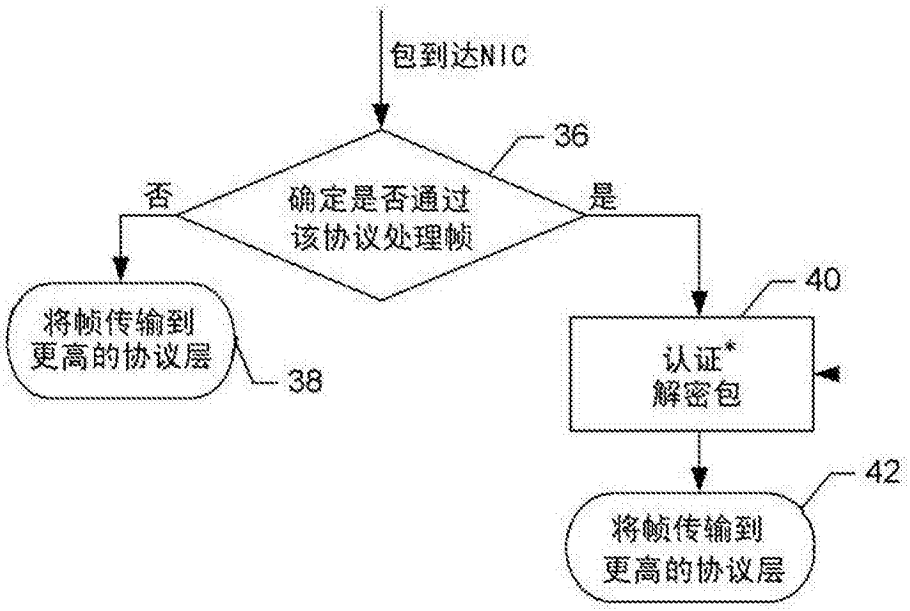

包到达NIC
36
确定是否通过该协议处理帧
否
是
将帧传输到更高的协议层
38
40
认证*解密包
42
将帧传输到更高的协议层

图3

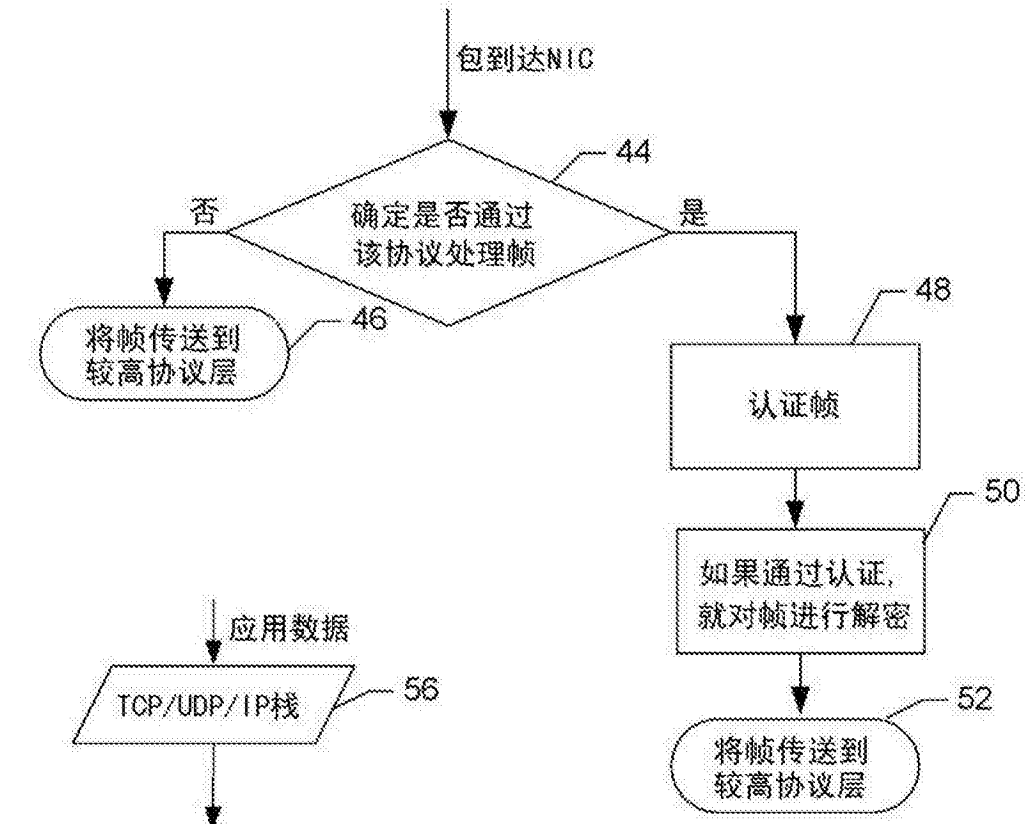

包到达NIC
44
确定是否通过
该协议处理帧
否
是
将帧传送到
较高协议层
46
48
认证帧
50
如果通过认证，
就对帧进行解密
52
将帧传送到
较高协议层

图4

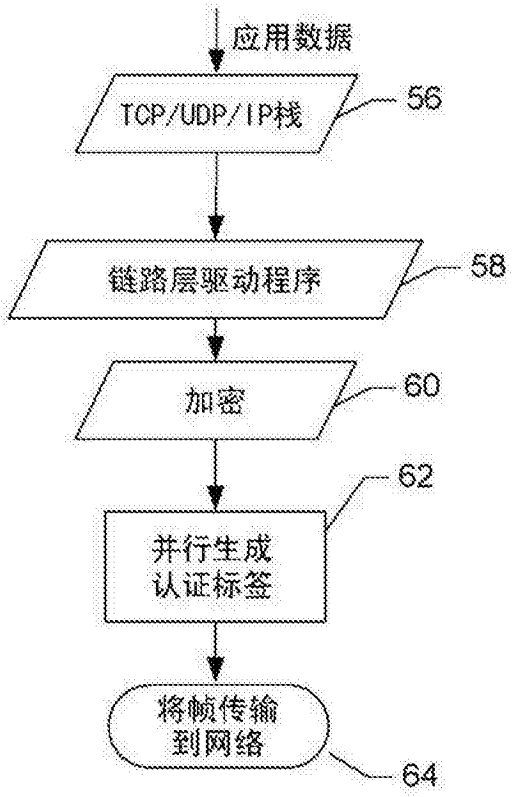

应用数据
TCP/UDP/IP栈
56
链路层驱动程序
58
加密
60
62
并行生成
认证标签
将帧传输
到网络
64

图5

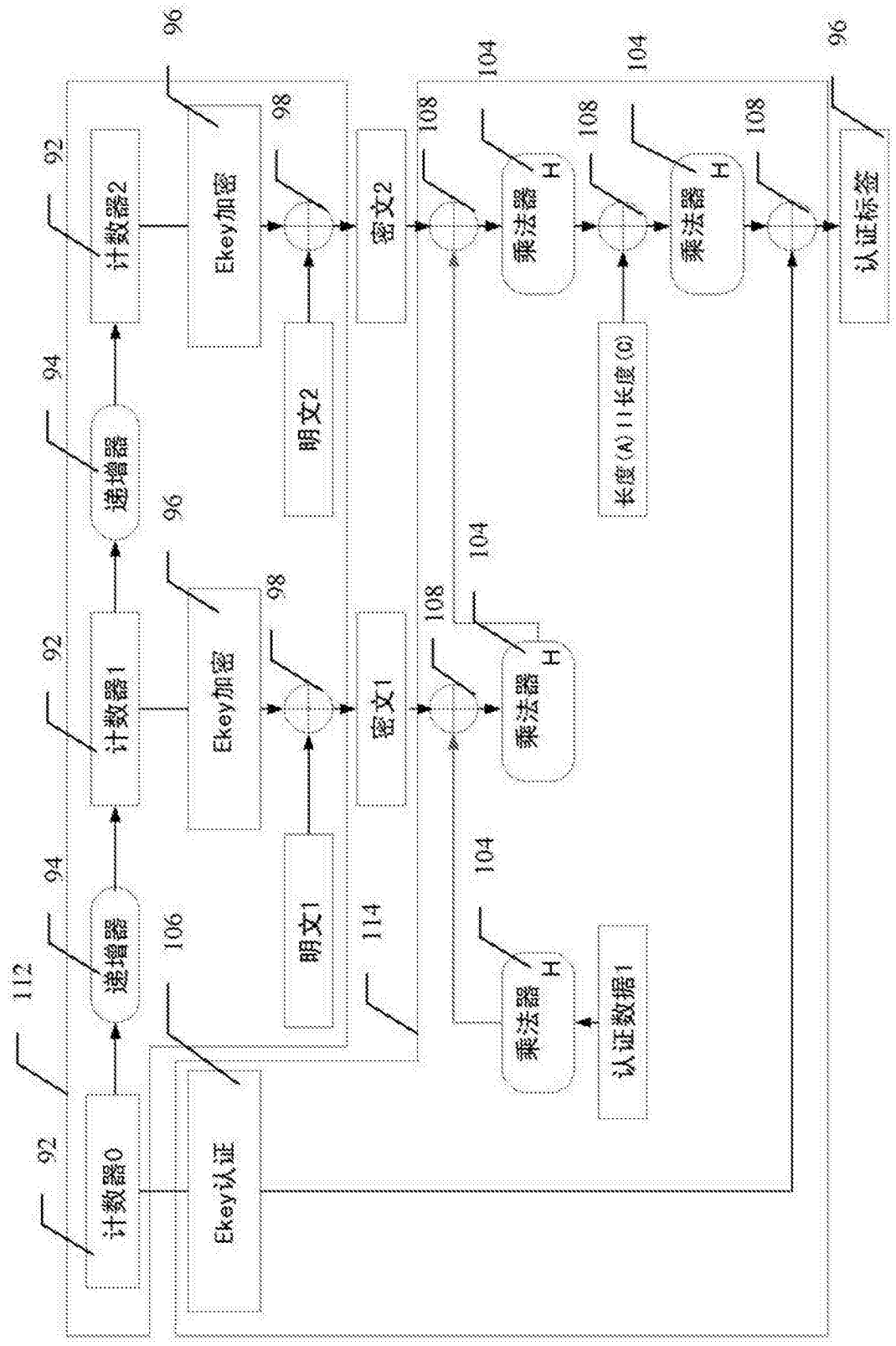

计数器0
计数器1
计数器2
递增器
递增器
递增器
Ekey认证
Ekey加密
Ekey加密
明文1
明文2
密文1
密文2
乘法器 H
乘法器 H
乘法器 H
乘法器 H
长度(A)||长度(C)
认证数据1
认证标签
92
94
96
98
104
106
108
112
114

图7