EUROPEAN PATENT SPECIFICATION

METHOD AND APPARATUS FOR IMPLEMENTING AND MANAGING VIRTUAL SWITCHES

VERFAHREN UND APPARAT ZUR IMPLEMENTIERUNG UND MANAGEN VON VIRTUELLEN SWITCHEN

PROCEDE ET APPARAIL POUR L'IMPLEMENTATION ET GESTION DES COMMUTATEURS VIRTUELS

Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO SE SI SK SM TR

Priority: 01.04.2009 US 165875 P

Application number: 10716930.2
Date of filing: 01.04.2010

Date of publication and mention of the grant of the patent:
07.05.2014 Bulletin 2014/19

Int Cl.:
H04L 12/937 (2013.01)

International application number:
PCT/US2010/029717

International publication number:
WO 2010/115060 (07.10.2010 Gazette 2010/40)

METHOD AND APPARATUS FOR IMPLEMENTING AND MANAGING VIRTUAL SWITCHES

PROCEDE ET APPARAIL POUR L'IMPLEMENTATION ET GESTION DES COMMUTATEURS VIRTUELS

Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO SE SI SK SM TR

Priority: 01.04.2009 US 165875 P

Application number: 10716930.2
Date of filing: 01.04.2010

Date of publication and mention of the grant of the patent:
07.05.2014 Bulletin 2014/19

Int Cl.:
H04L 12/937 (2013.01)

International application number:
PCT/US2010/029717

International publication number:
WO 2010/115060 (07.10.2010 Gazette 2010/40)

METHOD AND APPARATUS FOR IMPLEMENTING AND MANAGING VIRTUAL SWITCHES

VERFAHREN UND APPARAT ZUR IMPLEMENTIERUNG UND MANAGEN VON VIRTUELLEN SWITCHEN

PROCEDE ET APPARAIL POUR L'IMPLEMENTATION ET GESTION DES COMMUTATEURS VIRTUELS

Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO SE SI SK SM TR

Priority: 01.04.2009 US 165875 P

Application number: 10716930.2
Date of filing: 01.04.2010

Date of publication and mention of the grant of the patent:
07.05.2014 Bulletin 2014/19

Int Cl.:
H04L 12/937 (2013.01)

International application number:
PCT/US2010/029717

International publication number:
WO 2010/115060 (07.10.2010 Gazette 2010/40)

METHOD AND APPARATUS FOR IMPLEMENTING AND MANAGING VIRTUAL SWITCHES

VERFAHREN UND APPARAT ZUR IMPLEMENTIERUNG UND MANAGEN VON VIRTUELLEN SWITCHEN

PROCEDE ET APPARAIL POUR L'IMPLEMENTATION ET GESTION DES COMMUTATEURS VIRTUELS

Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO SE SI SK SM TR

Priority: 01.04.2009 US 165875 P

Application number: 10716930.2
Date of filing: 01.04.2010

Date of publication and mention of the grant of the patent:
07.05.2014 Bulletin 2014/19

Int Cl.:
H04L 12/937 (2013.01)

International application number:
PCT/US2010/029717

International publication number:
WO 2010/115060 (07.10.2010 Gazette 2010/40)

METHOD AND APPARATUS FOR IMPLEMENTING AND MANAGING VIRTUAL SWITCHES

VERFAHREN UND APPARAT ZUR IMPLEMENTIERUNG UND MANAGEN VON VIRTUELLEN SWITCHEN

PROCEDE ET APPARAIL POUR L'IMPLEMENTATION ET GESTION DES COMMUTATEURS VIRTUELS

Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO SE SI SK SM TR

Priority: 01.04.2009 US 165875 P

Application number: 10716930.2
Date of filing: 01.04.2010

Date of publication and mention of the grant of the patent:
07.05.2014 Bulletin 2014/19

Int Cl.:
H04L 12/937 (2013.01)

International application number:
PCT/US2010/029717

International publication number:
WO 2010/115060 (07.10.2010 Gazette 2010/40)

METHOD AND APPARATUS FOR IMPLEMENTING AND MANAGING VIRTUAL SWITCHES

VERFAHREN UND APPARAT ZUR IMPLEMENTIERUNG UND MANAGEN VON VIRTUELLEN SWITCHEN

PROCEDE ET APPARAIL POUR L'IMPLEMENTATION ET GESTION DES COMMUTATEURS VIRTUELS

Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO SE SI SK SM TR

Priority: 01.04.2009 US 165875 P

Application number: 10716930.2
Date of filing: 01.04.2010

Date of publication and mention of the grant of the patent:
07.05.2014 Bulletin 2014/19

Int Cl.:
H04L 12/937 (2013.01)

International application number:
PCT/US2010/029717

International publication number:
WO 2010/115060 (07.10.2010 Gazette 2010/40)

METHOD AND APPARATUS FOR IMPLEMENTING AND MANAGING VIRTUAL SWITCHES

VERFAHREN UND APPARAT ZUR IMPLEMENTIERUNG UND MANAGEN VON VIRTUELLEN SWITCHEN

PROCEDE ET APPARAIL POUR L'IMPLEMENTATION ET GESTION DES COMMUTATEURS VIRTUELS

Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO SE SI SK SM TR

Priority: 01.04.2009 US 165875 P

Application number: 10716930.2
Date of filing: 01.04.2010

Date of publication and mention of the grant of the patent:
07.05.2014 Bulletin 2014/19

Int Cl.:
H04L 12/937 (2013.01)

International application number:
PCT/US2010/029717

International publication number:
WO 2010/115060 (07.10.2010 Gazette 2010/40)
The present invention relates to networking, and more particularly to the design and use of virtual switches in virtual networking.

BACKGROUND OF THE INVENTION

The increased sophistication of computing, including mobility, virtualization, dynamic workloads, multi-tenancy, and security needs, require a better paradigm for networking. Virtualization is an important catalyst of the new requirements for networks. With it, multiple VMs can share the same physical server, those VMs can be migrated, and workloads are being built to "scale-out" dynamically as capacity is needed. In order to cope with this new level of dynamics, the concept of a distributed virtual switch has arisen. The idea behind a distributed virtual switch is to provide a logical view of a switch which is decoupled from the underlying hardware and can extend across multiple switches or hypervisors.

One example of a conventional distributed virtual switch is the Nexus 1000V provided by Cisco of San Jose, California see for example US2009/0083445. Another example is the DVS provided by VMWare of Palo Alto. While both of these are intended for virtual-only environments, there is no architectural reason why the same concepts cannot be extended to physical environments.

Three of the many challenges of large networks (including datacenters and the enterprise) are scalability, mobility, and multi-tenancy and often the approaches taken to address one hamper the other. For instance, one can easily provide network mobility for VMs within an L2 domain, but L2 domains cannot scale to large sizes. And retaining tenant isolation greatly complicates mobility.

Conventional distributed virtual switches fall short of addressing these problems in a number of areas. First, they don’t provide multi-tenancy, they don’t bridge IP subnets, and cannot scale to support tens of thousands of end hosts.

Accordingly, the concepts have not effectively moved beyond virtual environments to include physical hosts in a general and flexible manner.

Accordingly, a need remains in the art for a distributed virtual networking platform that addresses these and other issues.

SUMMARY OF THE INVENTION

In general, the present invention relates to a virtual platform in which one or more distributed virtual switches can be created for use in virtual networking. According to some aspects, the distributed virtual switch according to the invention provides the ability for virtual and physical machines to more readily, securely, and efficiently communicate with each other even if they are not located on the same physical host and/or in the same subnet or VLAN. According other aspects, the distributed virtual switches of the invention can support integration with traditional IP networks and support sophisticated IP technologies including NAT functionality, stateful firewalling, and notifying the IP network of workload migration. According to further aspects, the virtual platform of the invention manages and/or uses VLAN or tunnels (e.g, GRE) to create a distributed virtual switch for a network while working with existing switches and routers in the network. The present invention finds utility in both enterprise networks, datacenters and other facilities.

In accordance with these and other aspects, a method of managing networking resources in a site comprising a plurality of hosts and physical forwarding elements according to embodiments of the invention includes identifying a first set of virtual machines using a first set of the plurality of hosts and physical forwarding elements, identifying a second set of virtual machines using a second set of the plurality of hosts and physical forwarding elements, certain of the hosts and physical forwarding elements in the first and second sets being the same, and providing first and second distributed virtual switches that exclusively handle communications between the first and second sets of virtual machines, respectively, while maintaining isolation between the first and second sets of virtual machines.

In additional furtherance of these and other aspects, a method of managing communications in a network comprising one or more physical forwarding elements according to embodiments of the invention includes providing a network virtualization layer comprising a logical forwarding element, providing a mapping between a port of the logical forwarding element to a port of certain of the physical forwarding elements, and causing the physical forwarding element to forward a packet using the provided mapping.

FIG. 1 is a block diagram illustrating aspects of providing a virtual platform according to embodiments of the invention;

FIG. 2 illustrates a packet forwarding scheme implemented in a network using principles of the invention;

FIG. 3 illustrates an example of providing a distributed virtual switch in accordance with the invention
in a data center having several virtual machines and physical hosts; and
FIG. 4 is a functional block diagram of an example distributed virtual switch according to embodiments of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0010] The present invention will now be described in detail with reference to the drawings, which are provided as illustrative examples of the invention so as to enable those skilled in the art to practice the invention. Notably, the figures and examples below are not meant to limit the scope of the present invention to a single embodiment, but other embodiments are possible by way of interchange of some or all of the described or illustrated elements. Moreover, where certain elements of the present invention can be partially or fully implemented using known components, only those portions of such known components that are necessary for an understanding of the present invention will be described, and detailed descriptions of other portions of such known components will be omitted so as not to obscure the invention. Embodiments described as being implemented in software should not be limited thereto, but can include embodiments implemented in hardware, or combinations of software and hardware, and vice-versa, as will be apparent to those skilled in the art, unless otherwise specified herein. In the present specification, an embodiment showing a singular component should not be considered limiting; rather, the invention is intended to encompass other embodiments including a plurality of the same component, and vice-versa, unless explicitly stated otherwise herein. Moreover, applicants do not intend for any term in the specification or claims to be ascribed an uncommon or special meaning unless explicitly set forth as such. Further, the present invention encompasses present and future known equivalents to the known components referred to herein by way of illustration.

[0011] According to general aspects, the invention relates to a virtual platform for use with a network that provides the ability for physical and virtual machines associated with it to more readily, securely, and efficiently communicate with each other even if they are not located on the same physical host and/or in the same VLAN or subnet. According to further aspects, it also allows multiple different tenants sharing the same physical network infrastructure to communicate and set configuration state in isolation from each other.

[0012] An example implementation of aspects of the invention is illustrated in FIG. 1. As shown in FIG. 1, a site such as a data center or an enterprise network can include a physical network 104. The physical network 104 includes a plurality of VMs and/or non-virtualized physical servers, as well as physical and virtual switches. VMs are hosted by a virtualization platform such as that provided by VMware, (e.g. included in vSphere, vCenter etc.) and physical servers may be any generic computational unit such as those provided by HP, Dell and others. It should be apparent that large hosting services or enterprise networks can maintain multiple data centers, or networks at several sites, which may be geographically dispersed (e.g. San Francisco, New York, etc.).

[0013] FIG. 1 further depicts how the invention introduces a network virtualization layer 106 on top of which one or more distributed virtual switches 108 are maintained by a network hypervisor 102. These distributed virtual switches 108 may extend across subnets, may include physical hosts or physical network ports, and can share the same physical hardware. According to aspects of the invention, these distributed virtual switches can provide isolated contexts for multi-tenant environments, can support VM migration across subnets, can scale to tens or hundreds of thousands of physical servers, and can support seamless integration with physical environments.

[0014] As a particular example, the invention could be deployed by service providers (such as San Antonio based Rackspace) which often support both virtual and physical hosting of servers for a plurality of customers. In such an example, a single customer may have both VMs and physical servers hosted at the same service provider. Further, a service provider may have multiple datacenters in geographically distinct locations. The invention could be deployed within the service provider operations such that each customer/tenant can be allocated one or more distributed virtual switches (DVS’s) 108. These DVS’s can be independently configured and given minimum resource guarantees as specified by the service provider operators using hypervisor 102. A single DVS may contain both physical and virtual hosts and may bridge multiple subnets or VLANs. For example, a single DVS 108 may connect to virtual machines at the service provider, physical machines as part of a managed hosting service, and may even extend across the Internet to connect to the customer premises.

[0015] According to further aspects, the invention introduces a new abstraction between the physical forwarding elements and control plane. The abstraction exposes the forwarding elements as one or more logical forwarding elements for the control plane. The logical forwarding elements possess similar properties and functionalities as their physical counterparts, i.e., lookup tables, ports, counters, as well as associated capacities (e.g., port speeds and/or bisectional bandwidth).

[0016] Although shown separately for ease of illustrating aspects of the invention, the network hypervisor 102 and network virtualization layer 106 are preferably implemented by a common set of software (described in more detail below) that creates and maintains the logical forwarding elements and maps them to the underlying hardware. Nominally, this means exposing forwarding state, counters, and forwarding element events in their corresponding logical context. The control plane, rather than driving the physical forwarding elements directly,
then interfaces with the logical forwarding elements. [0017] More particularly, network virtualization layer 106 presents a forwarding abstraction to the control plane which is minimally affected by changes in the physical topology of network 104. From the point of view of the control plane, the addition of switches to the physical topology provides more forwarding bandwidth, but should not require any changes to the control logic, or the existing state in the logical forwarding tables. [0018] Layer 106 allows logical forwarding element ports to be bound to physical ports, or to provide other port abstractions such as virtual machine interfaces, VLANs, or tunnels. It is the job of the network hypervisor 102 (described below) to maintain the mappings between the ports on the logical forwarding elements in layer 106 and the underlying network 104, and to update flow tables in physical and/or virtual switches in the physical network accordingly. [0019] Each logical forwarding element in layer 106 provides an interface compatible with a traditional switch datapath. This is desirable for two reasons. First, the invention is preferably compatible with existing hardware and to be useful, all forwarding should remain on the hardware fast path. Thus, the logical forwarding plane should preferably map to existing forwarding pipelines. Second, existing network control stacks are preferably compatible with the invention. Accordingly, the interface of a logical element in layer 106 includes:

- Lookup tables: The logical forwarding element exposes one or more forwarding tables. Typically this includes an L2, L3, and ACL table. One example implementation is designed around OpenFlow (see www.openflow.org), according to which a more generalized table structure is built around a pipeline of TCAMs with forwarding actions specified for each rule. This structure provides quite a bit of flexibility allowing for support of forwarding rules, ACLs, SPAN, and other primitives.
- Ports: The logical forwarding element contains ports which represent bindings to the underlying network. Ports may appear and leave dynamically as they are either administratively added, or the component they are bound to fails or leaves. In embodiments of the invention, ports maintain much of the same qualities of their physical analogs including rx/tx counters, MTU, speed, error counters, and carrier signal.
- Physical network 104 consists of the physical forwarding elements. In embodiments of the invention, the forwarding elements can be traditional hardware switches with standard forwarding silicon, as well as virtual switches such as those included with hypervisors. In embodiments of the invention, certain or all of the existing switches provide support for a protocol to allow their flow tables to be adjusted to implement the distributed virtual switches of the present invention. Such a protocol can include OpenFlow, but other proprietary and open protocols such as OSPF may be used. In other embodiments of the invention, and according to certain beneficial aspects to be described in more detail below, some or all of the existing physical switches (and perhaps some of the virtual switches) need not support such a protocol and/or have their flow tables adjusted. In such embodiments, tunneling may be used to route traffic through such existing switches. [0023] At a high level, forwarding elements in the physical network 104 that are used by network hypervisor 102 to implement distributed virtual switches 108 have four primary responsibilities: i) to map incoming packets to the correct logical context, ii) to make logical forwarding decisions, iii) map logical forwarding decisions back to the physical next-hop address, and iv) to make physical forwarding decisions in order to send packets to the physical next hop.
- Physical network 104 consists of the physical forwarding elements. In embodiments, tunneling may be used to route traffic through such existing switches. [0024] More particularly, as shown in FIG. 2, all packets are handled by exactly one logical forwarding element in layer 106. However, multiple logical forwarding elements may be multiplexed over the same physical switch in physical network 104. So, on ingress, a packet must therefore be mapped to the correct logical context (S202). It may be the case that the current switch does not contain the logical forwarding state for a given packet, in which case it simply performs a physical forwarding decision (i.e., skip to step S208). Also, if all the physical switches are for implementing only a single logical forwarding element, the mapping becomes a no-op because the logical addressing may be used at the physical network.
- Physical network 104 consists of the physical forwarding elements. In embodiments of the invention, all packets are handled by exactly one logical forwarding element in layer 106. However, multiple logical forwarding elements may be multiplexed over the same physical switch in physical network 104. So, on ingress, a packet must therefore be mapped to the correct logical context. It may be the case that the current switch does not contain the logical forwarding state for a given packet, in which case it simply performs a physical forwarding decision (i.e., skip to step S208). Also, if all the physical switches are for implementing only a single logical forwarding element, the mapping becomes a no-op because the logical addressing may be used at the physical network.
- There are many different field(s) that can be used to map a packet to a logical context by the invention. For example, the field can be an identifying tag such as an MPLS header, or the ingress port. However, in order to provide transparency to end systems, the tag used for identifying logical contexts are preferably not exposed to the systems connecting to the logical switch. In general, this means that the first physical switch receiving a packet tags it to mark the context, and the last switch removes the tag. How the first tag is chosen depends largely on the deployment environment, as will be appreciated by those skilled in the art.
- In step S204, once a packet is mapped to its logical context, the physical switch performs a forwarding decision which is only meaningful within the logical context. This could be, for example, an L2 lookup for the logical switch or a sequence of lookups required for a logical L3 router. However, if the physical switch executing the logical decision does not have enough capacity to maintain all the logical state, the logical decision executed may be only a step in overall logical decision that needs be executed; and therefore, packet may require further logical processing before leaving the logical forwarding plane.
- In step S206, the logical decision is mapped to physical. The result of a logical forwarding decisions (assuming the packet wasn’t dropped) is one or more egress ports on the logical forwarding element in layer 106. Once these are determined, the network must send the packets to the physical objects in network 104 to which these
of a virtual machine on a different physical server. This could be, for example, a physical port on another physical switch, or a virtual port egress ports are bound. This could be, for example, a hypervisor 102, for example using a protocol such as OpenFlow. Other example methods for modifying flow tables include using an SDK such as that provided by networking chipset providers Marvell or Broadcom, or using a switch vendor API such as the OpenJunos API offered by Juniper. It should be noted that in some embodiments, and according to aspects of the invention, existing switches and routers can be used without having their flow tables adjusted by using tunneling.

The capacity of a logical forwarding element may exceed the capacity of an individual physical forwarding element. Therefore, the physical switch/forwarding element should preferably provide a traffic splitting action (e.g., ECMP or hashing) and link aggregation to distribute traffic over multiple physical paths/links. Finally, to effectively monitor links and tunnels the physical switches should provide a hardware based link and tunnel monitoring protocol implementation (such as BFD). Those skilled in the art will recognize how to implement physical switches and other elements in physical network 104 based on these examples, as well as from the overall descriptions herein.

In embodiments, the network hypervisor 102 implementation is decoupled from the physical forwarding elements, so that the hypervisor implementation has a global view over the network state. Therefore, the network hypervisor 102 needs to be involved whenever the state is changed on either side of it, by adjusting mappings and/or flow tables for all affected switches in network 104 accordingly. In other words, when there’s a network topology event on the physical network or when the control implementation changes the state of the logical forwarding plane, the network hypervisor 102 needs to be involved. In addition, the hypervisor will execute resource management tasks on a regular intervals on its own to keep the physical network resource usage optimal.

Example mechanisms of hypervisor 102 used to map the abstractions in the logical interface 106 to the physical network 104 according to embodiments of the invention will now be described. For example, assume there is a separate mechanism for creating, defining, and managing what should be in the logical interface - i.e., for example, how many logical forwarding elements the interface should expose and what are their interconnections alike.

If one assumes the used physical switches all provide all the primitives discussed above, the hypervisor 102 has two challenges to meet while mapping the logical interface abstractions to the physical hardware:

- Potentially limited switching capacity of individual physical forwarding elements, as well as the limited number and capacity of the ports.

- Potentially limited capacity of the TCAM tables of individual physical forwarding elements.

In the context of the data centers, the task of the network hypervisors is simplified since the network topology is likely to be a fat-tree; therefore, multi-pathing, either implemented by offline load-balancing (e.g. ECMP) or online (e.g. TeXCP), will provide unified capacity between any points in the network topology. As a result, the network hypervisor 102 can realize the required capacity even for an extremely high capacity logical switch without having a physical forwarding element with a matching capacity.

Placement problem: If the TCAM table capacity associated with physical forwarding elements is a non-
issue (for the particular control plane implementation), the network hypervisor’s tasks are simplified because it can have all the logical forwarding state in every physical forwarding element. However, if the available physical TCAM resources are more scarce, the hypervisor 102 has to be more intelligent in the placement of the logical forwarding decisions within the physical network. In a deployment where the physical network elements are not equal (in terms of the TCAM sizes), and some do have enough capacity for the logical forwarding tables, the network hypervisor 102 may use these elements for logical forwarding decisions and then use the rest only to forward packets between them. Those skilled in the art will appreciate that the exact topological location of the high capacity physical forwarding elements can be left to be a deployment specific issue, but either having them in the edge as a first-hop elements or in the core (where they are shared) is a reasonable starting point.

If the deployment has no physical forwarding elements capable of holding the complete logical forwarding table(s), the hypervisor 102 can partition the problem either by splitting the problematic logical lookup step to span multiple physical elements or using separate physical forwarding elements to implement separate logical lookup steps (if the logical forwarding is a chain of steps). In either case, the physical forwarding element should send the processed packets to the next physical forwarding element in a way that conveys the necessary context for the next to continue the processing where the previous physical forwarding stopped.

If the deployment specific limitations are somewhere between the above two extremes, the network hypervisor 102 can explicitly do trade-offs between the optimal forwarding table resource usage and optimal physical network bandwidth usage.

Finally, note that as with all the physical forwarding elements, if the forwarding capacity of an individual element with the required capacity for the logical forwarding table(s) becomes a limiting factor, the hypervisor 102 may exploit load-balancing over multiple such elements circumvent this limit.

In one particular example implementation shown in FIG. 3, the invention provides a distributed virtual network platform that distributes across multiple virtual and physical switches, and that combines both speed, security and flexibility in a novel manner. As shown in FIG. 3, the invention provides a distributed virtual switch (DVS) 108 that allows VMs to communicate across hosts and/or virtual LANs and/or subnets in an efficient manner similar to being within the same L2 network. Further, the invention allows multiple distributed virtual switches 108 to be instantiated on the same physical host or within the same data-center allowing multiple tenants to share the same physical hardware while remaining isolated both from addressing each other and consuming each others’ resources.

As shown in FIG. 3, an organization (e.g. data center tenant) has a plurality of physical hosts and VMs using services of the data center having hosts 300-A to 300-X. As shown, these include at least VMs 302-1 and 302-3 on host 300-A, VM 302-4 on host 300-C and VM 302-6 on host 300-D. Although a data center can attempt to include these VMs in a common VLAN for management and other purposes, this does not become possible when the number of VMs exceeds the VLAN size supported by the data center. Further, VLANs require configuration of the network as VMs move, and VLANs cannot extend across a subnet without an additional mechanism.

As further shown in FIG. 3, virtual switches 304 - possibly also distributed on a plurality of different hosts 300 - and physical switches 306 are used by the virtualization layer 106 of the invention and/or hypervisor 102 to collectively act as a single distributed virtual switch 308 to collectively allow these diverse VMs to communicate with each other, and further also with authorized hosts 305 (e.g. authorized users of a tenant organization which may be on a separate external customer premises, and/or connected to the resources of the data center via a public or private network), even if they are located on different hosts and/or VLANs (i.e. subnets). As mentioned above, and will be discussed in more detail below, hypervisor 102 can be used to manage the virtual network, for example by configuring QOS settings, ACLs, firewalls, load balancing, etc.

In embodiments, hypervisor 102 can be implemented by a controller using a network operating system such as that described in co-pending application Pub-No. US2009138577. However, other OpenFlow standard or other proprietary or open controllers may be used. Hypervisor 102 and/or distributed virtual switch 108 can also leverage certain techniques described in U.S. Patent Application Pub-No. US2008189769.

Virtual switches 304 can include commercially available virtual switches such as those provided by Cisco and VMware, or other proprietary virtual switches. Preferably, most or all of the virtual switches 304 include OpenFlow or other standard or proprietary protocol support for communicating with network hypervisor 102. Physical switches 306 can include any commercially available (e.g. NEC (IP8800) or HP (ProCurve 5406ZL)) or propriety switch that includes OpenFlow or other standard or proprietary protocol support such as those mentioned above for communicating with network hypervisor 102. However, in embodiments of the invention mentioned above, and described further below, some or all of the existing physical switches and routers 306 in the network are used without having flow tables affected by using tunneling.

As shown in FIG. 3, virtual switches 304 communicate with virtual machines 302, while physical switches 306 communicate with physical hosts 305.

An example host 300 includes a server (e.g. Dell, HP, etc.) running a VMware ESX hypervisor, for example. However, the invention is not limited to this example embodiment, and those skilled in the art will un-
derstand how to implement this and equivalent embodiments of the invention using other operating systems and/or hypervisors, etc. These include, for example, Citrix XenServer, Linux KVM. Moreover, it should be noted that not all of the physical hosts included in an organization managed by hypervisor 102 need to run any virtualization software (e.g. some or all of hosts 305).

[0051] An example implementation of a distributed virtual switch 108 according to an embodiment of the invention will now be described in connection with FIG. 4. As set forth above, a distributed virtual switch 108 such as that shown in FIG. 4 harnesses multiple traditional virtual switches 304 and physical switches 306 to provide a logical abstraction that is decoupled from the underlying configuration.

[0052] It can be seen in FIG. 4, and should be noted, that distributed virtual switch 108 preferably includes its own L2 and L3 logical flow tables, which may or may not be the same as the flowtables in the underlying switches 304 and 306. This is to implement the logical forwarding elements in the control plane of the virtualization layer 106 as described above.

[0053] As shown in FIG. 4, each virtual and physical switch used by distributed virtual switch 108 includes a secure channel for communicating with network hypervisor 102. This can be, for example, a communication module that implements the OpenFlow standard (See www.openflow.org) and is adapted to communicate with a controller using the OpenFlow protocol. However, other proprietary and open protocols are possible.

[0054] Each virtual and physical switch 304 and 306 also includes its own logical and physical flowtables, as well as a mapper to map an incoming packet to a logical context (i.e. such that a single physical switch may support multiple logical switches). These can be implemented using the standard flowtables and forwarding engines available in conventional switches, as manipulated by the hypervisor 102. In other words, hypervisor 102 adjusts entries in the existing flowtables so that the existing forwarding engines in 304 and 306 implement the logical and other mappings described above. It should be appreciated that switches 304 and 306 can have additional flow table entries that are not affected by the present invention, and which can be created and maintained using conventional means (e.g. network administration, policies, routing requirements, etc.).

[0055] As further shown in FIG. 4, in order to support communications across different subnets, and also to adapt to existing physical and/or virtual switches and routers that are not affected by having adjusted flow tables, the certain physical and virtual switches 306 and 304 used in the invention to implement a distributed virtual switch 108 preferably include a tunnel manager. In one example embodiment, tunnel manager uses VLANs or Generic Route Encapsulation (GRE) tunnels to a set of virtual private networks (PVNs), which function as virtual private L2 broadcast domains. Controller 110 maintains a database that maps VMs 102 to one or more associated PVNs. For each PVN controller 110 and/or switch 104 create and maintain a set of PVN tunnels connecting the hosts along which broadcast and other packets are carried. In this way, VMs 102 in the same PVN can communicate with each other, even if they are in different L2 domains and/or different hosts. Moreover, all the VMs associated with hosts in a PVN see all broadcast packets sent by VMs on other hosts within the PVN, and these packets are not seen by any hosts outside of that PVN.

[0056] There are many different ways that tunnels can be created and/or how hosts can be interconnected via PVNs using tunnel manager 204 in accordance with the invention, as will be appreciated by those skilled in the art.

Claims

1. A method of managing networking resources by a network hypervisor (102) in a network (104) comprising a plurality of hosts (300) and physical forwarding elements (304, 306), the method being characterised by the steps of:

   identifying a first set of virtual machines (302) which use a first set of the plurality of hosts and are communicatively coupled to a first set of the physical forwarding elements (304, 306); identifying a second set of virtual machines (302) which use a second set of the plurality of hosts (300) and are communicatively coupled to a second set of the physical forwarding elements (304, 306), wherein at least one of the physical forwarding elements (304, 306) in the first set is also in the second set;

   providing a first distributed virtual switch (308) arranged to handle communications between the virtual machines (302) of the first set, wherein a first virtual machine of the first set of virtual machines that is on a first virtual local area network VLAN communicates with a second virtual machine of the first set of virtual machines that is on a second VLAN via the first distributed virtual switch (308); and

   providing a second distributed virtual switch (308) arranged to handle communications between the virtual machines (302) of the second set of virtual machines;

   wherein said first and second distributed virtual switches (308) are arranged to maintain isolation between the first and second sets of virtual machines (302).

2. The method of claim 1, wherein the physical forwarding elements (304, 306) comprise one or more virtual switches (304).

3. The method of claim 1, wherein the physical forward-
ing elements (304, 306) comprise one or more physical switches (306).

4. The method of claim 1, wherein the first and second distributed virtual switches (308) further handle communications between external hosts (305) and the first and second sets of virtual machines (302), respectively.

5. The method of claim 1, wherein the first and second sets of virtual machines (302) are associated with first and second data center tenants, respectively.

6. The method of claim 1, further comprising:

- providing a logical flow table and a physical flow table in one of the physical forwarding elements (304, 306);
- determining a mapping between the logical flow table and a physical flow table; and
- causing the physical forwarding element (304, 306) to forward packets using the logical flow table and the physical flow table and the determined mapping.

7. The method of claim 4, wherein the one physical forwarding elements (304, 306) is used by both the first and second sets of virtual machines (302).

8. The method of claim 1, wherein certain of the plurality of hosts (300) are located in geographically separate premises.

9. The method of claim 1, wherein certain of the first set of virtual machines (302) are in different subnets.

10. A network (104), comprising:

- a plurality of hosts (300); and
- a plurality of physical forwarding elements (304, 306) coupled to the hosts (300);

characterised in that the network further comprises:

- a first set of virtual machines (302) which are communicatively coupled to a first set of the plurality of hosts (300) and physical forwarding elements (304, 306);
- a second set of virtual machines (302) which are communicatively coupled to a second set of the plurality of hosts (300) and physical forwarding elements (304, 306), wherein at least one of the physical forwarding elements (304, 306) in the first set is also in the second set;
- a first distributed virtual switch (308) arranged to handle communications between the virtual machines (302) of the first set, wherein a first virtual machine of the first set of virtual machines that is on a first virtual local area network VLAN communicates with a second virtual machine of the first set of virtual machines that is on a second VLAN via the first distributed virtual switch (308); and
- a second distributed virtual switch (308) arranged to handle communications between the virtual machines (302) of the second set of virtual machines;

wherein said first and second distributed virtual switches (308) are arranged to maintain isolation between the first and second sets of virtual machines (302).

11. The network of claim 10, wherein the first and second distributed virtual switches (308) further handle communications between external hosts (305) and the first and second sets of virtual machines (302), respectively.

12. The network of claim 10, wherein the first and second sets of virtual machines (302) are associated with first and second data center tenants, respectively.

13. The network of claim 10, wherein certain of the first set of virtual machines are in different subnets.

14. The network of claim 10, further comprising:

- tunnels through certain of the physical forwarding elements (304, 306).

15. The network of claim 14, wherein the tunnels comprise GRE tunnels.

Patentansprüche

1. Verfahren zum Managen von Netzwerkressourcen durch einen Netzwerk-Hypervisor (102) in einem mehrere Hosts (300) und physische Forwarding-Elemente (304, 306) aufweisenden Netzwerk (104),
durch gekennzeichnet, dass das Verfahren folgende Schritte aufweist:

Identifizieren eines ersten Satzes von virtuellen Maschinen (302), die einen ersten Satz der mehreren Hosts benutzen und kommunikativ mit einem ersten Satz physischer Forwarding-Elemente (304, 306) verbunden sind; Identifizieren eines zweiten Satzes von virtuellen Maschinen (302), die einen zweiten Satz der mehreren Hosts (300) benutzen und kommunikativ mit einem zweiten Satz physischer Forwarding-Elemente (304, 306) verbunden sind, wobei wenigstens eines der physischen Forwar-
Bereitstellen eines ersten verteilten virtuellen Switches (308), der für die Abwicklung von Kommunikationen zwischen den virtuellen Maschinen (302) des ersten Satzes angeordnet ist, wobei eine erste virtuelle Maschine des ersten Satzes von virtuellen Maschinen, die sich auf einem ersten virtuellen lokalen Netzwerk VLAN befinden, mit einer zweiten virtuellen Maschine des ersten Satzes von virtuellen Maschinen, die sich auf einem zweiten VLAN befinden, über den ersten verteilten virtuellen Switch (308) kommuniziert; und

Bereitstellen eines zweiten verteilten virtuellen Switches (308), der für die Abwicklung von Kommunikationen zwischen den virtuellen Maschinen (302) des zweiten Satzes von virtuellen Maschinen angeordnet ist; wobei die ersten und zweiten verteilten virtuellen Switches (308) für ein Aufrechterhalten einer Isolation zwischen den ersten und zweiten Sätzen von virtuellen Maschinen (302) angeordnet sind.

2. Verfahren nach Anspruch 1, wobei die physischen Forwarding-Elemente (304, 306) einen oder mehrere virtuelle Switches (304) aufweisen.

3. Verfahren nach Anspruch 1, wobei die physischen Forwarding-Elemente (304, 306) einen oder mehrere virtuelle Switches (306) aufweisen.

4. Verfahren nach Anspruch 1, wobei die ersten und zweiten verteilten virtuellen Switches (308) ferner Kommunikationen zwischen externen Hosts (305) und den ersten beziehungsweise zweiten Sätzen von virtuellen Maschinen (302) abwickeln.

5. Verfahren nach Anspruch 1, wobei die ersten beziehungsweise zweiten Sätze von virtuellen Maschinen (302) ersten beziehungsweise zweiten Rechenzentrumsmietern zugeordnet sind.

6. Verfahren nach Anspruch 1, ferner aufweisend:

Bereitstellen einer logischen Flusstabelle und einer physischen Flusstabelle in einem der physischen Forwarding-Elemente (304, 306);

Bestimmen eines Mappings zwischen einer logischen Flusstabelle und einer physischen Flusstabelle; und

11. Netzwerk nach Anspruch 10, wobei die ersten und zweiten verteilten virtuellen Switches (308) ferner Kommunikationen zwischen externen Hosts (305) und den ersten beziehungsweise zweiten Sätzen von virtuellen Maschinen (302) abwickeln.

12. Netzwerk nach Anspruch 10, wobei die ersten beziehungsweise zweiten Sätze von virtuellen Maschinen (302) ersten beziehungsweise zweiten Rechenzentrumsmitgliedern zugeordnet sind.


14. Netzwerk nach Anspruch 10, ferner aufweisend:
   Tunnel durch bestimmte der physischen Forwarding-Elemente (304, 306).

15. Netzwerk nach Anspruch 14, wobei die Tunnel GRE-Tunnel aufweisen.

Revendications

1. Procédé pour gérer des ressources de réseau par l’intermédiaire d’un hyperviseur de réseau (102) dans un réseau (104) comprenant une pluralité d’hôtes (300) et d’éléments de transfert physiques (304, 306), le procédé étant caractérisé par les étapes suivantes :

   - identifier un premier ensemble de machines virtuelles (302) qui utilisent un premier ensemble de la pluralité d’hôtes et qui sont couplées, de manière communicative, à un premier ensemble des éléments de transfert physiques (304, 306) ;
   - identifier un second ensemble de machines virtuelles (302) qui utilisent un second ensemble de la pluralité d’hôtes (300) et qui sont couplées, de manière communicative, à un second ensemble des éléments de transfert physiques (304, 306), où au moins un des éléments de transfert physiques (304, 306) situé dans le premier ensemble se trouve également dans le second ensemble ;
   - pourvoir un premier commutateur virtuel distribué (308) conçu pour traiter des communications entre les machines virtuelles (302) du premier ensemble, où une première machine virtuelle du premier ensemble de machines virtuelles qui se trouve sur un premier réseau local virtuel VLAN communique avec une seconde machine virtuelle du premier ensemble de machines virtuelles qui se trouve sur un second VLAN, via le premier commutateur virtuel distribué (308) ; et

   - pourvoir un second commutateur virtuel distribué (308) conçu pour traiter des communications entre les machines virtuelles (302) du second ensemble de machines virtuelles ; où lesdits premier et second commutateurs virtuels distribués (308) sont conçus pour maintenir un isolement entre les premier et second ensembles de machines virtuelles (302).

2. Procédé selon la revendication 1, dans lequel les éléments de transfert physiques (304, 306) comprennent un ou plusieurs commutateurs virtuels (304).

3. Procédé selon la revendication 1, dans lequel les éléments de transfert physiques (304, 306) comprennent un ou plusieurs commutateurs physiques (306).

4. Procédé selon la revendication 1, dans lequel les premier et second commutateurs virtuels distribués (308) traitent en outre des communications entre des hôtes externes (305) et les premier et second ensembles de machines virtuelles (302), respectivement.

5. Procédé selon la revendication 1, dans lequel les premier et second ensembles de machines virtuelles (302) sont associés à des premier et second locataires de centre informatique, respectivement.

6. Procédé selon la revendication 1, comprenant en outre les étapes suivantes :

   - pourvoir une table de flux logiques et une table de flux physiques dans l’un des éléments de transfert physiques (304, 306) ;
   - déterminer une correspondance entre la table de flux logiques et une table de flux physiques ;
   - et
   - amener l’élément de transfert physique (304, 306) à transférer des paquets au moyen de la table de flux logiques et de la table de flux physiques et de la correspondance déterminée.

7. Procédé selon la revendication 4, dans lequel l’un des éléments de transfert physiques (304, 306) est utilisé à la fois par les premier et second ensembles de machines virtuelles (302).

8. Procédé selon la revendication 1, dans lequel certains parmi la pluralité d’hôtes (300) sont situés dans des locaux séparés géographiquement.

9. Procédé selon la revendication 1, dans lequel certaines machines virtuelles parmi les premier et second ensembles de machines virtuelles (302) se trouvent dans des sous-réseaux différents.
10. Réseau (104), comprenant :

une pluralité d’hôtes (300) ; et
une pluralité d’éléments de transfert physiques (304, 306) couplés aux hôtes (300) ;
caractérisé en ce que le réseau comprend en outre :

un premier ensemble de machines virtuelles (302) qui sont couplées, de manière communicative, à un premier ensemble de la pluralité d’hôtes (300) et d’éléments de transfert physiques (304, 306) ; un second ensemble de machines virtuelles (302) qui sont couplées, de manière communicative, à un second ensemble de la pluralité d’hôtes (300) et d’éléments de transfert physiques (304, 306), où au moins un des éléments de transfert physiques (304, 306) dans le premier ensemble se trouve également dans le second ensemble ;
un premier commutateur virtuel distribué (308) conçu pour traiter des communications entre les machines virtuelles (302) du premier ensemble, où une première machine virtuelle du premier ensemble de machines virtuelles qui se trouve sur un premier réseau local virtuel VLAN communique avec une seconde machine virtuelle du premier ensemble de machines virtuelles qui se trouve sur un second VLAN, via le premier commutateur virtuel distribué (308) ; et un second commutateur virtuel distribué (308) conçu pour traiter des communications entre les machines virtuelles (302) du second ensemble de machines virtuelles ; où lesdits premier et second commutateurs virtuels distribués (308) sont conçus pour maintenir un isolement entre les premier et second ensembles de machines virtuelles (302).

11. Réseau selon la revendication 10, dans lequel les premier et second commutateurs virtuels distribués (308) traitent en outre des communications entre des hôtes externes (305) et les premier et second ensembles de machines virtuelles (302), respectivement.

12. Réseau selon la revendication 10, dans lequel les premier et second ensembles de machines virtuelles (302) sont associés à des premier et second locataires de centre informatique, respectivement.

13. Réseau selon la revendication 10, dans lequel certaines machines virtuelles du premier ensemble de machines virtuelles se trouvent dans des sous-réseaux différents.

14. Réseau selon la revendication 10, comprenant en outre :

des tunnels traversant certains des éléments de transfert physiques (304, 306).

15. Réseau selon la revendication 14, dans lequel les tunnels comprennent des tunnels GRE.
FIG. 1

Network Hypervisor 102

Network Virtualization Layer

Physical Network

FIG. 2

Logical
Lookup S204

Mapping
Lookup S202

Packet in

Mapping
Lookup S206

Packet out

Physical
Lookup S208
FIG. 3
REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 2009138577 A [0047]
- US 2008189769 A [0047]