(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 103930882 A
(43)申请公布日 2014.07.16

(21)申请号 201280055039.2
(22)申请日 2012.11.15
(30)优先权数据
61/560,279 2011.11.15 US
(85)PCT国际申请进入国家阶段日
2014.05.09
(86)PCT国际申请的申请数据
PCT/US2012/065339 2012.11.15
(87)PCT国际申请的公布数据
WO2013/074827 EN 2013.05.23
(71)申请人 NICIRA股份有限公司
地址 美国加利福尼亚
(72)发明人 T·考珀内恩 张荣华 P·萨卡尔 M·卡萨多
(74)专利代理机构 中国国际贸易促进委员会专利商标事务所 11038
代理人 罗银燕
(51)Int.Cl.
G06F 15/177(2006.01)

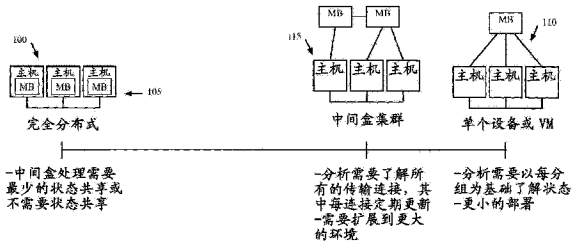
权利要求书2页 说明书25页 附图13页
按照条约第19条修改的权利要求书2页

(54)发明名称

具有中间盒的网络架构

(57)摘要

本发明提供一种用于实现包括由一组逻辑转发元件连接的一组终端机、第一逻辑中间盒以及第二逻辑中间盒的逻辑网络的系统。所述系统包括一组节点。若干节点中的每一个包括：(i)用于所述实现逻辑网络的终端机的虚拟机，(ii)用于实现所述逻辑网络的一组逻辑转发元件的受管理交换元件，以及(iii)用于实现所述逻辑网络的第一逻辑中间盒的中间盒元件。所述系统包括用于实现所述第二逻辑中间盒的物理中间盒设备。



CN 103930882 A

1. 一种用于在包括多个节点的托管系统中配置逻辑网络的方法，所述逻辑网络包括第一中间盒和第二中间盒，所述方法包括：

接收对于所述第一中间盒的第一配置和对于所述第二中间盒的第二配置；

识别用于实现所述第一中间盒的多个节点；

分发用于在所识别的节点上实现的所述第一配置；以及

分发用于在单个物理机器上实现的所述第二配置。

2. 根据权利要求 1 所述的方法，其中，所述单个物理机器包括托管用于实现所述第二中间盒的虚拟机的单个节点。

3. 根据权利要求 1 所述的方法，其中，所述单个物理机器包括物理中间盒设备。

4. 根据权利要求 1 所述的方法，其中，所述方法由第一网络控制器执行。

5. 根据权利要求 4 所述的方法，其中，分发所述第一配置包括：

自动地识别管理所识别的节点的多个附加网络控制器；以及

将所述第一配置分发给所识别的网络控制器以用于随后分发给所识别的节点。

6. 根据权利要求 5 所述的方法，其中，所识别的节点中的每一个由所述物理控制器中的单个物理控制器管理。

7. 根据权利要求 4 所述的方法，其中，分发所述第二配置包括：

自动地从一组附加网络控制器识别管理所述单个物理机器的特定的附加网络控制器；以及

将所述第二配置分发给所识别的特定的网络控制器以用于随后分发给所述单个物理机器。

8. 根据权利要求 1 所述的方法，其中，接收对于所述第一中间盒的第一配置和对于所述第二中间盒的第二配置包括：通过特定于所述第一中间盒的第一应用程序接口 API 接收配置，以及通过特定于所述第二中间盒的第二 API 接收第二配置。

9. 根据权利要求 1 所述的方法，其中，所述第一配置被作为一组数据库表记录接收。

10. 根据权利要求 9 所述的方法，其中，所述方法在分发记录之前不对所述记录进行转换。

11. 一种用于实现包括由一组逻辑转发元件连接的一组终端机、第一逻辑中间盒以及第二逻辑中间盒的逻辑网络的系统，所述系统包括：

一组节点，其中，若干节点中的每一个包括：

虚拟机，所述虚拟机用于实现所述逻辑网络的终端机；

受管理交换元件，所述受管理交换元件用于实现所述逻辑网络的一组逻辑转发元件；和

中间盒元件，所述中间盒元件用于实现所述逻辑网络的第一逻辑中间盒；以及

物理中间盒设备，所述物理中间盒设备用于实现所述第二逻辑中间盒。

12. 根据权利要求 11 所述的系统，其中，所述系统进一步用于实现包括由第二组逻辑转发元件连接的第二组终端机和第三逻辑中间盒的第二逻辑网络。

13. 根据权利要求 12 所述的系统，其中，所述一组节点中的特定节点包括用于实现所述第二逻辑网络的第二终端机的第二虚拟机。

14. 根据权利要求 13 所述的系统，其中，所述受管理交换元件进一步用于实现所述第

二逻辑网络的第二组逻辑转发元件，并且所述中间盒元件进一步用于实现所述逻辑网络的第三逻辑中间盒。

15. 根据权利要求 11 所述的系统，其中，所述第一逻辑中间盒和第二逻辑中间盒是不同类型的中间盒。

16. 根据权利要求 11 所述的系统，其中，所述第一逻辑中间盒和第二逻辑中间盒是在所述逻辑网络中执行不同功能的相同类型的中间盒。

17. 一种用于实现逻辑网络的系统，所述系统包括 ：

多个主机，所述多个主机用于实现所述逻辑网络的第一逻辑中间盒，其中，所述第一中间盒在不在中间盒元件之间传送状态信息的情况下独立地对所述多个主机中的每一个的中间盒元件进行操作 ；以及

一组独立的物理中间盒，所述一组独立的物理中间盒用于实现所述逻辑网络的第二逻辑中间盒，其中，所述第二逻辑中间盒执行需要与所述逻辑网络的若干不同组的终端机之间的分组相关的状态信息的操作。

18. 根据权利要求 17 所述的系统，其中，每个特定的中间盒元件存储关于其分组被所述特定的中间盒元件处理的传输连接的状态信息。

19. 根据权利要求 18 所述的系统，其中，对于特定传输连接的处理不需要关于其它传输连接的任何状态信息。

20. 根据权利要求 17 所述的系统，其中，所述一组独立的物理中间盒包括单个物理中间盒。

21. 根据权利要求 17 所述的系统，其中，所述一组独立的物理中间盒包括作为资源池操作的中间盒集群。

22. 根据权利要求 21 所述的系统，还包括用于在集群中的物理中间盒之间共享状态信息的所述中间盒集群之间的专用网络连接。

23. 根据权利要求 17 所述的系统，其中，所述第一逻辑中间盒包括防火墙、负载均衡器和源网络地址转换器中的一个。

24. 根据权利要求 17 所述的系统，其中，所述第二逻辑中间盒包括执行需要针对每个被处理的分组的状态信息的操作的入侵检测系统。

25. 根据权利要求 17 所述的系统，其中，所述第二逻辑中间盒包括广域网优化器。

具有中间盒的网络架构

背景技术

[0001] 许多当前的企业具有大型且复杂的网络，其包括交换机、集线器、路由器、中间盒（例如，防火墙、负载均衡器、源网络地址转换等）、服务器、工作站、以及支持各种连接、应用和系统的其它联网装置。计算机网络增加的复杂性，包括虚拟机迁移、动态工作负载、多租户、以及客户特定的服务质量和安全配置需要更好的范式以用于网络控制。网络传统地已通过单个网络组件的低级配置来管理。网络配置通常取决于底层网络：例如，利用访问控制列表（“ACL”）条目阻止用户的访问需要知道用户的当前 IP 地址。更复杂的任务需要更广泛的网络知识：迫使访客用户的端口 80 业务（traffic）穿越 HTTP 代理需要知道当前的网络拓扑和每个访客的位置。在网络交换元件在多用户之间共享的情况下，该处理具有增加的难度。

[0002] 作为响应，存在朝向被称为软件定义网络（SDN）的新的网络控制范式的日渐发展。在 SDN 范式中，在网络中的一个或多个服务器上运行的网络控制器以每用户为基础控制、维护并且实现对共享的网络交换元件的转发行为进行管控的控制逻辑。做出网络管理决策通常需要了解网络状态。为了便于做出管理决策，网络控制器创建并且维护网络状态的视图，并且提供管理应用可以访问网络状态的视图的应用程序接口。

[0003] 维护大型网络（包括数据中心和企业网络两者）的主要目标中的一些是可扩展性、移动性和多租户。处理这些目标中的一个所采用的许多方法导致阻碍其它目标中的至少一个。例如，可以容易地在 L2 域内为虚拟机提供网络移动性，但是 L2 域不能扩展到大的尺寸。此外，保持用户隔离极大地使移动性复杂化。这样，需要可以满足可扩展性、移动性和多租户目标的改进的解决方案。

发明内容

[0004] 一些实施例提供一种系统，该系统允许用户指定包括一个或多个中间盒（例如，防火墙、负载均衡器、网络地址转换器、入侵检测系统（IDS）、广域网（WAN）优化器等）的逻辑网络。该系统通过在许多受管理交换元件上分布逻辑转发元件（例如，逻辑交换机、逻辑路由器等）来实现逻辑网络，所述许多受管理交换元件在也托管逻辑网络的虚拟机的许多物理机器上操作。在实现这样的逻辑网络时，一些实施例的系统以不同的方式实现不同的中间盒。例如，系统可以以分布式的方式实现第一中间盒（例如，该中间盒在与受管理交换元件一起也在物理机器上操作的许多受管理中间盒元件上实现），并且以集中式的方式实现第二中间盒（例如，作为单个设备或虚拟机，作为集群）。在一些实施例中，关于是以分布式的方式还是以集中式的方式实现特定的中间盒的确定基于当中间盒为分布式时不同的中间盒元件之间的状态共享要求。

[0005] 在一些实施例中，逻辑中间盒在物理网络中的可能实现的范围从完全分布式的中间盒到完全集中式的中间盒变动，其中在沿着这样的范围的不同点处实现不同的中间盒。另外，可以以集中式或分布式两种方式（包括在同一受管理逻辑网络内）实现单个类型的中间盒。例如，用户可能想要用于对从外部网络传入的所有业务进行过滤的第一防火墙以

及用于对逻辑网络的不同子网之间的业务进行过滤的第二防火墙。在一些情况下，最好的解决方案可以是将第一防火墙实现为向其转发所有外部传入的业务的单个设备，同时以分布式的方式在托管逻辑网络的虚拟机的所有物理机器上实现第二防火墙。

[0006] 所述范围的一端是完全分布式的中间盒架构。在这种情况下，在许多节点（物理主机）上实现中间盒。在一些实施例中，物理主机中的每一个托管含有逻辑中间盒的逻辑网络中的至少一个虚拟机。另外，受管理交换元件在主机的每一个上运行，以便实现逻辑网络的逻辑转发元件。因为特定的物理主机可以托管多于一个的逻辑网络（例如，属于不同的租户）中的虚拟机，所以在主机上运行的分布式中间盒和受管理交换元件都可以被虚拟化，以便实现来自不同逻辑网络的中间盒和逻辑转发元件。

[0007] 在一些实施例中，当中间盒实例之间需要最少的状态共享（或者根本没有状态共享）时，可以以这样的分布式方式实现中间盒。至少一些类型的中间盒是有状态的，因为它们建立用于机器之间（例如，网络中的两个虚拟机之间、网络中的虚拟机与外部机器之间等）的连接的状态。在一些实施例中，中间盒建立用于每个传输层连接（例如，TCP 连接、UDP 连接）的状态。在一些实施例的分布式情况下，在特定主机处操作的中间盒元件创建用于通过它的传输连接的状态，但是不需要与在其它主机上操作的其它中间盒元件共享这些状态。当状态仅应用于托管在特定主机上的虚拟机，并且中间盒不需要使用对于其它虚拟机建立的状态信息来执行任何分析时，那么中间盒可以是分布式的。这样的中间盒的示例包括源网络地址转换 (S-NAT)、目的地网络地址转换 (D-NAT)、以及防火墙。

[0008] 另外，一些实施例允许分布具有最少水平的状态共享的中间盒。例如，负载均衡器可以查询它们对业务进行均衡的机器以确定发送到机器中的每一个的业务的当前水平，然后将此分发给其它的负载均衡器。然而，每个负载均衡元件可以独自运行负载均衡算法，并且以定期的时间间隔执行查询，而不是每次分组被路由到虚拟机中的一个或者每次与虚拟机中的一个建立传输（例如，TCP、UDP 等）连接就与每一个其它的负载均衡元件共享状态信息。

[0009] 范围的另一侧是完全集中式的中间盒实现。在这样的集中式实现中，主机中的受管理交换元件将中间盒要处理的所有业务发送到同一中间盒设备。该单个中间盒可以是在其自己的主机上（或者在与网络中的虚拟机中的一个相同的主机中）在物理网络内操作的单独的物理机器或单独的虚拟机。当受管理交换元件识别分组应被发送到中间盒时，该交换元件通过物理网络将该分组发送到中间盒（例如，经由隧道）。中间盒对该分组进行处理，然后将分组（实际上新的分组）发送到另一个受管理交换元件以进行处理（例如，池节点）。

[0010] 当分布式中间盒将需要数据层速度的分组共享时，一些实施例使用这样的集中式中间盒。也就是说，对于中间盒元件处理的每个业务分组，该元件将必须利用源于分组处理的状态变化来更新所有其它的中间盒实例。因此，通过中间盒的每个业务分组将导致附加业务的暴增，以便更新所有其它的中间盒实例。这样的中间盒的示例包括 IDS 和 WAN 优化器。例如，为了适当地对入侵进行监视，IDS 处理需要知道网络内的所有连接。这样，如果 IDS 是分布式的，则将必须对于分布式 IDS 元件处理的每一个分组发送新的状态更新。

[0011] 作为第三选择，一些实施例对于一些中间盒使用集群架构，该集群架构类似于完全集中式的架构，除了该集群充当集中式资源池，而不是单个的物理机器之外。在一些实施

例中，当使用中间盒的一个网络（或多个网络）较大并且单个设备可能不具有足够的资源（例如，存储器、处理能力等）以处理较大的部署时，中间盒集群（例如，IDS 盒的集群）可能是有益的。然而，当集群是需要了解所有状态信息的中间盒时，那么该状态信息将在集群中的各个机器之间共享。在一些实施例中，当分析不需要以每分组为基础的状态更新、而是以每传输连接（或者每连接若干次更新，但是通常少于每分组）为基础的状态更新时，中间盒集群可能是比单个设备更好的选择。为了执行所需的高速状态共享，一些实施例经由用于共享状态信息的单独的专用高速连接来链接集群中的中间盒机器。

[0012] 前述的发明内容旨在用作对于本发明的一些实施例的简要介绍。它意不在于作为本文档中所公开的所有创造性主题的介绍或概述。后面的具体实施方式以及在具体实施方式中参照的附图将进一步描述在该发明内容中所描述的实施例以及其它的实施例。因此，为了理解本文档所描述的所有实施例，需要发明内容、具体实施方式和附图的全面审阅。而且，因为所要求保护的主题可以在不脱离主题的精神的情况下以其它特定的形式体现，所以要求保护的主题不受发明内容、具体实施方式以及附图中的说明性的细节限制，而是由所附的权利要求限定。

附图说明

[0013] 本发明的新颖特征在所附权利要求中阐述。然而，为了解释的目的，本发明的若干实施例在以下附图中阐述。

[0014] 图 1 概念性地示出逻辑中间盒在物理网络中的实现的范围，其从完全分布式中间盒到完全集中式中间盒变动。

[0015] 图 2 概念性地示出一些实施例的逻辑网络拓扑。

[0016] 图 3 概念性地示出一些实施例的分布式中间盒实现。

[0017] 图 4 概念性地示出一些实施例的中间盒的完全集中式实现。

[0018] 图 5 概念性地示出一些实施例的中间盒作为资源集群的实现。

[0019] 图 6 概念性地示出一些实施例的实现包括入侵检测系统的逻辑网络的网络。

[0020] 图 7 概念性地示出一些实施例的包括分布式软件中间盒元件和软件交换元件两者的主机的架构图。

[0021] 图 8 示出一些实施例的用于配置受管理交换元件和分布式中间盒元件以便实现逻辑网络的网络控制系统。

[0022] 图 9 概念性地示出一些实施例的通过网络控制系统的数据传播。

[0023] 图 10 示出一些实施例的网络控制器的示例架构。

[0024] 图 11 概念性地示出涉及许多中间盒的复杂的逻辑网络拓扑。

[0025] 图 12 概念性地示出图 11 的网络在托管的虚拟化的环境中的一种特定的物理实现。

[0026] 图 13 概念性地示出实现本发明的一些实施例的电子系统。

具体实施方式

[0027] 在以下本发明的详细描述中，对本发明的许多细节、示例和实施例进行阐述和描述。然而，对于本领域技术人员将清楚和明显的是，本发明不限于所阐述的实施例，并且可

以在没有所讨论的特定细节和示例的情况下实施本发明。

[0028] 一些实施例提供一种系统，该系统允许用户指定包括一个或多个中间盒（例如，防火墙、负载均衡器、网络地址转换器、入侵检测系统（IDS）、广域网（WAN）优化器等）的逻辑网络。该系统通过在许多受管理交换元件上分布逻辑转发元件（例如，逻辑交换机、逻辑路由器等）来实现逻辑网络，所述许多受管理交换元件在也托管（host）逻辑网络的虚拟机的许多物理机器上操作。

[0029] 在实现这样的逻辑网络时，一些实施例的系统以不同的方式实现不同的中间盒。例如，系统可以以分布式的方式实现第一中间盒（例如，该中间盒在与受管理交换元件一起也在物理机器上操作的许多受管理中间盒元件上实现），并且以集中式的方式实现第二中间盒（例如，作为单个设备或虚拟机，作为集群）。在一些实施例中，关于是以分布式的方式还是以集中式的方式实现特定的中间盒的确定基于当中间盒为分布式时不同的中间盒元件之间的状态共享要求。

[0030] 图 1 概念性地示出了逻辑中间盒在物理网络中的实现的范围（spectrum）100，其从完全分布式的中间盒到完全集中式的中间盒变动。如所提及的，可以在沿着这个范围的不同点处实现不同的中间盒。另外，可以以集中式或分布式两种方式（包括在同一受管理逻辑网络内）实现单个类型的中间盒。例如，用户可能想要用于对从外部网络传入（incoming）的所有业务进行过滤的第一防火墙、以及用于对逻辑网络的不同子网之间的业务进行过滤的第二防火墙。在一些情况下，最好的解决方案可以是将第一防火墙实现为向其转发所有外部传入的业务的单个设备，同时以分布式的方式在托管逻辑网络的虚拟机的所有物理机器上实现第二防火墙。

[0031] 范围 100 的左端是一些实施例的完全分布式的中间盒架构 105。如该图中所示，在许多节点（物理主机）上实现中间盒。在一些实施例中，物理主机中的每一个托管含有逻辑中间盒的逻辑网络中的至少一个虚拟机。另外，受管理交换元件在主机的每一个上运行，以便实现逻辑网络的逻辑转发元件（例如，逻辑路由器、逻辑交换机）。因为特定的物理主机可以托管多于一个的逻辑网络（例如，属于不同的租户）中的虚拟机，所以在主机上运行的分布式中间盒和受管理交换元件都可以被虚拟化，以便实现来自不同逻辑网络的中间盒和逻辑转发元件。在一些实施例中，中间盒被实现为在主机的超管理器（hypervisor）内运行的一个模块或一组模块。

[0032] 如该图中所示的，当中间盒实例之间需要最少的状态共享（或者根本没有状态共享）时，可以以这样的分布式方式实现中间盒。至少一些类型的中间盒是有状态的，因为它们建立用于机器之间（例如，网络中的两个虚拟机之间、网络中的虚拟机与外部机器之间等）的连接的状态。在一些实施例中，中间盒建立用于每个传输层连接（例如，TCP 连接、UDP 连接）的状态。在一些实施例的分布式情况下，在特定主机处操作的中间盒元件创建用于通过它的传输连接的状态，但是不需要与在其它主机上操作的其它中间盒元件共享这些状态。当状态仅应用于托管在特定主机上的虚拟机，并且中间盒不需要使用对于其它虚拟机建立的状态信息来执行任何分析时，那么中间盒可以是分布式的。这样的中间盒的示例包括源网络地址转换（S-NAT）、目的地网络地址转换（D-NAT）、以及防火墙。

[0033] 另外，一些实施例允许分布具有最少水平（level）的状态共享的中间盒。例如，负载均衡器可以查询它们对业务进行均衡的机器以确定发送到机器中的每一个的业务的当

前水平，然后将此分发给其它的负载均衡器。然而，每个负载均衡元件可以独自运行负载均衡算法，并且以定期的时间间隔执行查询，而不是每次分组被路由到虚拟机中的一个或者每次与虚拟机中的一个建立传输（例如，TCP、UDP 等）连接就与每一个其它的负载均衡元件共享状态信息。

[0034] 范围 100 的另一侧是完全集中式的中间盒实现 110。在该实现中，主机中的受管理交换元件将中间盒要处理的所有业务发送到同一中间盒设备。该单个中间盒可以是在其自己的主机上（或者在与网络中的虚拟机中的一个相同的主机中）在物理网络内操作的单独的物理机器或单独的虚拟机。当受管理交换元件识别分组应被发送到中间盒时，该交换元件通过物理网络将该分组发送到中间盒（例如，经由隧道）。中间盒对该分组进行处理，然后将分组（实际上新的分组）发送到另一个受管理交换元件以进行处理。

[0035] 在一些实施例中，新的分组被发送到的受管理交换元件是池节点。在一些实施例中，池节点是位于网络内部（即，不直接连接到网络边缘处的虚拟机中的任何一个）的特定类型的受管理交换元件，其用于处理边缘交换元件（即，位于主机上、直接连接到虚拟机的那些交换元件）不能处理的业务。在其它实施例中，不是将分组发送到池节点，中间盒而是将业务直接发送到直接构建到中间盒设备中的受管理交换元件。

[0036] 当分布式中间盒将需要数据层速度的分组共享时，一些实施例使用这样的集中式中间盒。也就是说，对于中间盒元件处理的每个业务分组，该元件将必须利用源于分组处理的状态变化来更新所有其它的中间盒实例。因此，通过中间盒的每个业务分组将导致附加业务的暴增，以便更新所有其它的中间盒实例。这样的中间盒的示例包括 IDS 和 WAN 优化器。例如，为了适当地对入侵进行监视，IDS 处理需要知道网络内的所有连接。这样，如果 IDS 是分布式的，则将必须对于分布式 IDS 元件处理的每一个分组发送新的状态更新。

[0037] 中间盒集群架构 115 类似于完全集中式的架构 110，除了该集群充当集中式资源池，而不是单个的物理机器之外。如该图中所示的，当使用中间盒的一个网络（或多个网络）较大并且单个设备可能不具有足够的资源（例如，存储器、处理能力等）以处理较大的部署时，中间盒集群（例如，IDS 盒的集群）可能是有益的。然而，当集群是需要了解所有状态信息的中间盒时，那么该状态信息将在集群中的各个机器之间共享。在一些实施例中，当分析不需要以每分组为基础的状态更新、而是以每传输连接（或者每连接若干次更新，但是通常少于每分组）为基础的状态更新时，中间盒集群可能是比单个设备更好的选择。为了执行所需的高速状态共享，一些实施例经由用于共享状态信息的单独的专用高速连接来链接集群中的中间盒机器。

[0038] 以上示出了一些实施例的网络中的逻辑中间盒的不同实现的示例。下面描述若干更详细的实施例。节 I 描述一些实施例的不同的中间盒架构。节 II 描述一些实施例的分布式中间盒实现。接着，节 III 描述一些实施例的用于配置网络以便实现包括一个或多个中间盒的逻辑网络的网络控制系统。节 IV 然后描述具有许多不同的中间盒的网络的说明性示例。最后，节 V 描述实现本发明的一些实施例的电子系统。

[0039] I. 不同的中间盒架构

[0040] 如上所述，一些实施例在受管理网络内使用不同的架构来实现不同的中间盒。即使对于相同的逻辑网络拓扑，一些中间盒以分布式方式实现（中间盒元件在网络的虚拟机在其上操作的每个主机上操作），而其它中间盒则以集中式方式实现（主机上的受管理交

换元件连接到单个设备或集群）。

[0041] 图 2 概念性地示出了一些实施例的逻辑网络拓扑 200。为了解释的目的，网络拓扑 200 为简化的网络。该网络包括由逻辑 L3 路由器 215 连接的两个逻辑 L2 交换机 205 和 210。逻辑交换机 205 连接虚拟机 220 和 225，而逻辑交换机 210 连接虚拟机 230 和 235。逻辑路由器 215 还连接到外部网络 250。

[0042] 另外，中间盒 240 附连到逻辑路由器 215。本领域普通技术人员将认识到，网络拓扑 200 表示可以将中间盒并入到其中的仅一个特定的逻辑网络拓扑。在各个实施例中，中间盒可以直接位于两个其它的组件（例如，）之间、直接位于网关与逻辑路由器之间（例如，以便监视和处理进入或退出逻辑网络的所有业务）、或者位于更复杂网络中的其它位置中。

[0043] 在图 2 中所示的架构中，中间盒 240 不位于从一个域到另一个域或者外部世界与域之间的直接业务流内。因此，分组将不被发送到中间盒，除非对于逻辑路由器 215 指定（例如，由诸如网络管理员的用户指定）确定哪些分组应被发送到中间盒以进行处理的路由策略。一些实施例使得能够使用基于目的地地址（例如，目的地 IP 或 MAC 地址）以外的数据来转发分组的策略路由规则。例如，用户可以指定（例如，通过网络控制器应用程序接口 (API)）应将下述分组引导到中间盒 240 以进行处理 ：具有由逻辑交换机 205 交换的逻辑子网中的源 IP 地址的所有分组、或者从外部网络 250 进入网络的、去往由逻辑交换机 210 交换的逻辑子网的所有分组。

[0044] 不同的中间盒可以在网络内执行不同的功能。例如，防火墙分析数据分组以确定是否应允许分组通过（即，类似于 ACL 流条目 (flow entry)）。防火墙存储确定防火墙是否丢弃（即，舍弃）或允许分组通过（或者，在一些情况下，通过丢弃分组并且将错误响应发送回发送者来拒绝分组）的一组规则（例如，由用户键入）。在一些实施例中，防火墙是保持跟踪传输（例如，TCP 和 / 或 UDP）连接、并且使用所存储的状态信息来做出更快速的分组处理决策的状态性防火墙。

[0045] 源网络地址转换 (S-NAT) 修改分组头中的分组的源 IP 地址。例如，可以使用 S-NAT，使得可以通过将来自不同机器的分组的源改变为单个 IP 地址来使具有不同 IP 地址的许多不同机器的 IP 地址对于目的地机器是隐藏的。目的地网络地址转换 (D-NAT) 类似地修改分组的目的地 IP 地址，以便使真实 IP 地址对于源机器隐藏。负载均衡是使用各种算法（例如，轮询 (round robin)、随机分配等）来在许多目的地机器之间均衡业务的 D-NAT 的形式。负载均衡器接收显露给源机器的特定 IP 地址的分组，并对该分组的目的地 IP 地址修改以与通过负载均衡算法选择的目的地机器中的特定一个匹配。

[0046] 在一些实施例中，入侵检测系统 (IDS) 是对于恶意活动或策略违反监视逻辑网络的被动 (passive) 中间盒。IDS 可以检查传输连接（例如，TCP 连接、UDP 连接等）以确定网络上的攻击是否发生。

[0047] WAN 优化器是用于提高 WAN 上的数据传送的效率（例如，使 WAN 上的数据的流加速）的中间盒装置。WAN 优化技术的示例包括重复数据删除、数据压缩、延迟优化、缓存和 / 或代理、转发纠错、协议欺骗、业务整形、均衡、连接限制、简单速率限制等。尽管以上是若干不同的中间盒中的一些的列表，但是本领域普通技术人员将认识到，一些实施例可以包括可以以分布式或集中式的方式实现的各种不同的中间盒。

[0048] 取决于中间盒的类型，在一些情况下，还取决于用户请求的实现的类型，诸如图 2

中所示的中间盒将以集中式方式或分布式方式实现。图 3 概念性地示出了一些实施例的这样的分布式实现 300。具体地，图 3 示出了若干节点，包括第一主机 305、第二主机 310、第三主机 315、以及第 N 主机 320。头三个节点中的每一个托管网络 200 的若干虚拟机，其中，虚拟机 220 托管在第一主机 305 上，虚拟机 225 和 235 托管在第二主机 310 上，虚拟机 230 托管在第三主机 315 上。

[0049] 另外，主机中的每一个包括受管理交换元件（“MSE”）。一些实施例的受管理交换元件是实现用于一个或多个逻辑网络的逻辑转发元件的软件转发元件。例如，主机 305-320 中的 MSE 包括实现网络 200 的逻辑转发元件的转发表中的流条目。具体地，主机上的 MSE 实现逻辑交换机 205 和 210、以及逻辑路由器 215。另一方面，一些实施例仅当连接到逻辑交换机的至少一个虚拟机位于特定节点处时才在该节点处实现逻辑交换机（即，在主机 305 处的 MSE 中仅实现逻辑交换机 205 和逻辑路由器 215）。第 N 主机 320 不包括来自网络 200 的任何虚拟机，因此，驻留在该主机上的 MSE 不实现来自网络 200 的任何逻辑转发元件。

[0050] 一些实施例的实现 300 还包括连接到主机的池节点 340。在一些实施例中，驻留在主机上的 MSE 执行第一跳处理。也就是说，这些 MSE 是分组在从虚拟机被发送之后到达的第一转发元件，并且试图在该第一跳处执行所有的逻辑交换和路由。然而，在一些情况下，特定的 MSE 可能没有存储含有网络的所有逻辑转发信息的流条目，并因此可能不知道对特定的分组如何处理。在一些这样的实施例中，MSE 将分组发送到池节点 340 以进行进一步处理。这些池节点是内部的受管理交换元件，在一些实施例中，这些受管理交换元件存储包含逻辑网络的比边缘软件交换元件更大的一部分的流条目。

[0051] 类似于逻辑交换元件在网络 200 的虚拟机驻留在其上的主机上的分布，中间盒 240 分布在这些主机 305-315 上的中间盒元件上。在一些实施例中，中间盒模块（或一组模块）驻留在主机上（例如，在主机的超管理器中操作）。当用户设置逻辑网络（例如，网络 200）时，输入包括来自中间盒的配置。例如，对于防火墙，用户将输入用于分组过滤的一组规则（例如，基于 IP 地址、TCP 连接等）。在一些实施例中，被用于对受管理交换元件进行预配置以实现逻辑转发元件的网络控制系统还可以被用于对在主机上操作的各个中间盒元件进行预配置。当用户将中间盒配置输入到网络控制系统的控制器中时，控制器识别应在其上实现中间盒配置的特定节点，并将该配置分发给这些节点（例如，通过一组控制器）。

[0052] 当虚拟机中的一个发送分组（例如，发送到虚拟机中的另一个、外部地址等）时，分组首先进入本地受管理交换元件以进行处理。MSE 可以使用其存储的流条目来做出将分组发送到中间盒的转发决策，在这种情况下，一些实施例将分组发送到同一主机上的本地中间盒元件。在一些实施例中，中间盒元件和 MSE 协商以最小延迟通过其传送分组的软件端口。在中间盒对分组进行处理之后，一些实施例然后通过该相同的端口将分组发送回 MSE。在一些实施例中，该分组作为新的分组被从中间盒发送到 MSE，因此需要通过 MSE 进行新的处理。然而，在一些情况下，没有分组被发送回。例如，如果中间盒是防火墙，则该中间盒可以阻止或丢弃分组。另外，中间盒的一些实施例是被动的，并且分组的副本被发送到中间盒以便中间盒保持跟踪统计数据，但是这些副本不被发送回交换元件。

[0053] 尽管图 3 仅示出了在主机 305-320 上实现的一个逻辑网络，但是一些实施例在一组主机上实现许多逻辑网络（例如，对于不同的租户）。这样，特定主机上的中间盒元件实际上可以存储对于属于若干不同逻辑网络的若干不同中间盒的配置。例如，可以将防火墙

元件虚拟化以实现两个（或更多个）不同的防火墙。这些将有效地作为两个独立的中间盒处理进行操作，使得中间盒元件被切分（slice）成若干（同一类型的）“虚拟”中间盒。另外，当主机上的MSE将分组发送到中间盒时，一些实施例在分组上附加（例如，前置（prepend））切分标识符（或标签）以识别该分组正被发送给若干虚拟中间盒中的哪个。当在用于单个逻辑网络的同一中间盒元件上实现多个中间盒（例如，两个不同的负载均衡器）时，切分标识符将需要识别特定的中间盒切分，而不仅仅识别分组所属的逻辑网络。不同的实施例可以对于中间盒使用不同的切分标识符。

[0054] 在一些实施例中可以是分布式的中间盒的示例包括防火墙、S-NAT 和负载均衡器。在这些情况的每一个下，中间盒在分组处理中起主动作用（即，S-NAT 和负载均衡器分别修改分组的源地址和目的地地址，同时防火墙做出关于是否允许分组还是丢弃分组的决策）。然而，特定节点上的这些中间盒元件中的每一个可以独自运行，而不需要来自其它节点上的相应中间盒元件的信息。甚至分布式负载均衡器元件均可以单独地在不同虚拟机之间对到来的业务进行负载均衡，假定条件是没有虚拟机有可能变得过载，只要其它的负载均衡器元件使用相同的算法即可。然而，在一些实施例中，负载均衡器元件将在某一层次上共享状态（例如，在查询目的地虚拟机以用于使用和健康统计之后）。

[0055] 如所说明的，图 3 概念性地示出了一些实施例的用于逻辑网络 200 的中间盒 240 的分布式实现。另一方面，图 4 概念性地示出了中间盒 240 的完全集中式的实现 400。像图 3 的分布式示例那样，该实现也包括托管虚拟机 220-235 的若干节点 405-420。这些虚拟机再次被布置为第一虚拟机 220 托管在节点 405 上、第二虚拟机 225 和第四虚拟机 235 托管在节点 410 上、以及第三虚拟机 230 托管在节点 415 上。类似地，处于节点 405-415 上的受管理交换元件实现逻辑转发元件 205-215。如在分布式中间盒示例中那样，受管理交换元件对来源于虚拟机的分组执行第一跳处理。

[0056] 然而，在该示例中，中间盒 240 不分布在主机 405-415 上。相反，中间盒被实现为主机外部的单个机器。在不同的实施例中，对于不同类型的中间盒，该单个盒可以是单个物理设备（例如，单独的物理装置）或单个虚拟机（其实际上可以在主机中的一个或不同的主机上操作）。例如，一些实施例可以提供第一类型的中间盒（例如，WAN 优化器）作为虚拟机，而提供第二类型的中间盒（例如，IDS）作为单个设备。另外，一些实施例可以为单个类型的中间盒提供两种选择。

[0057] 与分布式中间盒一样，在一些实施例中，网络控制系统用于对集中式中间盒设备或虚拟机进行预配置。不是控制器接收配置信息并且识别应将该配置信息分发到的许多节点，一些实施例的控制器而是识别在其上实现中间盒的设备，并将配置分发给该设备（例如，通过管理该设备的中间控制器）。在一些实施例中，若干物理设备可以存在于物理网络内，并且控制器选择这些设备中的一个来实现中间盒。当中间盒被实现为虚拟机时，一些实施例为该虚拟机选择托管节点，并然后将配置分发给该节点。在任何一种情况下，网络控制系统还指定主机上的各个受管理交换元件与中间盒之间的连接或附连。在一些实施例中，中间盒设备支持一种或多种类型的隧道，并且分发给受管理交换元件的流条目包括指定为了将分组发送到中间盒而使用的隧道封装的条目。

[0058] 当受管理交换元件中的流条目指定将业务发送到中间盒时，受管理交换元件还使用该隧道信息来封装分组，并通过该隧道将分组从其主机送出到中间盒。与分布式中间盒

一样，一些集中式中间盒是主动中间盒。也就是说，中间盒在执行其中间盒处理之后将分组发送回网络。在一些实施例中，这样的中间盒被配置为总是将分组（作为新的分组）发送到池节点（例如，总是相同的池节点，若干池节点中的一个）。在图 4 中，集中式中间盒 425 将所有其传出 (outgoing) 业务发送到池节点 430。也实现逻辑转发元件 205-215 的池节点然后将分组转发给适当的目的地机器。

[0059] 正如可以将分布式中间盒元件虚拟化以对于若干不同的逻辑网络执行中间盒实例那样，也可以对集中式中间盒 425 进行虚拟化。许多不同的逻辑网络可以使用同一物理设备（或虚拟机）。在一些实施例中，使用与在分布式架构中使用的切分技术类似的切分技术。也就是说，受管理交换元件添加标签以指示分组正被发送到的逻辑网络（或该逻辑网络中的特定的逻辑中间盒），并且中间盒 425 使用该标签来识别它所实现的中间盒实例中的哪个应被用于对分组进行处理。在一些实施例中，集中式中间盒设备包括许多端口，这些端口中的每一个映射到不同的虚拟中间盒实例。在这样的实施例中，可以不使用切分技术，而是使用传入端口来识别正确的虚拟中间盒。

[0060] 鉴于中间盒 425 是单个资源，如图 5 的实现 500 中所示，一些实施例将中间盒实现为集中式的资源集群。该示例与图 4 中所示的示例相同，除了不是单个中间盒装置，网络包括具有三个中间盒资源 510-520 的中间盒集群 505。在一些实施例中，中间盒资源 510-520 中的每一个是单独的装置或虚拟机（即，中间盒 425 的等同物）。

[0061] 不同的实施例可以在中间盒集群内使用不同的架构。一些实施例包括集群的入口点（例如，单个物理装置），其在资源池上对分组进行负载均衡。其它实施例具有直接连接到集群内的不同资源的不同主机。例如，网络 500 可以被设置为第一主机 405 连接到资源 510、第二主机 410 连接到资源 515、以及第三主机 415 连接到资源 520。其它实施例使用其中集群具有两个装置的主 - 备份设置。主机全部连接到主装置 (master)，其执行中间盒处理，同时与备份资源共享状态数据。

[0062] 如以上通过参照图 2 所描述的，当以每分组为基础需要状态共享时，一些实施例使用集中式中间盒实现。也就是说，对于某些中间盒，中间盒处理需要了解中间盒处理的所有分组。对于分布式中间盒，这将需要正在通过连接中间盒元件的网络送出的状态更新的暴增。然而，当中间盒是单个设备时，该单个设备将始终存储所有的状态。

[0063] 对于诸如集群 505 的中间盒集群，该要求意味着必须在集群中的中间盒资源之间高速共享状态。一些实施例使用中间盒资源之间的专用连接来共享该状态信息。也就是说，每个中间盒装置上的特定端口仅专用于集群中的若干装置之间的状态共享。通常，中间盒集群将仅为近距离操作的、使这样的专用连接更加可行的两个机器或若干机器。对于多于两个中间盒的情况，一些实施例使用网状网络，其中，每个中间盒资源通过网络将状态更新广播给所有其它的中间盒资源。其它实施例使用星形网络，其中，中间盒资源将它们的状态更新发送到中央资源，其合并更新并将它们发送到其它资源。尽管与集中式情况相比，中间盒集群需要该附加基础设施，但是集群具有能够应对其中对更多的分组进行处理的更大部署的益处。

[0064] 如所提及的，因为状态共享要求，WAN 优化器和入侵检测系统都是一些实施例将其实现为集中式中间盒的中间盒的示例。WAN 优化器例如通过使用各种优化技术来提高 WAN 上的数据传送的效率。执行这些优化技术需要访问通过 WAN 发送的所有业务，因此，集中式

实现是更优化的。此外，WAN 优化器可以用于缓存通过 WAN 发送的内容，并且如果缓存被一起存储、而不是分布在许多主机上，则缓存恰好符合其目的。

[0065] 入侵检测系统是被动系统（即，不丢弃或修改分组），其监视连接总数、这些连接上的地址、对于每个连接的分组的数量等。为了检测入侵，IDS 查找连接、探试 (heuristic) 等中的模式，对于这些模式，IDS 处理必须知道所有的被监视的业务。如果一个分布式元件具有关于第一连接的信息并且第二分布式元件具有关于第二连接的信息，则没有一个元件具有用于适当地对网络进行入侵评估的足够信息。

[0066] 图 6 概念性地示出了实现一些实施例的包括入侵检测系统 625 的逻辑网络的网络 600。该逻辑网络包括托管在三个节点 605、610 和 620 之间的四个虚拟机。这种情况下的逻辑网络的逻辑拓扑与图 2 中所示的逻辑拓扑相同（其中，IDS 作为中间盒），但是这里描述的 IDS 实现的方面也适用于其它的网络拓扑。网络 600 还包括没有托管特定的逻辑网络的任何虚拟机（但是托管来自另一网络的至少一个虚拟机）的第四节点 615。

[0067] 与前面的图不同，图 6 连同用于示出系统中的机器之间的分组传送的方向的箭头一起示出连接。因此，例如，所有的主机 605-620 可以在两个方向上彼此发送分组。也就是说，主机 605 处的虚拟机将分组发送到 620 处的虚拟机（经由主机 605 处的受管理交换元件、然后主机 620 处的受管理交换元件），并且主机 620 处的虚拟机也将分组发送到主机 605 处的虚拟机。另外，主机处的所有 MSE 可以使用池节点来处理边缘 MSE 不能对其做出转发决策的分组。

[0068] 托管来自特定的逻辑网络的虚拟机的主机 605、610 和 620 中的每一个将分组发送到 IDS。在一些实施例中，逻辑网络上的所有业务被发送到 IDS。然而，这些箭头是单向的，因为一些实施例的入侵检测系统是被动中间盒。一些实施例不是通过中间盒转发业务，而是将重复的分组发送到 IDS 盒 625。IDS 接收这些重复分组（即，对于在主机和 / 或外部网络之间通过网络发送的每一个的分组），并执行其入侵检测分析。因为入侵检测系统 625 不输出任何业务分组，所以在该图中，在 IDS625 与池节点 630 之间不需要连接。

[0069] II. 分布式中间盒实现

[0070] 如上所述，与一些实施例的集中式中间盒实现相比，一些实施例以分布式方式实现一个或多个不同的中间盒，其中，中间盒元件在逻辑网络的虚拟机和受管理交换元件所位于的主机中的一些或全部中操作。本节描述一些实施例的主机内的分布式中间盒实现。

[0071] 图 7 概念性地示出了一些实施例的主机 700 的架构图，该主机 700 包括分布式软件中间盒元件和软件交换元件两者。分布式软件中间盒元件可以是网络地址转换元件、防火墙元件、负载均衡元件、或以分布式方式实现的任何其它中间盒。

[0072] 在这个示例中，中间盒元件包括主机上的三个组件—在主机 700 的用户空间中运行的中间盒守护进程 (daemon)790、以及在主机 700 的内核中运行的中间盒内核模块 795。尽管该图为了解释的目的示出了作为两个组件的分布式中间盒，但是中间盒守护进程 790 和中间盒内核模块 795 共同形成在主机 700 上运行的中间盒元件。软件交换元件（在该示例中，开放虚拟交换机（“OVS”））包括三个组件—在主机 700 的内核中运行的 OVS 内核模块 745、以及 OVS 守护进程 765 和 OVS 数据库 (DB) 守护进程 767，这两者在主机的用户空间中运行。

[0073] 如图 7 中所示，主机 700 包括硬件 705、内核 720、用户空间 721 以及 VM785-795。

硬件 705 可以包括典型的计算机硬件，诸如处理单元、易失性存储器（例如，随机存取存储器 (RAM)）、非易失性存储器（例如，硬盘驱动器、闪存、光盘等）、网络适配器、视频适配器、或任何其它类型的计算机硬件。如所示的，硬件 705 包括 NIC710 和 715，在一些实施例中，NIC710 和 715 是典型的用于将计算装置连接到网络的网络接口控制器。

[0074] 如图 7 中所示，主机 700 包括内核 720 和用户空间 721。在一些实施例中，内核是在单独的存储器空间中运行并且负责管理系统资源（例如，硬件与软件资源之间的通信）的操作系统的最基本的组件。相比之下，用户空间是所有用户模式应用可以在其中运行的存储器空间。

[0075] 一些实施例的内核 720 是在硬件 705 之上运行并且在任何操作系统下面运行的软件抽象层。在一些实施例中，内核 720 执行虚拟化功能（例如，将对于在主机上操作的若干虚拟机的硬件 705 虚拟化）。在一些实施例中，内核 720 则是超管理器的一部分。内核 720 处理各种管理任务，诸如存储器管理、处理器调度、或用于控制在主机上操作的 VM735 和 738 的执行的任何其它操作。

[0076] 如所示的，内核 720 包括分别用于 NIC710 和 715 的装置驱动器 725 和 730。装置驱动器 725 和 730 允许（例如，虚拟机的）操作系统与主机 700 的硬件交互。在该示例中，装置驱动器 725 允许与 NIC710 交互，而驱动器 730 允许与 NIC715 交互。内核 720 可以包括用于允许虚拟机与主机 700 中的其它硬件（未示出）交互的其它装置驱动器（未示出）。

[0077] 虚拟机 735 和 738 是在主机 700 上运行的、使用由内核 720 虚拟化的资源的独立的虚拟机（例如，用户虚拟机，诸如图 3-6 中所示的那些虚拟机）。这样，VM 运行任何数量的不同操作系统。这样的操作系统的示例包括 Solaris、FreeBSD、或任何其它类型的基于 Unix 的操作系统。其它示例还包括基于 Windows 的操作系统。

[0078] 如所示的，主机 700 的用户空间 721 包括中间盒守护进程 790、OVS 守护进程 765 以及 OVS DB 守护进程 767。其它应用（未示出）也可以包括在用户空间 721 中，包括用于附加的分布式中间盒（例如，防火墙、负载均衡器、网络地址转换器等）的守护进程。OVS 守护进程 765 是在用户空间 721 中运行的应用。OVS 守护进程 765 的一些实施例与网络控制器 780 通信，以便接收如下面更详细地描述的用于处理和转发发送到虚拟机 735 和 738 的以及从虚拟机 735 和 738 发送的分组的指令。一些实施例的 OVS 守护进程 765 通过 OpenFlow 协议与网络控制器 780 通信，而其它实施例则使用不同的通信协议以传送物理控制平面数据。另外，在一些实施例中，在网络控制器 780 将配置信息发送到 OVS DB 守护进程之后，OVS 守护进程 765 从 OVS DB 守护进程 767 检索配置信息。

[0079] 在一些实施例中，OVS DB 守护进程 767 也在用户空间 721 中运行。一些实施例的 OVS DB 守护进程 767 与网络控制器 780 通信，以便配置 OVS 交换元件（例如，OVS 守护进程 765 和 / 或 OVS 内核模块 745）。例如，OVS DB 守护进程 767 从网络控制器 780 接收配置信息，并将该配置信息存储在一组数据库中。在一些实施例中，OVS DB 守护进程 767 通过数据库通信协议与网络控制器 780 通信。在一些情况下，OVS DB 守护进程 767 可以从 OVS 守护进程 765 接收对于配置信息的请求。在这些情况下，OVS DB 守护进程 767（例如，从一组数据库）检索所请求的配置信息，并将该配置信息发送到 OVS 守护进程 765。

[0080] OVS 守护进程 765 包括 OpenFlow 协议模块 770 和流处理器 775。OpenFlow 协议模块 770 与网络控制器 780 通信，以从网络控制器 780 接收用于配置软件交换元件的配置信

息（例如，流条目）。当模块 770 从网络控制器 780 接收到配置信息时，它将该配置信息转换成流处理器 775 可理解的信息。

[0081] 流处理器 775 管理用于处理和路由分组的规则。例如，流处理器 775 存储从 OpenFlow 协议模块 770 接收的规则（例如，存储在诸如盘驱动器的存储介质中）。在一些实施例中，规则被存储为一组流表，每个流表包括一组流条目。流处理器 775 处理集成网桥 (integration bridge)750（下面描述）对其不具有匹配规则的分组。在这样的情况下，流处理器 775 将分组与其存储的规则匹配。当分组与规则匹配时，流处理器 775 将所匹配的规则和分组发送到集成网桥 750 以供集成网桥 750 处理。这样，当集成网桥 750 接收到与所产生的规则匹配的类似分组时，分组将在集成网桥 750 中与所产生的精确匹配规则匹配，并且流处理器 775 将不必对分组进行处理。

[0082] 在一些实施例中，流处理器 775 可能不具有分组所匹配的规则。在这样的情况下，流处理器 775 的一些实施例将分组发送到用于对边缘交换元件不能处理的分组进行处理的另一受管理交换元件（例如，池节点）。然而，在其它情况下，流处理器 775 可能已从网络控制器 780 接收到包罗万象的规则 (catchall rule)，其当分组所匹配的规则不存在于流处理器 775 中时丢弃分组。

[0083] 如图 7 中所示，内核 720 包括超管理器网络堆栈 (stack)740 和 OVS 内核模块 745。在一些实施例中，超管理器网络堆栈 740 是互联网协议 (IP) 网络堆栈。超管理器网络堆栈 740 对从 OVS 内核模块 745 以及 PIF 网桥 755 和 760 接收的 IP 分组进行处理和路由。当对去往主机 700 外部的网络主机的分组进行处理时，超管理器网络堆栈 740 确定应将分组发送到物理接口 (PIF) 网桥 755 和 760 中的哪个。

[0084] OVS 内核模块 745 对网络数据（例如，分组）进行处理，并且在主机 700 上运行的 VM 与主机 700 外部的网络主机之间对网络数据进行路由（例如，通过 NIC710 和 715 接收的网络数据）。在一些实施例中，OVS 内核模块 745 实现用于一个或多个逻辑网络的物理控制平面的转发表。为了便于网络数据的处理和路由，OVS 内核模块 745 与 OVS 守护进程 765 通信（例如，以从 OVS 守护进程 765 接收流条目）。在一些实施例中，OVS 内核模块 745 包括网桥接口（未示出），其允许超管理器网络堆栈 740 将分组发送到 OVS 内核模块 745 以及从 OVS 内核模块 745 接收分组。

[0085] 图 7 示出了包括集成网桥 750 以及 PIF 网桥 755 和 760 的 OVS 内核模块 745。在一些实施例中，OVS 内核模块 745 包括用于硬件 705 中的每个 NIC 的 PIF 网桥。在其它实施例中，OVS 内核模块 745 中的 PIF 网桥可以与硬件 705 中的多于一个的 NIC 交互。PIF 网桥 755 和 760 在超管理器网络堆栈 740 与主机 700 外部的网络主机之间路由网络数据（即，通过 NIC710 和 715 接收的网络数据）。

[0086] 集成网桥 750 对从超管理器网络堆栈 740、VM735 和 738（例如，通过 VIF) 以及 PIF 网桥 755 和 760 接收的分组进行处理和路由。在一些实施例中，集成网桥 750 存储流处理器 775 中所存储的规则（和 / 或从存储在流处理器 775 中的规则导出的规则）的子集，集成网桥 750 当前正在使用或者最近使用所存储的规则的子集来对分组进行处理和转发。

[0087] 在一些实施例中，一些实施例的流处理器 775 负责管理集成网桥 750 中的规则。在一些实施例中，集成网桥 750 仅存储活动的规则。流处理器 775 监视存储在集成网桥 750 中的规则，并移除对于限定的时间量（例如，1 秒、3 秒、5 秒、10 秒等）已没有被访问的活动

的规则。以这种方式，流处理器 775 管理集成网桥 750，使得集成网桥 750 存储正被使用或最近已被使用的规则。

[0088] 尽管图 7 示出了一个集成网桥，但是 OVS 内核模块 745 可以包括多个集成网桥。例如，在一些实施例中，OVS 内核模块 745 包括用于在软件交换元件所属的受管理网络上实现的每个逻辑交换元件的单独的集成网桥。也就是说，OVS 内核模块 745 具有用于在受管理网络上实现的每个逻辑交换元件的相应的集成网桥。

[0089] 以上描述涉及一些实施例的受管理软件交换元件的转发功能。正如软件交换元件包括实现控制平面的用户空间组件 (OVS 守护进程 765) 和实现数据平面的内核组件 (OVS 内核模块 745) 那样，一些实施例的分布式中间盒元件包括在用户空间中操作的控制平面组件 (中间盒守护进程 790) 和在内核中操作的数据平面组件 (中间盒内核模块 795)。

[0090] 如所示的，中间盒守护进程 790 包括中间盒配置接收器 791 和中间盒配置编译器 792。中间盒配置接收器 791 与网络控制器 780 通信，以便接收对于中间盒的配置、以及切分信息。在一些实施例中，中间盒配置是描述中间盒分组处理规则的一组记录 (例如，以与 OVS 守护进程接收的流条目记录相同的形式)。例如，防火墙配置包括描述何时丢弃分组、允许分组等的一组分组处理规则 (类似于 ACL 条目，但是还包括作为决策中的因素的 TCP 连接状态)。源网络地址转换配置包括虚拟机的一组隐藏 IP 地址，其应被转换器映射到显露的 IP 地址。在一些实施例中，负载均衡器配置包括显露的 IP 地址到若干不同的隐藏虚拟机地址的网络地址转换映射、以及用于确定应将新的 TCP 连接发送到若干机器中的哪个的负载均衡 (调度) 算法。

[0091] 如上所述，切分信息将标识符分配给分布式中间盒元件要执行的特定中间盒实例。在一些实施例中，将标识符绑定到特定租户的逻辑网络中的特定逻辑中间盒。也就是说，当特定的逻辑网络包括具有不同处理规则的若干不同中间盒时，中间盒守护进程 790 将创建若干中间盒实例。这些实例中的每一个利用发送到中间盒的分组上的不同切分标识符来识别。另外，在一些实施例中，中间盒守护进程 790 为这些实例中的每一个分配特定的内部标识符，中间盒在其内部处理中使用该特定的内部标识符 (例如，以便保持跟踪活动的 TCP 连接)。

[0092] 中间盒守护进程 790 还包括中间盒配置编译器 792。在一些实施例中，中间盒配置编译器 792 接收第一语言的特定中间盒实例的中间盒配置 (例如，分组处理、修改或分析规则)，并将这些编译成对于中间盒的内部处理更优化的第二语言的一组规则。中间盒配置编译器 792 将经编译的分组处理规则发送到中间盒内核模块 795 的中间盒处理器 796。

[0093] 中间盒内核模块 795 对从在主机 700 上运行的 VM 发送的和 / 或发送到在主机 700 上运行的 VM 的分组进行处理，以便确定是否允许分组通过、丢弃分组等。如所示的，中间盒内核模块 795 包括执行这些功能的中间盒处理器 795。中间盒处理器 795 从中间盒配置编译器 792 接收用于特定的中间盒实例的经转换的中间盒规则。在一些实施例中，这些转换的中间盒规则指定中间盒内的分组处理管道 (pipeline)。

[0094] 为了从受管理交换元件接收分组，一些实施例的中间盒处理器 796 连接到 OVS 内核模块的集成网桥 750 上的软件端口抽象。通过集成网桥上的该端口，受管理交换元件将分组发送到中间盒，并且在由中间盒处理之后从中间盒接收分组 (除非中间盒丢弃分组)。如所描述的，这些分组包括切分标识符标签，其被中间盒处理器 796 用于确定哪组编译的

分组处理规则应用于分组。

[0095] 图 7 中所示的分布式中间盒和软件交换元件的架构图是一个示例性配置。本领域普通技术人员将认识到其它的配置是可能的。例如，在一些实施例中，应用编译的分组处理规则的中间盒处理器位于用户空间 721 中，而不是内核 720 中。在这样的实施例中，内核显露用于用户空间完全控制的网络接口 710 和 715，使得中间盒处理器可以在与内核相比不损失速度的情况下在用户空间中执行其功能。

[0096] III. 网络控制系统

[0097] 以上节 1 描述了从完全分布式到完全集中式的不同的中间盒实现架构。如所提及的，在一些实施例中，可以通过网络控制系统来对这些中间盒进行预配置，该网络控制系统还用于对实现网络的逻辑转发元件的受管理交换元件进行预配置。在一些实施例中，网络控制系统是分层的一组网络控制器。

[0098] 图 8 示出一些实施例的网络控制系统 800，其用于配置受管理交换元件和分布式中间盒元件以便实现逻辑网络。如所示的，网络控制系统 800 包括输入转换控制器 805、逻辑控制器 810、物理控制器 815 和 820、主机 825-840、以及集中式中间盒 845。如所示的，主机 830-865 包括受管理交换元件和中间盒元件两者，其可以如以上在图 7 中所示那样实现。本领域普通技术人员将认识到，对于网络控制系统 800，各种控制器和主机的许多其它不同的组合是可能的。

[0099] 在一些实施例中，网络控制系统中的控制器中的每一个具有用作输入转换控制器、逻辑控制器和 / 或物理控制器的能力。可替代地，在一些实施例中，给定的控制器可以仅具有作为这些类型中的特定一种类型的控制器（例如，作为物理控制器）操作的功能。另外，不同组合的控制器可以在同一物理机器中运行。例如，输入转换控制器 805 和逻辑控制器 810 可以在用户与其交互的同一计算装置中运行。

[0100] 此外，图 8（以及后面的图 9）中所示的控制器中的每一个被示为单个控制器。然而，这些控制器中的每一个实际上可以是以分布式方式操作以执行逻辑控制器、物理控制器或输入转换控制器的处理的控制器集群。

[0101] 一些实施例的输入转换控制器 805 包括对从用户接收的网络配置信息进行转换的输入转换应用。例如，用户可以指定包括关于哪些机器属于哪个逻辑域中的指定的网络拓扑，诸如图 2 中所示的网络拓扑。这有效地指定逻辑数据路径集、或一组逻辑转发元件。对于逻辑交换机中的每一个，用户指定连接到逻辑交换机的机器（即，为逻辑交换机分配哪些逻辑端口）。在一些实施例中，用户还指定机器的 IP 地址。输入转换控制器 805 将键入的网络拓扑转换成描述网络拓扑的逻辑控制平面数据。例如，条目可以声明特定的 MAC 地址 A 位于特定的逻辑交换机的特定逻辑端口 X 处。

[0102] 在一些实施例中，每个逻辑网络由特定的逻辑控制器（例如，逻辑控制器 810）管控。一些实施例的逻辑控制器 810 将逻辑控制平面数据转换成逻辑转发平面数据，并将逻辑转发平面数据转换成通用控制平面数据。在一些实施例中，逻辑转发平面数据由在逻辑层次上描述的流条目组成。对于逻辑端口 X 处的 MAC 地址 A，逻辑转发平面数据可以包括指定如果分组的目的地与 MAC A 匹配，则将该分组转发给端口 X 的流条目。

[0103] 一些实施例的通用物理控制平面数据是这样的数据平面：即使当一些实施例的控制系统含有实现逻辑数据路径集的大量受管理交换元件（例如，数千个）时，该数据平面也

使得该控制系统能够扩展。通用物理控制平面对不同的受管理交换元件的共有特性进行抽象，以便在不考虑受管理交换元件的差异和 / 或受管理交换元件的位置详情的情况下表达物理控制平面数据。

[0104] 如所说明的，一些实施例的逻辑控制器 510 将逻辑控制平面数据转换成逻辑转发平面数据（例如，逻辑流条目），然后将逻辑转发平面数据转换成通用控制平面数据。在一些实施例中，逻辑控制器应用堆栈包括用于执行第一转换的控制应用和用于执行第二转换的虚拟化应用。在一些实施例中，这两个应用都使用用于将第一组表映射到第二组表的规则引擎。也就是说，将不同的数据平面表示为表（例如，nLog 表），并且控制器应用使用表映射引擎以在数据平面之间进行转换。

[0105] 物理控制器 815 和 820 中的每一个是一个或多个受管理交换元件（例如，位于主机内）的主装置。在该示例中，这两个物理控制器中的每一个是两个受管理交换元件的主装置。此外，物理控制器 815 是集中式中间盒 845 的主装置。在一些实施例中，物理控制器接收对于逻辑网络的通用物理控制平面信息，并将该数据转换成对于该物理控制器管理的特定的受管理交换机的定制 (customized) 物理控制平面信息。在其它实施例中，物理控制器将适当的通用物理控制平面数据传递给包括自己执行变换的能力的受管理交换机（例如，以在主机上运行的机箱 (chassis) 控制器的形式）。

[0106] 通用物理控制平面到定制物理控制平面转换涉及流条目中的各个数据的定制。对于以上提到的示例，通用物理控制平面将涉及若干流条目。第一条目声明，如果分组与特定的逻辑数据路径集匹配（例如，基于在特定的逻辑入站端口 (ingress port) 处接收到的分组），并且目的地地址与 MAC A 匹配，则将该分组转发给逻辑端口 X。在一些实施例中，该流条目在通用物理控制平面和定制物理控制平面中是相同的。附加流被产生以将物理入站端口（例如，主机的虚拟接口）与逻辑入站端口 X（用于从 MAC A 接收的分组）匹配、以及将逻辑端口 X 与物理受管理交换机的特定出站端口 (egress port) 匹配。然而，这些物理入站和出站端口特定于含有受管理交换元件的主机。这样，通用物理控制平面条目包括抽象物理端口，而定制物理控制平面条目包括所涉及的实际物理端口。

[0107] 在一些实施例中，网络控制系统还散播 (disseminate) 与逻辑网络的中间盒相关的数据。网络控制系统可以散播中间盒配置数据、以及与在受管理交换机处将分组发送到中间盒 / 从这些中间盒接收分组和在中间盒处将分组发送到受管理交换机 / 从这些受管理交换机接收分组相关的数据。

[0108] 如图 8 中所示，在一些实施例中，同一网络控制系统将数据分发给分布式中间盒和集中式中间盒两者。若干物理控制器用于散播分布式中间盒的配置，而一些实施例将特定的物理控制器分配给集中式中间盒设备。在这种情况下，分配物理控制器 815 散播集中式中间盒 845 的配置，而对于分布式中间盒的配置则通过物理控制器 815 和 820 两者来散播。

[0109] 为了并入中间盒，通过网络控制系统传播给受管理交换机的流条目将包括用于将适当的分组发送到适当的中间盒的条目（例如，指定具有特定子网中的源 IP 地址的分组被转发给特定的中间盒的流条目）。另外，用于受管理交换机的流条目将需要指定如何将这样的分组发送到中间盒。也就是说，一旦第一条目指定特定中间盒与其绑定的逻辑路由器的逻辑出站端口，就需要将该逻辑出站端口附连到中间盒的附加条目。

[0110] 对于集中式中间盒 845，这些附加条目将使逻辑路由器的逻辑出站端口与主机的特定物理端口（例如，物理网络接口）匹配，主机通过该特定物理端口连接到中间盒。另外，这些条目包括用于经由主机与中间盒之间的隧道将分组发送到集中式中间盒设备的封装信息。

[0111] 对于分布式中间盒，分组不必为了到达中间盒而实际离开主机。然而，受管理交换元件不过需要包括用于将分组发送到主机上的中间盒元件的流条目。这些流条目再次包括将逻辑路由器的逻辑出站端口映射到受管理交换元件通过其连接到中间盒的端口。然而，在这种情况下，中间盒附连到受管理交换元件中的端口的软件抽象，而不是主机的物理（或虚拟）接口。也就是说，在受管理交换元件内创建中间盒附连的端口。受管理交换元件中的流条目将分组发送到该端口，以便分组在主机内被路由到中间盒。

[0112] 对于分布式中间盒和集中式中间盒两者，在一些实施例中，受管理交换元件将切分信息添加到分组。本质上，该切分信息是指示应将分组发送到中间盒正在运行的（潜在）若干实例中的哪个的标签。因此，当中间盒接收到分组时，标签使得中间盒能够使用适当的一组分组处理、分析、修改等规则，以便对该分组执行其操作。一些实施例不是将切分信息添加到分组，而是定义受管理交换元件的用于每个中间盒实例的不同端口，并且基本上使用这些端口来切分去往防火墙的业务（在分布式情况下），或者连接到集中式设备的不同端口以区分实例（在集中式情况下）。

[0113] 以上描述了将转发数据传播给受管理交换元件。另外，一些实施例使用网络控制系统来将配置数据传播给中间盒。图 9 概念性地示出了通过一些实施例的网络控制系统的数据传播。该图的左侧是对于实现逻辑网络的受管理交换元件的数据流，而该图的右侧示出两个中间盒配置数据以及网络附连和切分数据对于中间盒的传播。

[0114] 在左侧，输入转换控制器 805 通过 API 接收网络配置，其被变换成逻辑控制平面数据。该网络配置数据包括逻辑拓扑，诸如图 2 中所示的逻辑拓扑。另外，一些实施例的网络配置数据包括指定将哪些分组发送到中间盒的路由策略。当中间盒位于两个逻辑转发元件之间（例如，逻辑路由器与逻辑交换机之间）的逻辑线上时，那么通过该逻辑线发送的所有分组将自动地被转发给中间盒。然而，对于带外 (out-of-band) 中间盒，诸如网络架构 200 中的中间盒，逻辑路由器将仅当用户指定特定的策略时才将分组发送到中间盒。

[0115] 鉴于路由器和交换机通常将根据分组的目的地地址（例如，MAC 地址或 IP 地址）来转发分组，策略路由允许基于分组所存储的其它信息（例如，源地址、源地址和目的地地址的组合等）做出转发决策。例如，用户可以指定应将具有特定子网中的源 IP 地址或者具有与特定的一组子网不匹配的目的地 IP 地址的所有分组转发给中间盒。

[0116] 如所示的，逻辑控制平面数据被逻辑控制器 810（具体地，被该逻辑控制器的控制应用）变换成逻辑转发平面数据，并然后随后被（该逻辑控制器的虚拟化应用）变换成通用物理控制平面数据。在一些实施例中，这些变换产生流条目（在逻辑转发平面上），然后通过逻辑数据路径集来添加匹配（在通用物理控制平面上）。通用物理控制平面还包括用于将通用物理入站端口（即，非特定于任何特定的物理主机的端口的通用抽象）映射到逻辑入站端口、以及用于将逻辑出站端口映射到通用物理出站端口的附加流条目。例如，对于映射到集中式中间盒，通用物理控制平面上的流条目将包括当路由策略匹配时将分组发送到中间盒所连接的逻辑端口的转发决策、以及逻辑端口到连接到中间盒的主机的通用物理

端口的映射。

[0117] 如所示的，物理控制器 815（若干物理控制器中的一个）将通用物理控制平面数据转换成用于它管理的特定受管理交换元件 830-840 的定制物理控制平面数据。该变换涉及用通用物理控制平面数据中的通用抽象替换特定的数据（例如，特定的物理端口）。例如，在以上段落的示例中，端口集成条目被配置为指定适合于特定中间盒配置的物理层端口。如果防火墙作为主机上的虚拟机运行，则该端口可以是虚拟 NIC，或者当防火墙作为虚拟机上的超管理器内的过程（例如，守护进程）运行时，该端口可以是前面描述的受管理交换元件内的软件端口抽象。在一些实施例中，对于后一种情况，该端口是 IPC 信道或类似 TUN/TAP 装置的接口。在一些实施例中，受管理交换元件包括对于防火墙模块的一个特定的端口抽象，并且将该信息发送到物理控制器，以便物理控制器定制物理控制平面流。另一方面，对于将分组发送到集中式中间盒 845 的流条目，插入的端口将是受管理交换元件在其上操作的特定主机的实际物理端口。

[0118] 另外，在一些实施例中，物理控制器添加指定特定于中间盒的切分信息的流条目。例如，对于特定的受管理交换元件，流条目可以指定在将分组发送到特定的防火墙之前将特定的标签（例如，VLAN 标签或类似标签）添加到分组。该切分信息使得中间盒能够接收分组并且识别其若干独立实例中的哪个应对分组进行处理。

[0119] 受管理交换元件 825（由物理控制器 815 管理的若干 MSE 中的一个）执行定制物理控制平面数据到物理转发平面数据的转换。在一些实施例中，物理转发平面数据是存储在交换元件（物理路由器或交换机或软件交换元件）内的流条目，交换元件实际上将所接收的分组与这些流条目匹配。

[0120] 图 9 的右侧示出了传播给中间盒（集中式或分布式中间盒）、而不是受管理交换元件的两组数据。这些组数据中的第一组数据是包括指定特定的逻辑中间盒的操作的各种规则的实际中间盒配置数据。该数据可以通过特定于中间盒实现的 API、在输入转换控制器 805 或不同的输入接口处接收。在一些实施例中，不同的中间盒实现将具有呈现给用户的不同接口（即，用户对于不同的特定中间盒将必须键入不同格式的信息）。如所示的，用户键入中间盒配置，其被中间盒 API 转换成中间盒配置数据。

[0121] 在一些实施例中，中间盒配置数据是一组记录，其中每个记录指定一个特定的规则。在一些实施例中，这些记录在格式上与传播给受管理交换元件的流条目类似。事实上，一些实施例使用控制器上的相同应用来传播关于流条目的防火墙配置记录，并且对于这些记录使用相同的表映射语言（例如，nLog）。

[0122] 在一些实施例中，中间盒配置数据不被逻辑控制器或物理控制器转换，而在其它实施例中，逻辑控制器和 / 或物理控制器至少执行中间盒配置数据记录的最少转换。因为许多中间盒分组处理、修改和分析规则对分组的 IP 地址（或 TCP 连接状态）进行操作，并且发送到中间盒的分组将使该信息显露（即，不被封装在逻辑端口信息内），所以中间盒配置不需要从逻辑数据平面到物理数据平面的转换。因此，相同的中间盒配置数据被从输入转换控制器 805（或其它接口）传递到逻辑控制器 810、再传递到物理控制器 815。

[0123] 在一些实施例中，逻辑控制器 810 存储逻辑网络和该物理网络的物理实现的描述。逻辑控制器接收对于分布式中间盒的一个或多个中间盒配置记录，并识别各个节点（即，主机）中的哪个将需要接收配置信息。在一些实施例中，整个中间盒配置被分发给所

有主机处的中间盒元件，所以逻辑控制器识别至少一个虚拟机驻留在其上的、其分组需要使用中间盒的所有机器。这可以是网络中的所有虚拟机（例如，就图 2 中所示的中间盒而言）、或网络中的虚拟机的子集（例如，当防火墙仅应用于网络内的特定域的业务时）。一些实施例以每记录为基础对将配置数据发送到哪些主机做出决策。也就是说，每个特定的规则可以仅应用于虚拟机的子集，并且仅运行这些虚拟机的主机需要接收记录。

[0124] 一旦逻辑控制器识别了接收记录的特定节点，逻辑控制器就识别管理这些特定节点的特定物理控制器。如所提及的，每个主机具有分配的主物理控制器。因此，如果逻辑控制器仅将第一主机和第二主机识别为配置数据的目的地，则对于这些主机的物理控制器将被识别为从逻辑控制器接收数据（并且其它物理控制器将不接收该数据）。对于集中式中间盒，逻辑控制器仅需要识别管理实现中间盒的设备的（单个）物理控制器。

[0125] 为了向主机供给中间盒配置数据，一些实施例的逻辑控制器将数据推送到物理控制器（通过使用访问逻辑控制器中的表映射引擎的输出的导出模块）。在其它实施例中，物理控制器从逻辑控制器的导出模块请求配置数据（例如，响应于配置数据可用的信号）。

[0126] 物理控制器将数据传递给它们管理的主机上的中间盒元件，尽管它们传递物理控制平面数据。在一些实施例中，中间盒配置和物理控制平面数据被发送到在主机上运行的同一数据库，并且受管理交换元件和中间盒模块从该数据库检索适当的信息。类似地，对于集中式中间盒 845，物理控制器 815 将中间盒配置数据传递给中间盒设备（例如，传递给用于存储配置数据的数据库）。

[0127] 在一些实施例中，中间盒对配置数据进行转换。以特定语言表达分组处理、分析、修改等规则的中间盒配置数据将被接收。一些实施例的中间盒（分布式和 / 或集中式）将这些规则编译成更优化的分组分类规则。在一些实施例中，该转化 (transformation) 类似于物理控制平面到物理转发平面数据转换。当分组被中间盒接收时，它应用编译的优化规则，以便有效地、快速地对该分组执行其操作。

[0128] 除了中间盒配置规则之外，中间盒模块接收切分和 / 或附连信息 (attachment information)，以便从受管理交换元件接收分组以及将分组发送到受管理交换元件。该信息对应于发送到受管理交换元件的信息。如所示的，在一些实施例中，物理控制器 815 产生对于中间盒的切分和 / 或附连信息（即，该信息不是在网络控制系统的输入或逻辑控制器层次上产生的）。

[0129] 对于分布式中间盒，在一些实施例中，物理控制器从受管理交换元件本身接收关于中间盒连接的受管理交换元件的软件端口的信息，然后将该信息向下传递给中间盒。然而，在其它实施例中，直接在主机内的中间盒模块与受管理交换元件之间对该端口的使用订约 (contract)，使得中间盒不需要从物理控制器接收附连信息。在一些这样的实施例中，受管理交换元件而是将该信息发送到物理控制器，以便该物理控制器定制用于从中间盒接收分组以及将分组发送到中间盒的通用物理控制平面流条目。

[0130] 对于集中式中间盒，一些实施例将隧道附连数据提供给中间盒。在一些实施例中，中间盒将需要知道各个主机将使用的将分组发送到中间盒的隧道封装的类型。在一些实施例中，中间盒具有接受的隧道协议（例如，STT、GRE 等）的列表，并且在一个（多个）受管理交换元件与中间盒之间协调所选择的协议。隧道协议可以由用户作为中间盒配置的一部分键入，或者在不同实施例中，可以由网络控制系统自动地确定。如以上参照图 4 所描述的，

除了连接到主机之外，将在集中式中间盒与它在处理之后将分组发送到的池节点之间设置隧道。

[0131] 在一些实施例中，由物理控制器产生的切分信息由要用于特定逻辑网络的中间盒实例的标识符组成。在一些实施例中，如所描述的，中间盒被虚拟化以供多个逻辑网络使用，而不管中间盒在主机上操作、还是作为集中式设备操作。当中间盒从受管理交换元件接收到分组时，在一些实施例中，分组包括前置的标签（例如，类似于VLAN标签），其识别在处理分组中所使用的中间盒实例中的特定一个（即，特定的配置的一组规则）。

[0132] 如图9中所示，中间盒将该切分信息转换成内部切分绑定。在一些实施例中，中间盒使用它自己的内部标识符（不同于前置到分组的标签），以便识别中间盒内的状态（例如，活动的TCP连接、关于各个IP地址的统计等）。当接收到创建新的中间盒实例和用于该新的实例的外部标识符（其在分组上使用）的指令时，一些实施例自动地创建新的中间盒实例，并为该实例分配内部标识符。另外，中间盒存储对于该实例的将外部切分标识符映射到内部切分标识符的绑定。

[0133] 以上附图示出了各种物理网络控制器和逻辑网络控制器。图10示出了网络控制器（例如，逻辑控制器或物理控制器）1000的示例架构。一些实施例的网络控制器使用表映射引擎来将来自输入的一组表的数据映射到输出的一组表中的数据。控制器中的输入的这组表包括要被映射到逻辑转发平面（LFP）数据的逻辑控制平面（LCP）数据、要被映射到通用物理控制平面（UPCP）数据的LFP数据、和/或要被映射到定制物理控制平面（CPCP）数据的UPCP数据。输入的这组表还可以包括要被发送到另一控制器和/或分布式中间盒实例的中间盒配置数据。如所示的，网络控制器1000包括输入表1015、规则引擎1010、输出表1020、导入器1030、导出器1035、转换器1035以及持久数据储存器（PTD）1040。

[0134] 在一些实施例中，取决于控制器1000在网络控制系统中的作用，输入表1015包括具有不同类型的数据的表。例如，当控制器1000用作用于用户的逻辑转发元件的逻辑控制器时，输入表1015包括用于逻辑转发元件的LCP数据和LFP数据。当控制器1000用作物理控制器时，输入表1015包括LFP数据。输入表1015还包括从用户或另一控制器接收的中间盒配置数据。中间盒配置数据与识别中间盒要被集成到的逻辑交换元件的逻辑数据路径集参数相关联。

[0135] 除了输入表1015之外，控制应用1000包括其它杂项表（未示出），规则引擎1010使用这些杂项表来采集用于其表映射操作的输入。这些杂项表包括常数表，其存储规则引擎1010执行其表映射操作所需的常数的定义值（例如，值0、用于重发的调度端口号等）。杂项表还包括函数表，其存储规则引擎1010使用其来计算填充输出表1025的值的函数。

[0136] 规则引擎1010执行指定用于将输入数据变换成输出数据的一种方式的表映射操作。每当输入表中的一个被修改（被称为输入表事件）时，规则引擎就执行可以导致一个或多个输出表中的一个或多个数据元组的修改的一组表映射操作。

[0137] 在一些实施例中，规则引擎1010包括事件处理器（未示出）、若干查询计划(plan)（未示出）、以及表处理器（未示出）。每个查询计划是指定当发生输入表事件时要执行的一组联合操作的一组规则。规则引擎1010的事件处理器检测每个这样的事件的发生。在一些实施例中，事件处理器向输入表注册回调以用于通知输入表1015中的记录的变化，并且当其记录中的一个已改变时，通过从输入表接收通知来检测输入表事件。

[0138] 响应于检测的输入表事件，事件处理器 (1) 选择适合于检测的表事件的查询计划，并且 (2) 引导表处理器执行该查询计划。为了执行查询计划，在一些实施例中，表处理器执行由查询计划指定的联合操作，以生成表示来自一个或多个输入表和杂项表的一组或多组数据值的一个或多个记录。一些实施例的表处理器然后 (1) 执行从通过联合操作生成的一个（多个）记录选择数据值的子集的选择操作，并且 (2) 将所选择的数据值的子集写入到一个或多个输出表 1020 中。

[0139] 一些实施例使用数据日志数据库语言的变型来允许应用开发者创建用于控制器的规则引擎，并从而指定控制器将逻辑数据路径集映射到受控物理交换底层结构 (infrastructure) 的方式。数据日志数据库语言的该变型在这里被称为 nLog。像数据日志那样，nLog 提供允许开发者指定当发生不同的事件时要执行的不同操作的一些声明规则和运算符。在一些实施例中，nLog 提供由数据日志提供的运算符的有限子集以便提高 nLog 的运算速度。例如，在一些实施例中，nLog 仅允许 AND 运算符在声明规则中的任何一个中使用。

[0140] 通过 nLog 指定的声明规则和操作然后由 nLog 编译器编译成大得多的一组规则。在一些实施例中，该编译器将意在于处理事件的每个规则转换成若干组数据库联合操作。更大的这组规则共同形成被称为 nLog 引擎的表映射规则引擎。

[0141] 一些实施例将规则引擎对于输入事件执行的第一联合操作指派为基于逻辑数据路径集参数。该指派确保当规则引擎已开始与不由控制器管理的逻辑数据路径集（即，逻辑网络）相关的一组联合操作时，规则引擎的联合操作失败并且立即终止。

[0142] 像输入表 1015 那样，取决于控制器 1000 的作用，输出表 1020 包括具有不同类型的数据的表。当控制器 1000 用作逻辑控制器时，输出表 1015 包括用于逻辑交换元件的 LFP 数据和 UPCP 数据。当控制器 1000 用作物理控制器时，输出表 1020 包括 CPCP 数据。像输入表那样，输出表 1015 还可以包括中间盒配置数据。此外，当控制器 1000 用作物理控制器时，输出表 1015 可以包括切分标识符。

[0143] 在一些实施例中，可以将输出表 1020 分组为若干不同的类别。例如，在一些实施例中，输出表 1020 可以是规则引擎 (RE) 输入表和 / 或 RE 输出表。当输出表中的变化使规则引擎检测到需要执行查询计划的输入事件时，输出表是 RE 输入表。输出表还可以是产生使规则引擎执行另一查询计划的事件的 RE 输入表。当输出表中的变化使导出器 1025 将该变化导出到另一控制器或 MSE 时，输出表是 RE 输出表。输出表可以是 RE 输入表、RE 输出表、或 RE 输入表和 RE 输出表两者。

[0144] 导出器 1025 检测输出表 1020 的 RE 输出表的变化。在一些实施例中，导出器向 RE 输出表注册回调以用于通知 RE 输出表的记录的变化。在这样的实施例中，当导出器 1025 从 RE 输出表接收到其记录中的一个已改变的通知时，导出器 1025 检测到输出表事件。

[0145] 响应于检测的输出表事件，导出器 1025 获取修改的 RE 输出表中的每个修改的数据元组，并将该修改的数据元组传播给一个或多个其它控制器或者一个或多个 MSE。当将输出表记录发送到另一控制器时，在一些实施例中，导出器使用单个通信的信道（例如，RPC 信道）来发送记录中所含有的数据。当将 RE 输出表记录发送到 MSE 时，在一些实施例中，导出器使用两个信道。一个信道通过使用交换机控制协议（例如，OpenFlow）建立以将流条目写入到 MSE 的控制平面中。另一信道通过使用数据库通信协议（例如，JSON）建立以

发送配置数据（例如，端口配置、隧道信息）。

[0146] 在一些实施例中，控制器 1000 不在输出表 1020 中保存对于控制器不负责管理的逻辑数据路径集（即，对于由其它逻辑控制器管理的逻辑网络）的数据。然而，这样的数据由转换器 1035 转换成可以存储在 PTD1040 中的格式，并然后被存储在 PTD 中。PTD1040 将该数据传播给一个或多个其它控制器的 PTD，使得负责管理逻辑数据路径集的这些其它控制器可以对该数据进行处理。

[0147] 在一些实施例中，为了数据的弹性，控制器还将存储在输出表 1020 中的数据引入 PTD。因此，在一些实施例中，控制器的 PTD 具有用于网络控制系统所管理的所有逻辑数据路径集的所有配置数据。也就是说，每个 PTD 含有所有用户的逻辑网络的配置的全局视图。

[0148] 导入器 1030 与多个不同的输入数据源接合，并且使用输入数据来修改或创建输入表 1010。一些实施例的导入器 1020 从另一控制器接收输入数据。导入器 1020 还与 PTD1040 接合，使得通过 PTD 从其它控制器实例接收的数据被转换并用作输入数据以修改或创建输入表 1010。而且，导入器 1020 还检测输出表 1030 中随 RE 输入表的变化。

[0149] IV. 若干中间盒的示例实现

[0150] 以上描述了用于实现和配置分布式中间盒和集中式中间盒两者的各个原理。以上在图 2 中所示的示例是具有仅单个中间盒的简化示例。另一方面，图 11 概念性地示出了涉及许多中间盒的更复杂的逻辑网络拓扑 1100。

[0151] 逻辑网络 1100 包括三个逻辑 L2 域 :连接到第一逻辑 L2 交换机 1140 的 web 服务器 1105-1115、连接到第二逻辑 L2 交换机 1145 的应用服务器 1120 和 1125、以及连接到第三逻辑交换机 1150 的数据服务器 1130 和 1135。这些逻辑交换机 1140-1150 中的每一个连接到逻辑路由器 1155（通过各个中间盒）。

[0152] 每个逻辑交换机与逻辑路由器 1155 之间是负载均衡器，以便该负载均衡器对传入到特定逻辑 L2 域的业务进行调度。也就是说，第一负载均衡器 1160 执行目的地网络地址转换 (D-NAT) 以均衡三个 web 服务器 1105-1115 之间的业务，第二负载均衡器 1165 执行 D-NAT 以均衡两个应用服务器 1120 与 1125 之间的业务，第三负载均衡器 1170 执行 D-NAT 以均衡两个数据服务器 1130 与 1135 之间的业务。另外，逻辑路由器 1155 与第二逻辑交换机 1145 之间是防火墙 1175。

[0153] 逻辑路由器 1155 还将三个逻辑 L2 域连接到外部网络 1195，客户端请求可以从该外部网络 1195 进入网络中。另外，三个中间盒挂在 (hang off)L3 路由器 1155 上以用于对受管理网络与外部网络之间的业务进行处理。这些中间盒包括防火墙 1180 和源 NAT1185，防火墙 1180 用于对传入业务进行处理，源 NAT1185 用于将传出业务的真实 IP 地址变换成一个或多个虚拟 IP 地址。这些中间盒有效地位于受管理网络与外部网络之间 ;然而，因为物理实现涉及将分组发送到中间盒、并然后从中间盒接收回分组（以发送到外部网络 1195 或适当的主机），所以逻辑拓扑将这些中间盒示出为挂在路由器上的带外中间盒。最后，IDS1190 也挂在逻辑路由器 1155 上。在网络 1100 中，逻辑路由器将所有处理的分组的副本转发给 IDS1190 以进行分析。

[0154] 图 12 概念性地示出了网络 1100 在托管的虚拟化的环境中的一种特定的物理实现 1200。如所示的，七个虚拟机 1105-1135 分布在五个不同的主机 1205-1225 上。主机中的一些仅托管一个虚拟机，而其它主机则托管两个 VM。主机 1205-1225 彼此连接以及连接到

池节点 1230，其连接到网关（也被称为扩展器）1235。网关 1235 将受管理网络 1200 连接到外部网络 1240（例如，互联网、不同的受管理网络、外部私有网络等）。尽管该示例将网关 1230 示出为仅通过池节点 1230 连接到主机 1205-1225，但是一些实施例实现网关与主机之间的直接连接。

[0155] 如所示的，每个主机 1205-1225 以及池节点 1230 和网关 1235 包括受管理交换元件。所有的主机 1205-1225 被配置为包括对于逻辑路由器 1155 以及逻辑交换机 1140-1150 的流条目。因此，包括应用服务器 1125 以及 web 服务器 1110 的第二主机 1210 和仅包括数据服务器 1130 的第五主机 1225 实现相同的受管理交换元件。网关 1235 中的受管理交换元件实现逻辑网络 1100 的逻辑路由器 1155 以及所有三个逻辑交换机 1140-1150。在一些实施例中，池节点也是实现 L3 路由器 1155 和所有三个逻辑交换机 1140-1150 的受管理交换元件。

[0156] 在实现 1200 中，逻辑中间盒中的一些是分布式的，而其它逻辑中间盒则是集中式的。例如，入侵检测服务 1190 被实现为集中式 IDS 设备 1245。主机 1205-1225 中的每一个如网关 1235 那样直接连接到 IDS 设备 1245。如所示的，这些机器仅将分组发送到 IDS 设备，并且不接收回分组。这是因为 IDS 仅接收重复分组（从网关 1235 传入的分组和从主机 1205-1225 传出的分组），并执行分析以检测威胁，但是在分析这些分组之后不将它们发送到任何地方。

[0157] S-NAT 中间盒 1185 在主机中的每一个之间分布（例如，作为在超管理器内运行的守护进程），因为所有的虚拟机可以将需要 IP 地址转换以将真实 IP 地址隐藏在虚拟 IP 地址后面的分组发送到外部网络。防火墙 1175 也是分布式的，但是仅在主机 1210 和 1220 上实现，因为这些是托管在该防火墙后面的应用服务器虚拟机 1120 和 1125 的节点。

[0158] 三个负载均衡器 1160-1170 在各个主机 1205-1225 以及网关 1235 上实现。如所示的，负载均衡器在主机中的每一个的负载均衡器元件内实现，使得该负载均衡器元件被虚拟化（即，切分）以实现七个不同的负载均衡器。鉴于防火墙 1175 位于应用服务器分别实现的主机处，每个特定的逻辑负载均衡器位于托管特定的负载均衡器不负责的机器的每个节点上。这是因为，例如，负载均衡器 1170 接收去往表示数据服务器 1130 和 1135 的虚拟 IP 地址的任何分组，确定将分组转发给这两个数据服务器中的哪个，然后修改目的地 IP 地址以反映所选择的数据服务器。因为只要有可能就在第一跳（分组源）处执行处理，在托管数据服务器的节点处不需要该功能（除非其它虚拟机也被托管），而是在托管可以将分组发送到数据服务器 1130 和 1135 的其它虚拟机的节点处需要该功能。因此，网关 1235 中的负载均衡器元件被切分以实现所有三个负载均衡器，因为从外部网络 1240 传入的分组可以去往三个逻辑 L2 域中的任何一个。

[0159] 另外，如所示的，一些实施例还在池节点 1230 和网关 1235 内实现用于逻辑网络的任何分布式中间盒。因为可能一开始难以确定在哪些位置需要哪些中间盒（并且用户随后可能修改路由策略、中间盒配置或网络架构），所以一些实施例不假定某些物理机器将不需要特定的中间盒。沿着相同的思路，一些实施例不将不同的中间盒分发给不同的主机子集。相反，在存在逻辑网络的每个主机处实现用于逻辑网络的所有中间盒。

[0160] 用于对外部网络 1180 与受管理网络之间的传入分组进行处理的防火墙 1180 在网关中的虚拟机（而不是在超管理器中运行的模块）中以集中式的方式实现。当网关接收到

传入分组时，在一些实施例中，它自动地将分组路由到防火墙 VM。尽管在该示例中，防火墙 VM 位于网关中，但是一些实施例将防火墙实现为主机（例如，与托管用户 VM 的主机不同的主机）中的虚拟机、或者使用防火墙设备来实现防火墙。

[0161] 为了配置网络 1100，在一些实施例中，如以上参照图 8 和图 9 所示的，用户将网络拓扑键入到逻辑控制器（例如，经由输入转换控制器）。在一些实施例中，用户键入交换机、路由器、中间盒和虚拟机之间的连接。基于各个网络组件的位置，输入转换控制器或逻辑控制器产生逻辑控制平面数据，并将该数据变换为逻辑转发平面中的流条目。然而，对于诸如防火墙 1180、S-NAT1185 以及 IDS1190 的中间盒，用户还必须键入指示何时将分组发送到这些组件的策略路由规则。例如，用于防火墙 1180 的路由策略将是发送具有逻辑网络外部的源 IP 的所有分组，而用于 S-NAT1185 的路由策略将是发送具有逻辑网络中的源 IP 的所有分组。逻辑控制器在将流条目变换成通用物理控制平面之后识别哪些物理控制器将接收哪些流条目，并然后分发这些流条目。物理控制器将特定的端口信息（除非主机包括执行通用物理控制平面到定制物理控制平面转换的机箱控制器）和其它定制添加到流条目，并将它们分发给受管理交换元件。

[0162] 另外，用户键入对于各个负载均衡器、防火墙等的中间盒配置。例如，该信息将包括用于不同负载均衡器中的每一个的调度算法、用于 S-NAT 的虚拟 IP 到真实 IP 映射、用于防火墙的分组处理规则等。用户通过用于各个中间盒的 API 来键入该信息，并且一些实施例将该信息变换成具有与流条目相同格式（例如，nLog）的记录。逻辑控制器识别需要将哪些中间盒配置发送到哪些主机或集中式中间盒设备（例如，对于防火墙 1175 的记录仅需要进入主机 1210 和 1220，而 S-NAT 记录进入所有五个主机 1205-1225）。逻辑控制器将记录分发给适当的物理控制器，其如上所述那样添加切分信息（并且，在一些情况下，隧道信息）并将该信息分发给中间盒。

[0163] 将参照从外部网络传入的分组、传出到外部网络的分组以及从一个 L2 域发送到另一个域的分组来描述网络的操作。当分组被从主机（例如，web 服务器）发送时，它首先到达在主机上运行的 MSE。该分组将首先进入逻辑交换机的逻辑 L2 处理，该逻辑交换机将分组发送到逻辑路由器（因为分组正在传出，所以它不需要被发送到本地负载均衡器）。除了将分组发送到主机上的 S-NAT 处理之外，逻辑路由器（也由主机处的 MSE 处理）将分组的副本发送到 IDS 设备 1245。S-NAT 处理修改源 IP 地址，并将新的分组返回给 MSE。在一些实施例中，如果分组是活动的 TCP 会话的一部分，则 S-NAT 可能已将流条目发送到 MSE，使得该 MSE 能够在不涉及 S-NAT 处理的情况下执行修改。MSE 实现的逻辑路由器然后将逻辑出站端口识别为面对外部网络的端口，其映射到将分组发送到池节点 1230 的物理端口。池节点将分组转发给网关 1235，其将分组送出到外部网络 1240。

[0164] 当在网关 1240 处从外部网络 1240 接收到分组时，网关中的交换元件处理首先将分组发送到防火墙虚拟机。如果防火墙不丢弃分组，则分组被返回到交换元件处理，其识别负载均衡器的正确切分，利用该切分信息对分组进行标记，并将该分组发送到负载均衡器处理。负载均衡器选择真实的目的地 IP，并发送具有被修改为反映所选择的目的地机器的 IP 地址的新分组。此刻，网关中的 MSE 将分组发送到正确的主机。如果目的地 VM 是应用服务器 1120 或 1125 中的一个，则网关的 MSE 中的逻辑路由器首先将分组发送到防火墙 1175 以进行处理，并然后在从防火墙元件接收回分组之后将该分组发送到正确的主机。MSE 然后

将该分组递送给目的地机器。

[0165] 从一个逻辑域行进到另一个域的分组将不需要行进通过网关 1235。分组最初在 MSE 处被接收，该 MSE 执行 L2 交换、并然后执行 L3 路由。逻辑路由器识别目的地域，并利用正确的负载均衡器切分信息对分组进行标记，然后将标记的分组发送到负载均衡器。负载均衡器修改目的地 IP，并将分组返回给 MSE，其然后将分组路由到正确的主机以用于递送给 VM。

[0166] V. 电子系统

[0167] 上述特征和应用中的许多被实现为被指定为记录在计算机可读存储介质（也被称为计算机可读介质）上的指令集的软件处理。当这些指令被一个或多个计算或处理单元（例如，一个或多个处理器、处理器的核或其它处理单元）执行时，它们使一个（多个）处理单元执行指令中所指示的动作。计算机可读介质的示例包括，但不限于，CD-ROM、闪存驱动器、随机存取存储器 (RAM) 芯片、硬盘驱动器、可擦式可编程只读存储器 (EPROM)、电可擦式可编程只读存储器 (EEPROM) 等。计算机可读介质不包括无线地或通过有线连接传递的载波和电信号。

[0168] 在本说明书中，术语“软件”意在于包括驻留在只读存储器中的固件或可被读取到存储器中以供处理器处理的存储在磁储存器中的应用。并且，在一些实施例中，在保留截然不同的软件发明的同时，多个软件发明可以被实现为更大程序的子部分。在一些实施例中，多个软件发明也可以被实现为单独的程序。最后，一起实现这里所描述的软件发明的单独的程序的任何组合在本发明的范围内。在一些实施例中，软件程序在被安装以在一个或多个电子系统上操作时定义实行和执行软件程序的操作的一个或多个特定的机器实现。

[0169] 图 13 概念性地示出了实现本发明的一些实施例的电子系统 1300。电子系统 1300 可以是计算机（例如，桌面计算机、个人计算机、平板计算机等）、服务器、专用交换机、电话、PDA、或任何其它种类的电子或计算装置。这样的电子系统包括各种类型的计算机可读介质以及用于各种其它类型的计算机可读介质的接口。电子系统 1300 包括总线 1305、一个（多个）处理单元 1310、系统存储器 1325、只读存储器 1330、永久存储装置 1335、输入装置 1340 以及输出装置 1345。

[0170] 总线 1305 共同表示通信地连接电子系统 1300 的许多内部装置的所有系统、外设和芯片组总线。例如，总线 1305 通信地将一个（多个）处理单元 1310 与只读存储器 1330、系统存储器 1325 以及永久存储装置 1335 连接。

[0171] 从这些各种存储器单元，一个（多个）处理单元 1310 检索执行的指令和处理的数据以便执行本发明的处理。在不同的实施例中，一个（多个）处理单元可以是单个处理器或多核处理器。

[0172] 只读存储器 (ROM) 1330 存储一个（多个）处理单元 1310 和电子系统的其它模块所需的静态数据和指令。另一方面，永久存储装置 1335 是读写存储器装置。该装置是即使当电子系统 1300 关闭时也存储指令和数据的非易失性存储器单元。本发明的一些实施例使用大容量存储装置（诸如磁盘或光盘以及其相应的盘驱动器）作为永久存储装置 1335。

[0173] 其它实施例使用可移动存储装置（诸如软盘、闪存装置等以及其相应的驱动器）作为永久存储装置。像永久存储装置 1335 那样，系统存储器 1325 是读写存储器装置。然而，与存储装置 1335 不同，系统存储器 1325 是易失性读写存储器，诸如随机存取存储器。系

统存储器 1325 存储处理器在运行时所需的指令和数据中的一些。在一些实施例中，本发明的处理被存储在系统存储器 1325、永久存储装置 1335 和 / 或只读存储器 1330 中。从这些各种存储器单元，一个（多个）处理单元 1310 检索执行的指令和处理的数据以便执行一些实施例的处理。

[0174] 总线 1305 还连接到输入装置 1340 和输出装置 1345。输入装置 1340 使得用户能够将信息和选择命令传送给电子系统。输入装置 1340 包括字母数字键盘和定点装置（也称为“光标控制装置”）、照相机（例如，网络摄像机）、麦克风或用于接收语音命令的类似装置等。输出装置 1345 显示由电子系统产生的图像，或者以其它方式输出数据。输出装置 1345 包括打印机和显示装置，诸如阴极射线管 (CRT) 或液晶显示器 (LCD)、以及扬声器或类似的音频输出装置。一些实施例包括诸如用作输入装置和输出装置两者的触摸屏的装置。

[0175] 最后，如图 13 中所示，总线 1305 还通过网络适配器（未示出）将电子系统 1300 耦合到网络 1365。以这种方式，计算机可以是以下网络的一部分 :计算机网络（诸如局域网（“LAN”））、广域网（“WAN”）、或内联网、或诸如互联网的网络的网络。电子系统 1300 的任何或所有组件可以与本发明结合使用。

[0176] 一些实施例包括电子组件，诸如将计算机程序指令存储在机器可读或计算机可读介质（可替代地被称为计算机可读存储介质、机器可读介质或机器可读存储介质）中的储存器、存储器和微处理器。这样的计算机可读介质的一些示例包括 RAM、ROM、只读紧凑盘 (CD-ROM)、可记录紧凑盘 (CD-R)、可重写紧凑盘 (CD-RW)、只读数字多功能盘（例如，DVD-ROM、双层 DVD-ROM）、各种可记录 / 可重写 DVD（例如，DVD-RAM、DVD-RW、DVD+RW 等）、闪存（例如，SD 卡、迷你 SD 卡、微 SD 卡等）、磁和 / 或固态硬盘驱动器、只读和可记录 **Blu-Ray®** 盘、超密度光盘、任何其它光或磁介质、以及软盘。计算机可读介质可以存储可由至少一个处理器单元执行并且包括用于执行各种操作的多个指令集的计算机程序。计算机程序或计算机代码的示例包括诸如由编译器生成的机器代码、以及包括由计算机、电子组件或微处理器通过使用解释器而执行的更高级代码的文件。

[0177] 尽管以上讨论主要指的是执行软件的微处理器或多核处理器，但是一些实施例由一个或多个集成电路（诸如专用集成电路 (ASIC) 或现场可编程门阵列 (FPGA)）执行。在一些实施例中，这样的集成电路执行存储在电路本身上的指令。另外，一些实施例执行存储在可编程逻辑器件 (PLD)、ROM 或 RAM 器件中的软件。

[0178] 如在本申请的本说明书和任何权利要求中所使用的，术语“计算机”、“服务器”、“处理器”和“存储器”全部指的是电子或其它技术装置。这些术语排除人或人的群组。为了本说明书的目的，术语显示器或显示意指在电子装置上显示。如在本申请的本说明书和任何权利要求中所使用的，术语“计算机可读介质”和“机器可读介质”完全限于以计算机可读的形式存储信息的有形的物理对象。这些术语排除任何无线信号、有线下载信号以及任何其它短暂信号。

[0179] 尽管已参照许多特定的细节描述了本发明，但是本领域普通技术人员将认识到，在不脱离本发明的精神的情况下，可以以其它特定的形式体现本发明。因此，本领域普通技术人员将理解，本发明不受前述说明性的细节限制，而是由所附的权利要求限定。

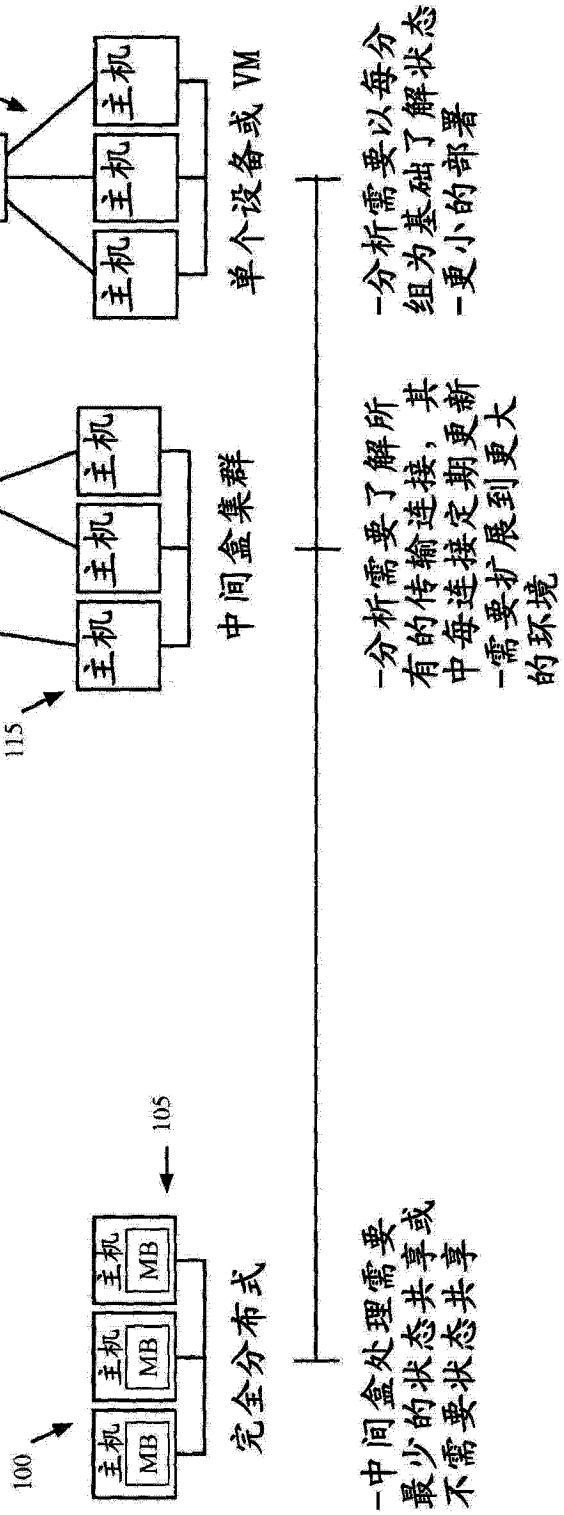

100
105
主机
MB
完全分布式
-中间盒处理需要最少的状态共享或不需要状态共享
115
中间盒集群
-分析需要了解所有的传输连接，其中每连接定期更新
-需要扩展到更大的环境
110
单个设备或 VM
-分析需要以每分组为基础了解状态
-更小的部署

图 1

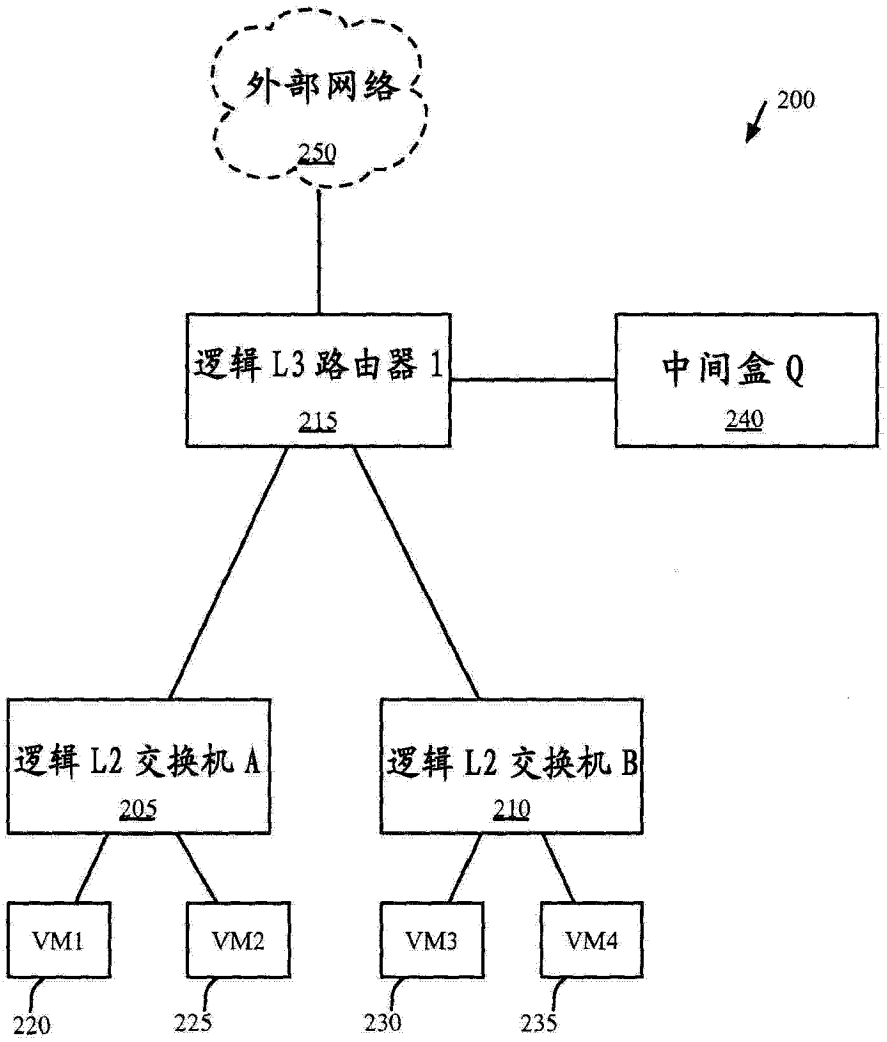

外部网络
250
200
逻辑 L3 路由器 1
215
中间盒 Q
240
逻辑 L2 交换机 A
205
逻辑 L2 交换机 B
210
VM1
VM2
VM3
VM4
220
225
230
235

图 2

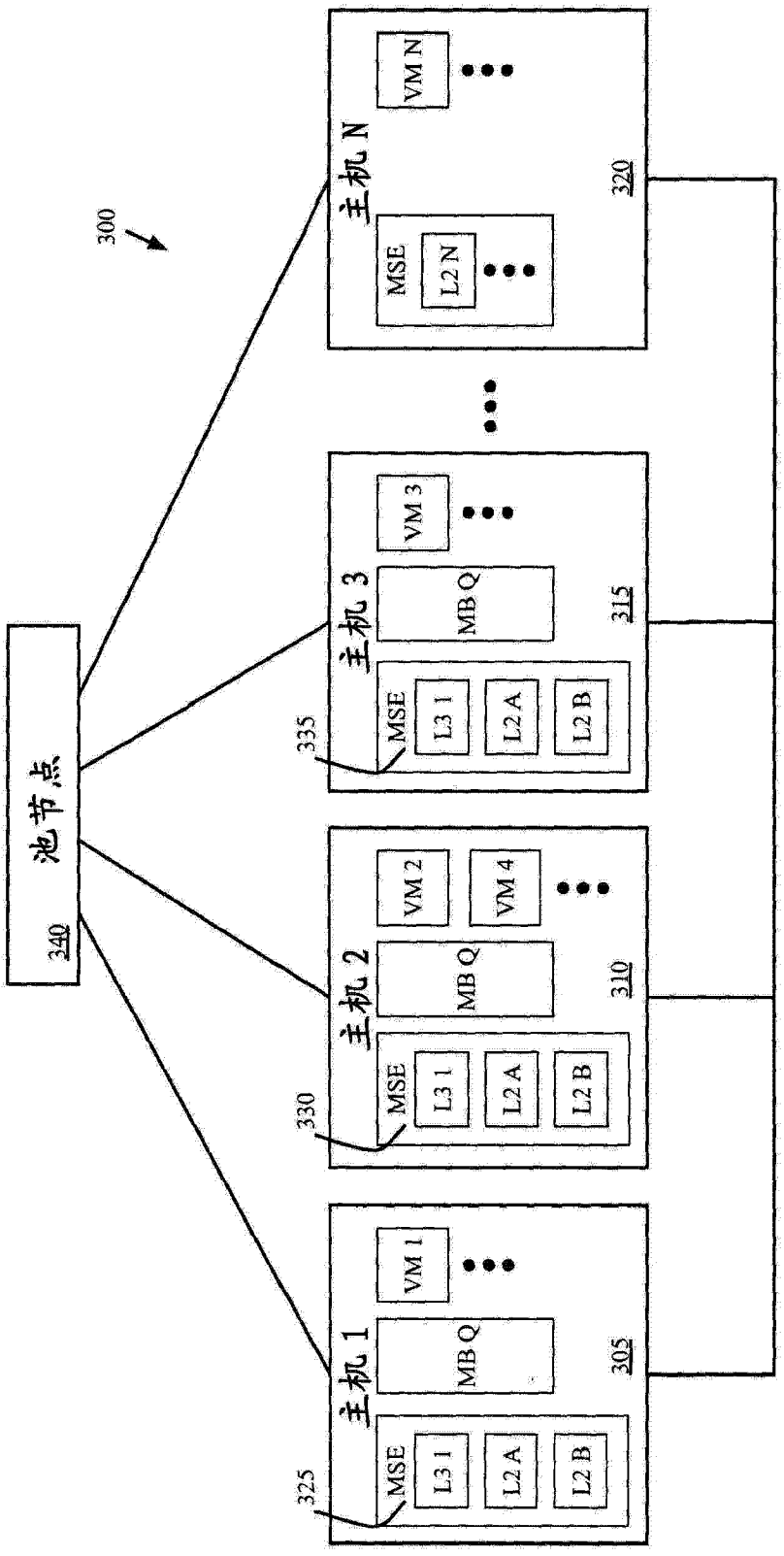

300
池节点
340
主机 1
VM 1
MB Q
MSE
L3 1
L2 A
L2 B
305
325
主机 2
VM 2
VM 4
MB Q
MSE
L3 1
L2 A
L2 B
310
330
主机 3
VM 3
MB Q
MSE
L3 1
L2 A
L2 B
315
335
主机 N
VM N
MSE
L2 N
320

图 3

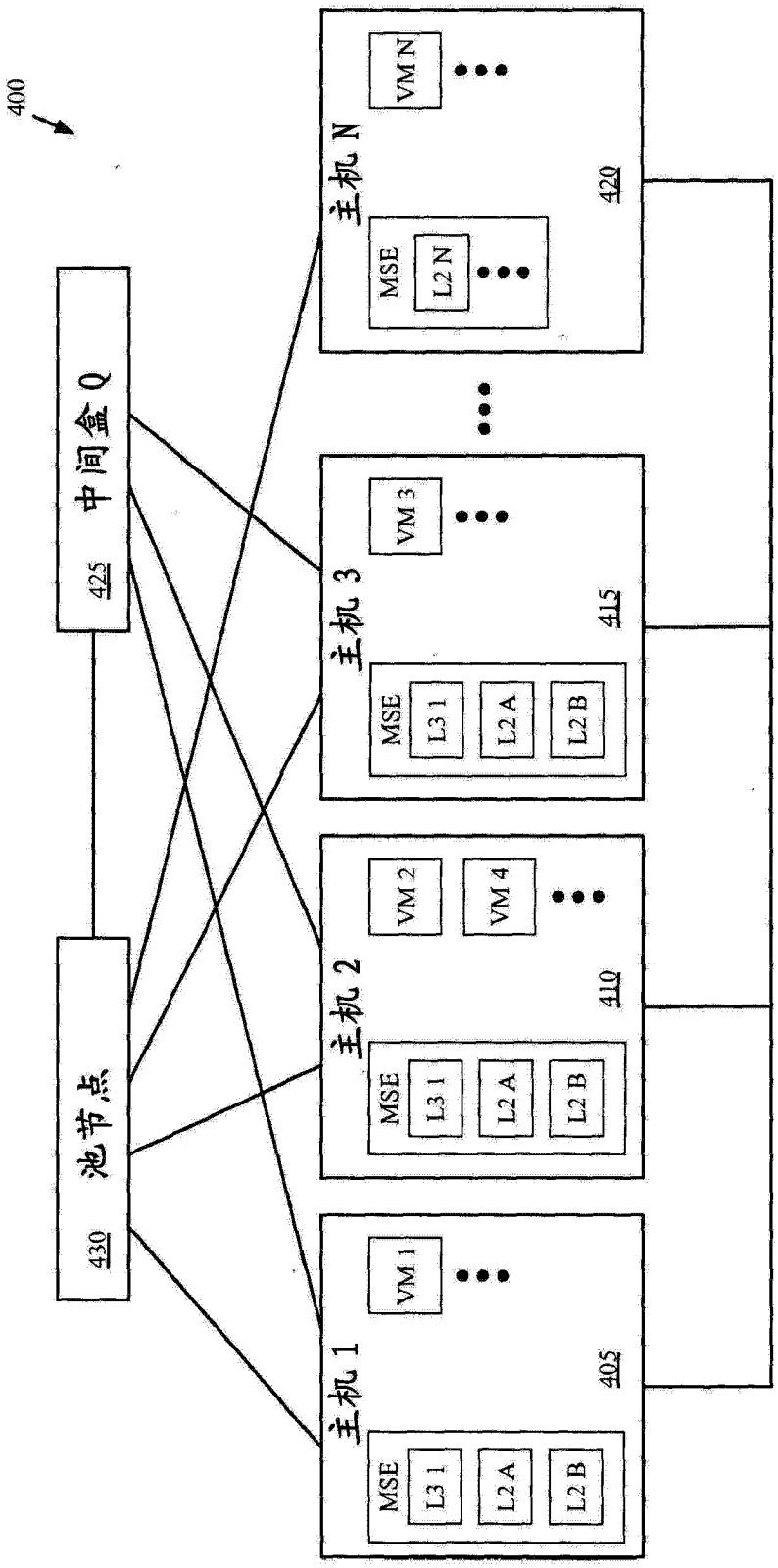

400
主机 N
420
VM N
MSE
L2 N
中间盒 Q
425
主机 3
415
VM 3
MSE
L3 1
L2 A
L2 B
池节点
430
主机 2
410
VM 2
VM 4
MSE
L3 1
L2 A
L2 B
主机 1
405
VM 1
MSE
L3 1
L2 A
L2 B

图 4

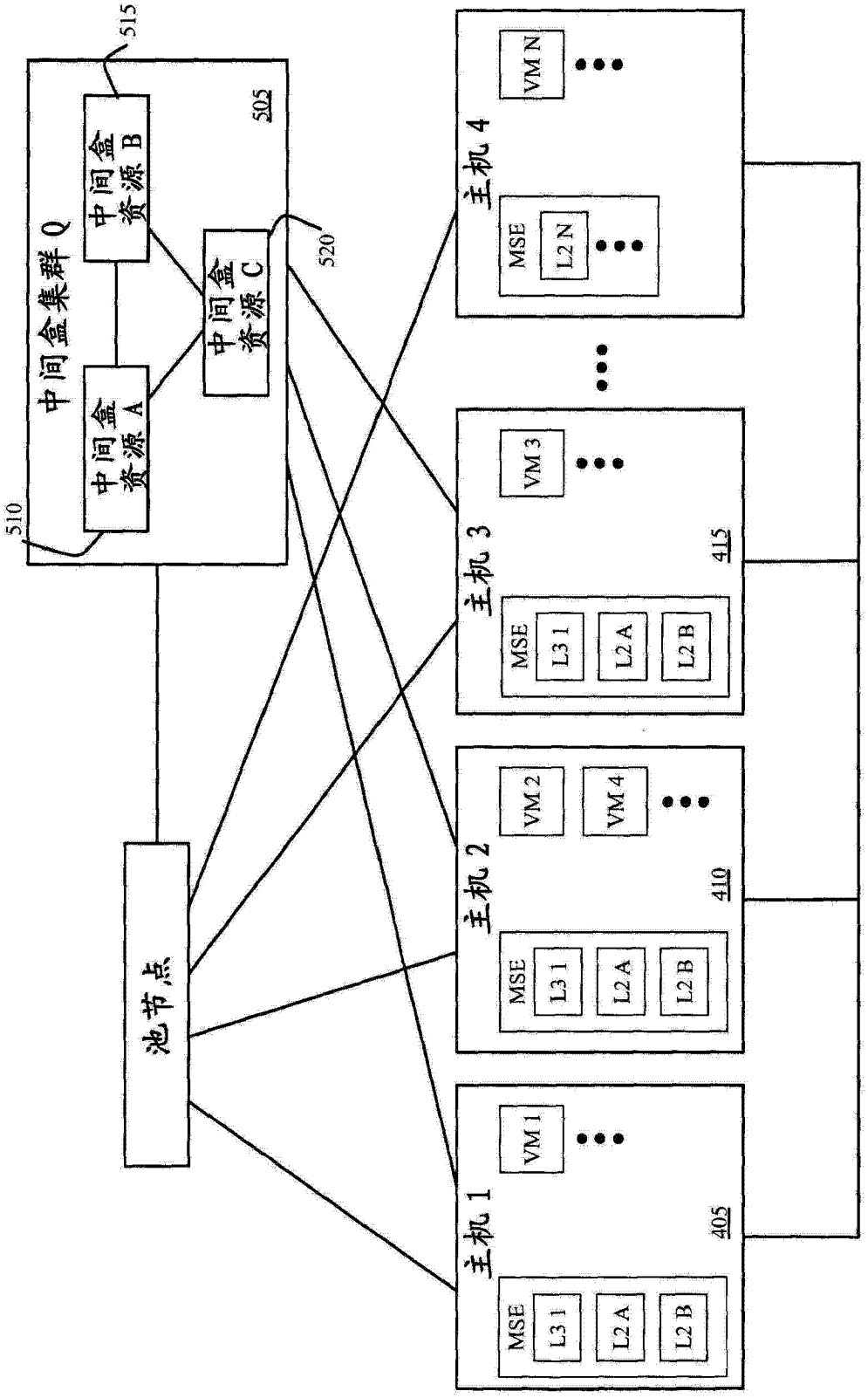

中间盒集群 Q
505
515
520
510
中间盒资源 B
中间盒资源 C
中间盒资源 A
池节点
主机 1
主机 2
主机 3
主机 4
405
410
415
MSE
L3 1
L2 A
L2 B
L2 N
VM 1
VM 2
VM 3
VM 4
VM N

图 5

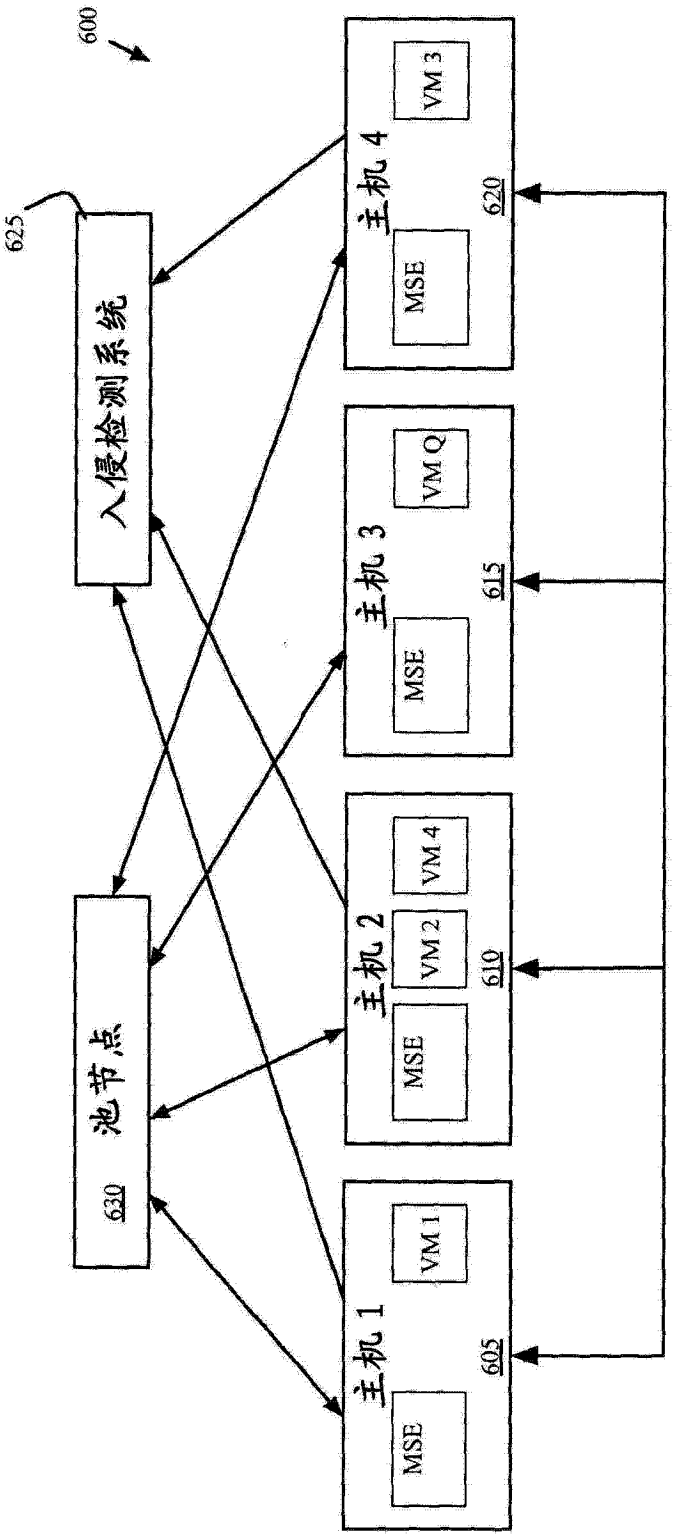

600
625
入侵检测系统
池节点
630
主机 1
MSE
VM 1
605
主机 2
MSE
VM 2
VM 4
610
主机 3
MSE
VM Q
615
主机 4
MSE
VM 3
620

图 6

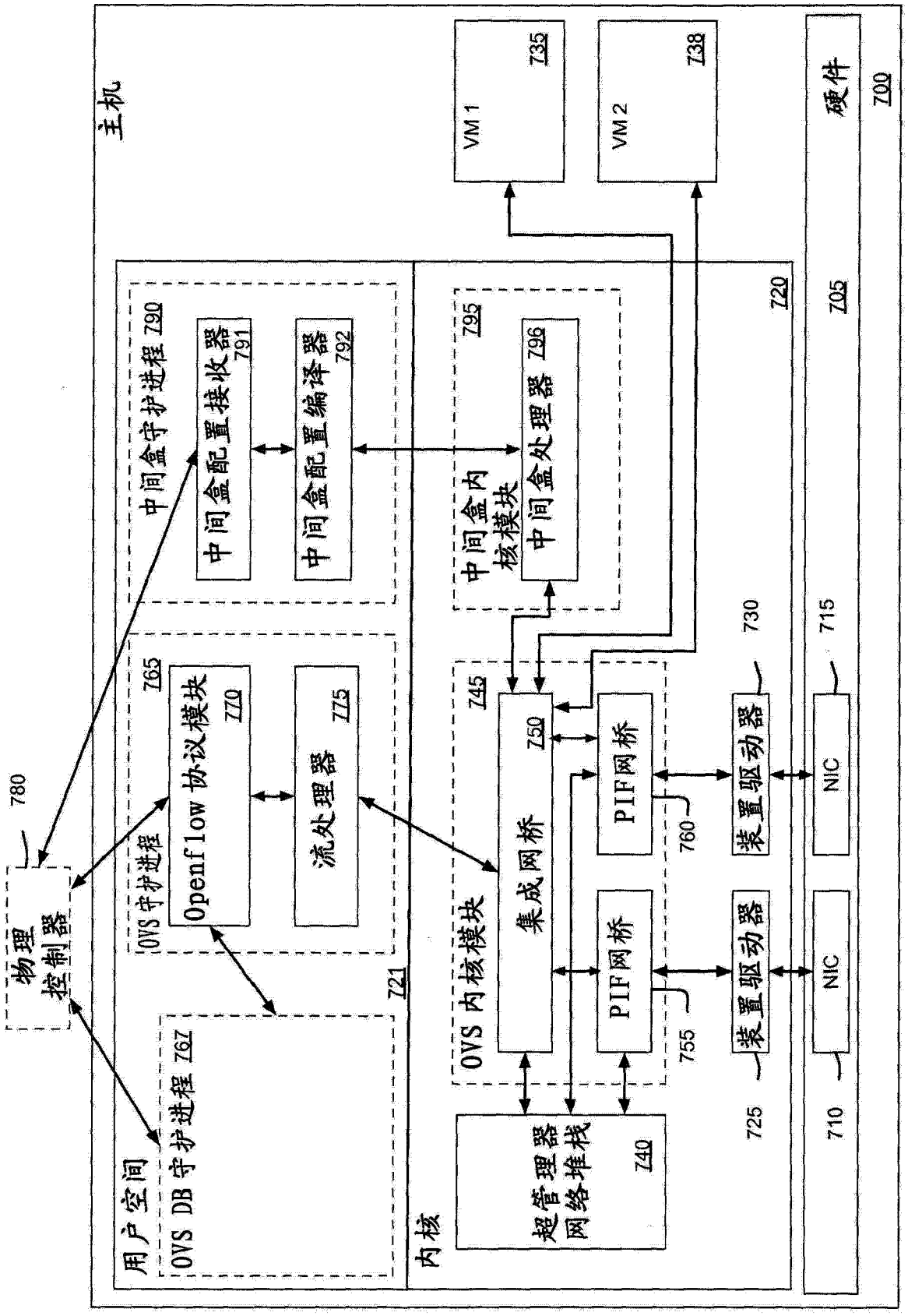

主机
VM 1
Z35
VM 2
Z38
硬件
Z00
Z05
Z20
中间盒守护进程 Z90
中间盒配置接收器 791
中间盒配置编译器 792
中间盒内核模块 Z95
中间盒处理器 Z96
780
物理控制器
OVS 守护进程 Z65
Openflow 协议模块 ZZ0
流处理器 ZZ5
Z21
用户空间
OVS DB 守护进程 Z67
OVS 内核模块 Z45
集成网桥 Z50
PIF网桥
PIF网桥
730
715
760
755
装置驱动器
装置驱动器
NIC
NIC
725
710
内核
超管理器网络堆栈 Z40

图 7

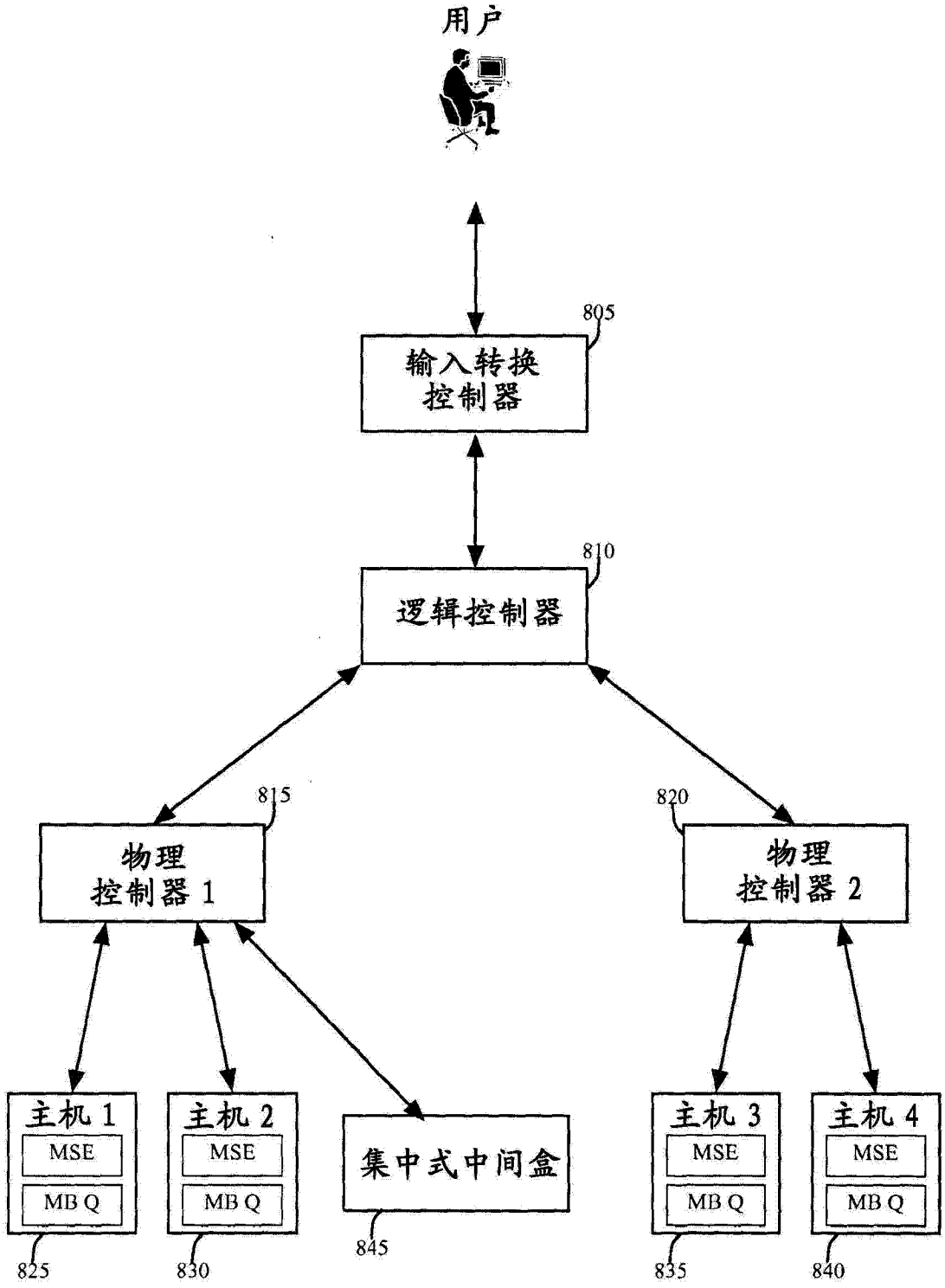

用户
805
输入转换
控制器
810
逻辑控制器
815
物理
控制器 1
820
物理
控制器 2
主机 1
MSE
MB Q
825
主机 2
MSE
MB Q
830
集中式中间盒
845
主机 3
MSE
MB Q
835
主机 4
MSE
MB Q
840

图 8

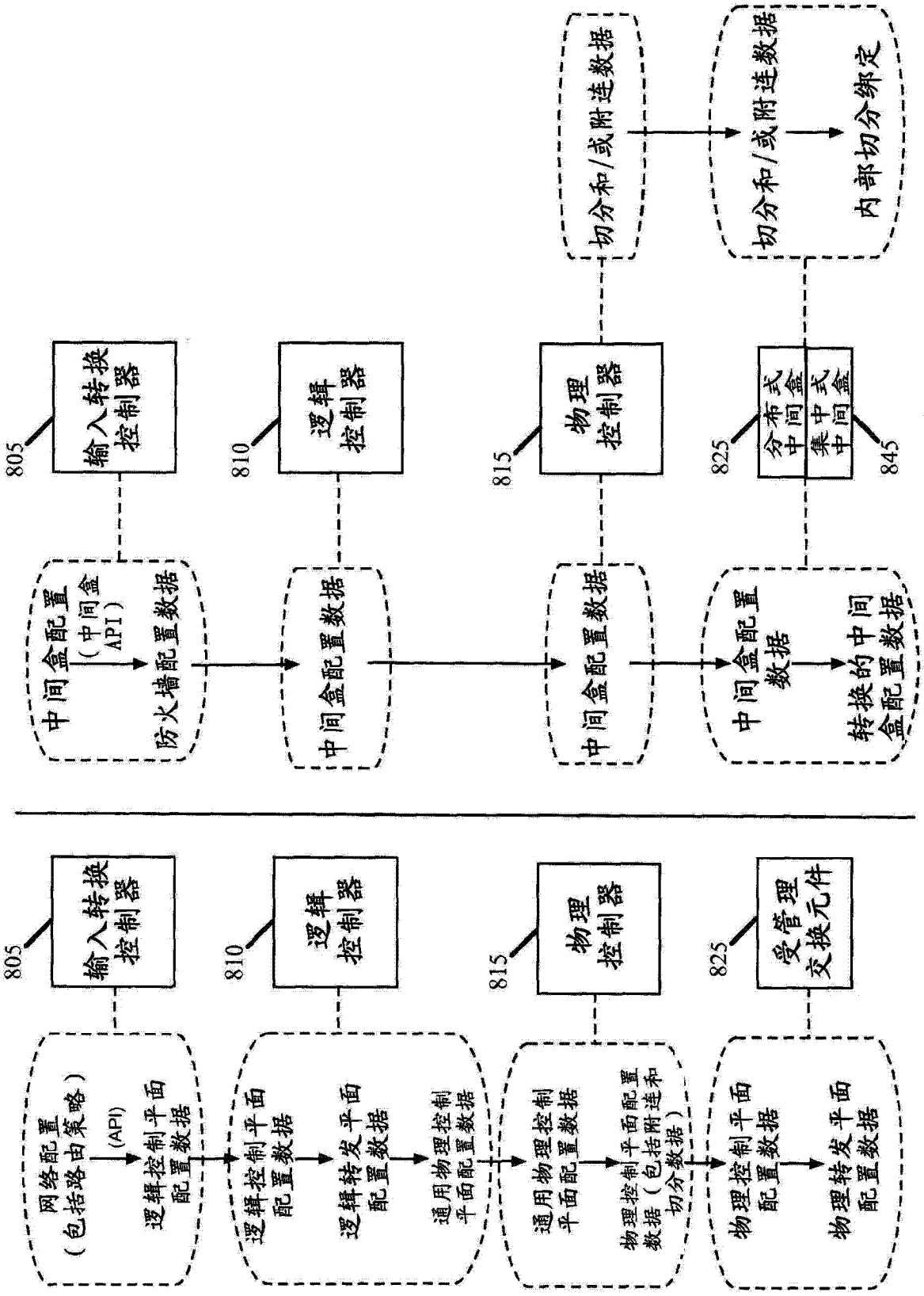

805
输入转换控制器
网络配置（包括路由策略）
(API)
逻辑控制平面配置数据
逻辑控制平面配置数据
逻辑转发平面配置数据
通用物理控制平面配置数据
810
逻辑控制器
815
物理控制器
通用物理控制平面配置数据
物理控制平面配置数据（包括附连和切分数据）
825
受管理交换元件
物理控制平面配置数据
物理转发平面配置数据
805
输入转换控制器
中间盒配置（中间盒API）
防火墙配置数据
810
逻辑控制器
中间盒配置数据
815
物理控制器
中间盒配置数据
切分和/或附连数据
825
分布式中间盒
集中中间盒
845
中间盒配置数据
转换的中间盒配置数据
切分和/或附连数据
内部切分绑定

图 9

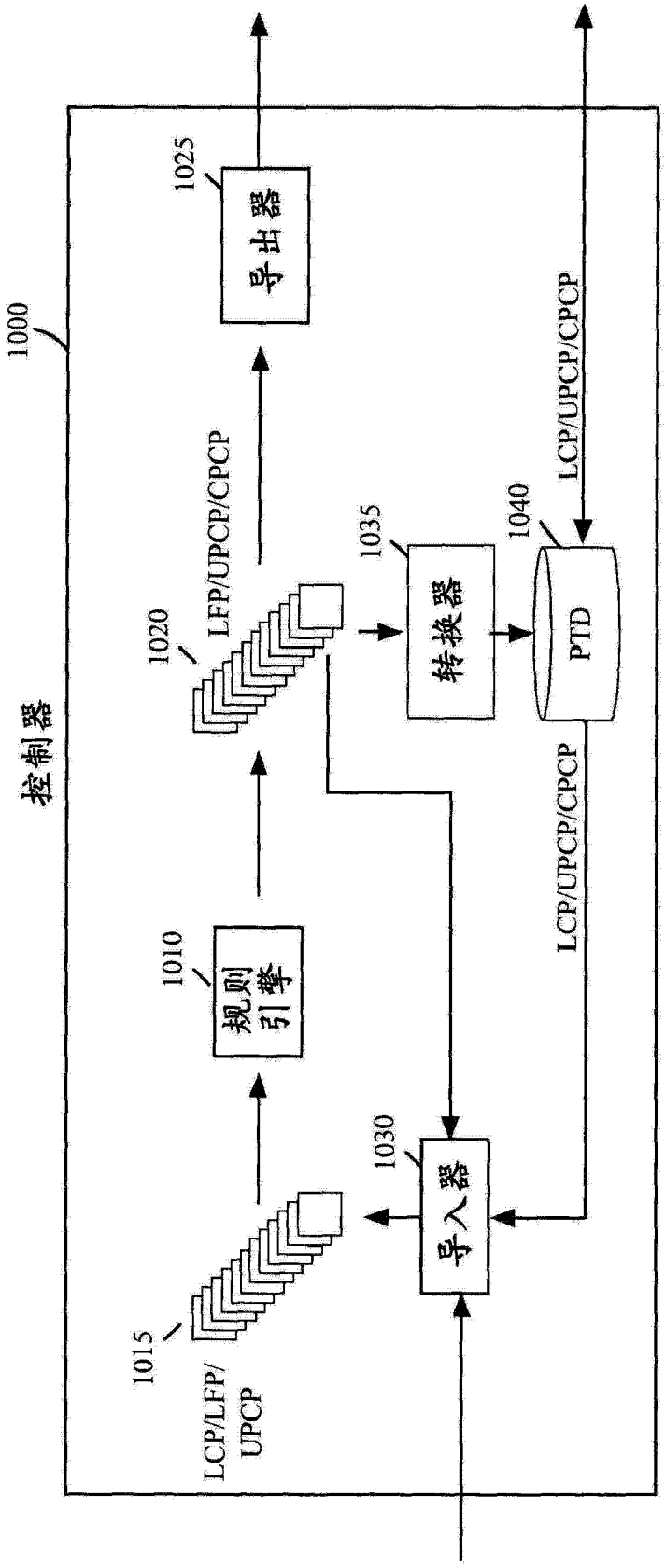

1000
控制器
1025
导出器
LFP/UPCP/CPCP
1020
1035
转换器
1040
PTD
LCP/UPCP/CPCP
LCP/UPCP/CPCP
1010
规则引擎
1030
导入器
1015
LCP/LFP/UPCP

图 10

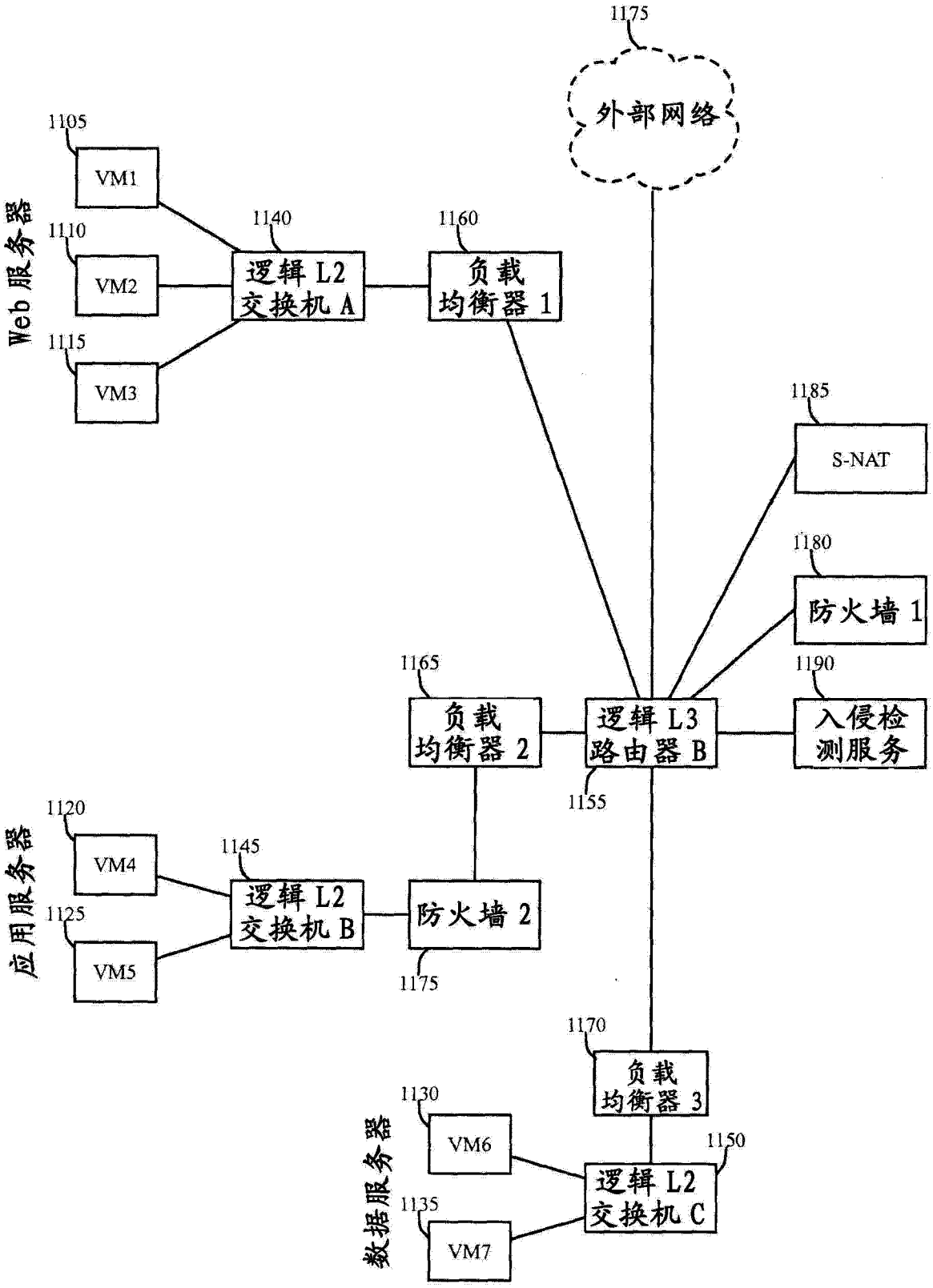

1175
外部网络
1105
VM1
Web 服务器
1110
VM2
1115
VM3
1140
逻辑 L2
交换机 A
1160
负载
均衡器 1
1185
S-NAT
1180
防火墙 1
1165
负载
均衡器 2
逻辑 L3
路由器 B
1155
1190
入侵检
测服务
1120
VM4
应用服务器
1125
VM5
1145
逻辑 L2
交换机 B
防火墙 2
1175
1170
负载
均衡器 3
1130
VM6
数据服务器
1135
VM7
1150
逻辑 L2
交换机 C

图 11

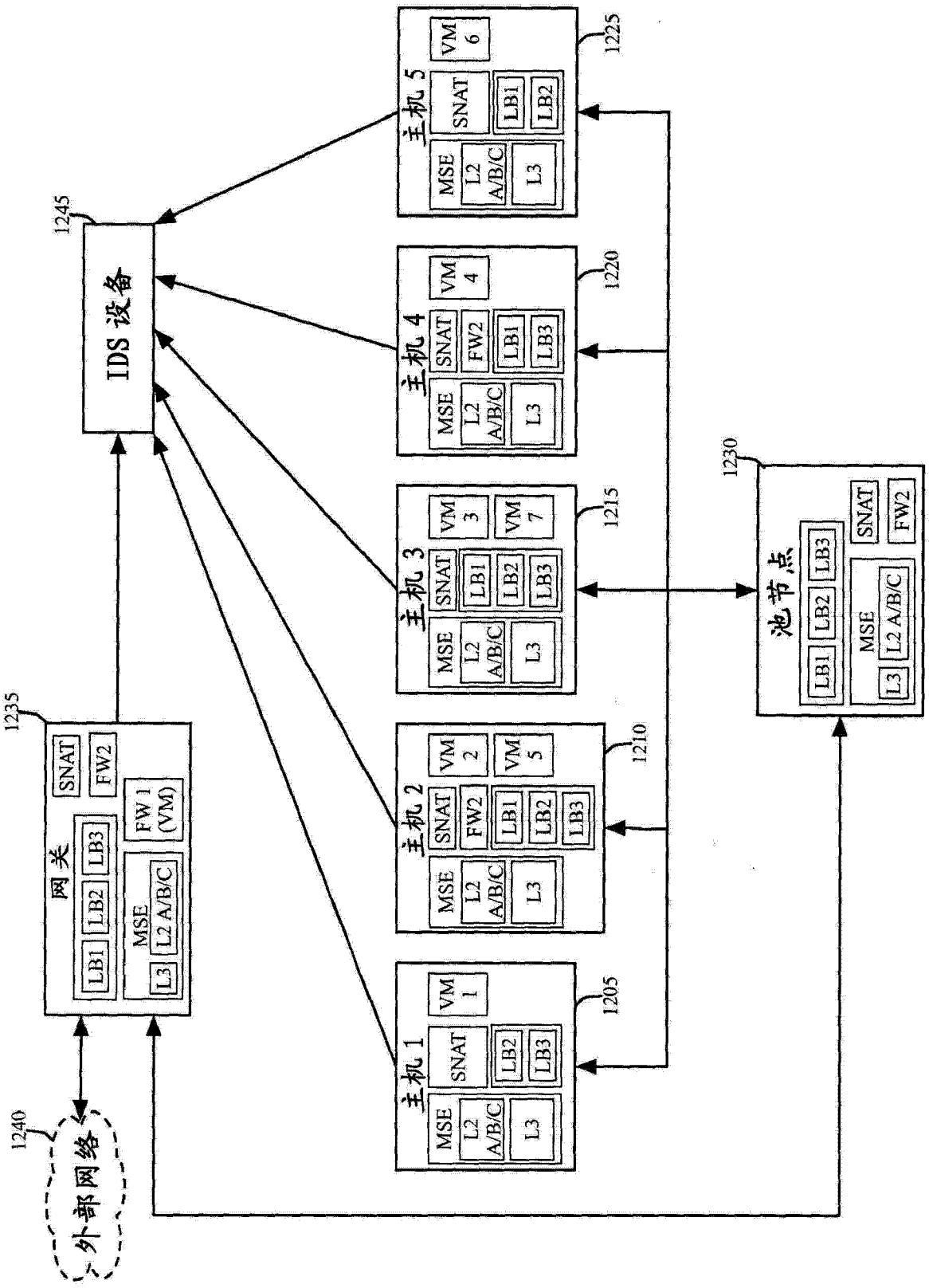

1245
IDS 设备
1235
网关
SNAT
FW2
LB1
LB2
LB3
MSE
L2 A/B/C
L3
FW 1 (VM)
1240
外部网络
主机 1
VM 1
SNAT
LB2
LB3
MSE
L2 A/B/C
L3
1205
主机 2
VM 2
VM 5
SNAT
FW2
LB1
LB2
LB3
1210
主机 3
VM 3
VM 7
SNAT
LB1
LB2
LB3
1215
主机 4
VM 4
SNAT
FW2
LB1
LB3
1220
主机 5
VM 6
SNAT
LB1
LB2
1225
池节点
LB1
LB2
LB3
MSE
L3
L2 A/B/C
SNAT
FW2
1230

图 12

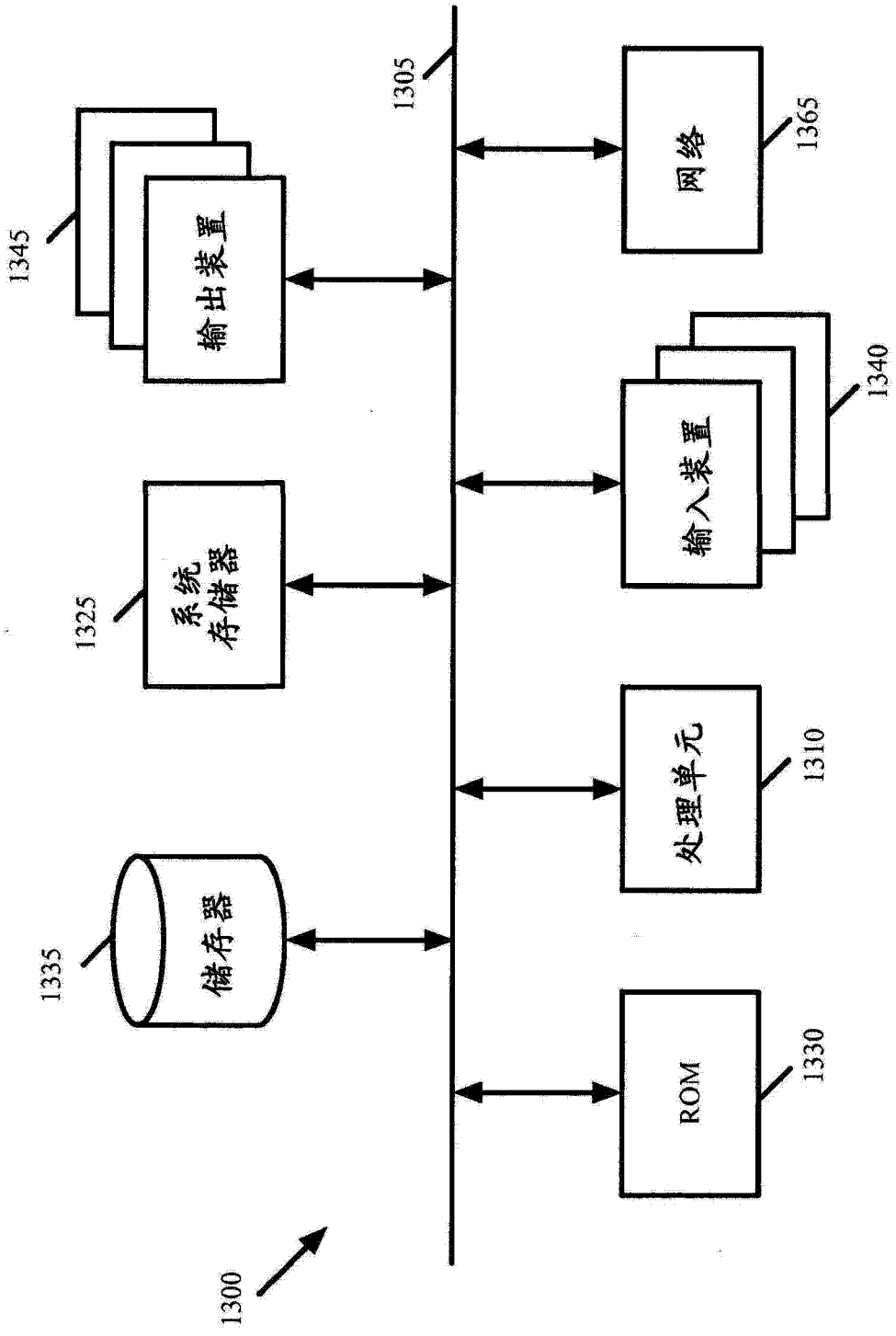

1305
网络
1365
1345
输出装置
输入装置
1340
系统存储器
1325
处理单元
1310
储存器
1335
ROM
1330
1300

图 13

1. 一种用于在包括多个节点的托管系统中配置逻辑网络的方法，所述方法包括：

接收对于所述逻辑网络的第一中间盒的第一配置和对于所述逻辑网络的第二中间盒的第二配置，所述逻辑网络包括托管在所述多个节点的子集上的多个虚拟机；

识别所述多个节点中作为用于实现所述第一中间盒的节点的子集；

分发用于在所识别的节点上实现的所述第一配置，其中，所识别的节点中的每一个接收对于所述第一中间盒的相同的第一配置；以及

分发用于在单个物理机器上实现的所述第二配置。

2. 根据权利要求 1 所述的方法，其中，所述单个物理机器包括托管用于实现所述第二中间盒的虚拟机的单个节点。

3. 根据权利要求 1 所述的方法，其中，所述单个物理机器包括与所述多个节点分离的物理中间盒设备。

4. 根据权利要求 1 所述的方法，其中，所述方法由第一网络控制器执行。

5. 根据权利要求 4 所述的方法，其中，分发所述第一配置包括：

自动地识别管理所识别的节点的多个附加网络控制器；以及

将所述第一配置分发给所识别的网络控制器以用于随后分发给所识别的节点。

6. 根据权利要求 5 所述的方法，其中，所识别的节点中的每一个由所述物理控制器中的单个物理控制器管理。

7. 根据权利要求 4 所述的方法，其中，分发所述第二配置包括：

自动地从一组附加网络控制器识别管理所述单个物理机器的特定的附加网络控制器；以及

将所述第二配置分发给所识别的特定的网络控制器以用于随后分发给所述单个物理机器。

8. 根据权利要求 1 所述的方法，其中，接收对于所述逻辑网络的第一中间盒的第一配置和对于所述逻辑网络的第二中间盒的第二配置包括：通过特定于所述第一中间盒的第一应用程序接口 API 接收配置，以及通过特定于所述第二中间盒的第二 API 接收第二配置。

9. 根据权利要求 1 所述的方法，其中，所述第一配置被作为一组数据库表记录接收。

10. 根据权利要求 9 所述的方法，其中，所述方法在分发记录之前不对所述记录进行转换。

11. 一种用于实现包括一组终端机、第一逻辑中间盒以及第二逻辑中间盒的逻辑网络的系统，所述终端机和逻辑中间盒由一组逻辑转发元件连接，所述系统包括：

一组节点，其中，若干节点中的每一个包括：

虚拟机，所述虚拟机用于实现所述逻辑网络的终端机；

受管理交换元件，所述受管理交换元件用于实现所述逻辑网络的一组逻辑转发元件；和

中间盒元件，所述中间盒元件用于实现所述逻辑网络的第一逻辑中间盒，其中，第一节点的第一中间盒元件和第二节点的第二中间盒元件实现对于所述第一逻辑中间盒的相同配置；以及

物理中间盒设备，所述物理中间盒设备用于实现所述第二逻辑中间盒。

12. 根据权利要求 11 所述的系统，其中，所述系统进一步用于实现包括由第二组逻辑

转发元件连接的第二组终端机和第三逻辑中间盒的第二逻辑网络。

13. 根据权利要求 12 所述的系统，其中，所述一组节点中的特定节点包括用于实现所述第二逻辑网络的第二终端机的第二虚拟机。

14. 根据权利要求 13 所述的系统，其中，所述特定节点的受管理交换元件进一步用于实现所述第二逻辑网络的第二组逻辑转发元件，并且所述中间盒元件进一步用于实现所述逻辑网络的第三逻辑中间盒。

15. 根据权利要求 11 所述的系统，其中，所述第一逻辑中间盒和第二逻辑中间盒是不同类型的中间盒。

16. 根据权利要求 11 所述的系统，其中，所述第一逻辑中间盒和第二逻辑中间盒是在所述逻辑网络中执行不同功能的相同类型的中间盒。

17. 一种用于实现逻辑网络的系统，所述系统包括 ：

多个主机，所述多个主机用于实现所述逻辑网络的第一逻辑中间盒，其中，所述第一逻辑中间盒在不在主机的中间盒元件之间传送状态信息的情况下独立地对所述多个主机中的每一个的中间盒元件进行操作 ；以及

一组独立的物理中间盒，所述一组独立的物理中间盒用于实现所述逻辑网络的第二逻辑中间盒，其中，所述第二逻辑中间盒执行需要与所述逻辑网络的若干不同组的终端机之间的分组相关的状态信息的操作。

18. 根据权利要求 17 所述的系统，其中，每个特定的中间盒元件存储关于其分组被所述特定的中间盒元件处理的传输连接的状态信息。

19. 根据权利要求 18 所述的系统，其中，对于特定传输连接的处理不需要关于其它传输连接的任何状态信息。

20. 根据权利要求 17 所述的系统，其中，所述一组独立的物理中间盒包括单个物理中间盒。

21. 根据权利要求 17 所述的系统，其中，所述一组独立的物理中间盒包括作为资源池操作的中间盒集群。

22. 根据权利要求 21 所述的系统，还包括用于在集群中的物理中间盒之间共享状态信息的所述中间盒集群之间的专用网络连接。

23. 根据权利要求 17 所述的系统，其中，所述第一逻辑中间盒包括防火墙、负载均衡器和源网络地址转换器中的一个。

24. 根据权利要求 17 所述的系统，其中，所述第二逻辑中间盒包括执行需要针对每个被处理的分组的状态信息的操作的入侵检测系统。

25. 根据权利要求 17 所述的系统，其中，所述第二逻辑中间盒包括广域网优化器。