虚拟网络初探

Weibo Rao

随着服务器虚拟化的广泛应用，在数据中心网络中，虚拟网络端口数量已经超过物理网络，足以引起人们的关注。虚拟化也颠覆了传统数据中心网络设计的两个假设：多个虚拟机会共享同一个物理的网卡，打破传统的服务器与网卡一对一的关联；并且，由于虚拟机具备漂移的能力，虚拟机与虚拟网络的连接是动态的，而不是在物理环境中相对静态的连接。本文对虚拟网络领域进行初步探讨，希望给对该领域感兴趣的技术爱好者以启发，起到抛砖引玉的效果。

为什么需要虚拟交换机

按照维基百科的定义，虚拟网络是由虚拟网络连接（部分或全部）组成的计算机网络。其中，虚拟网络连接是一种非物理连接（有线或无线），采用网络虚拟化的技术连接计算设备。虚拟网络通常有两种形式：一种是基于协议的虚拟化（如 VLAN、VPN，通过插入报文头实现），另一种是基于虚拟设备的（如虚拟机网络）。本文主要关注后者，如无单独注明，下文的虚拟网络均指基于虚拟设备的网络。

虚拟网络是伴随服务器虚拟化而出现的，早期的虚拟网络主要处理一个很有趣的问题，即物理服务器内的多个虚拟机如何共享同一个物理的网卡？虚拟机一方面需要共享物理网卡与外界资源通信，另一方面，同一广播域内的虚拟机之间也需要通信。如果虚拟机管理程序 Hypervisor 只管将来自虚拟机的报文转发给上层交换机，由于网桥的 MAC 过滤工作原理，上层交换机不会将来自同一端口的报文转发回去。

最早，在 2000 年，VMware 通过软件的方式，在 Hypervisor 中增加虚拟交换机 vSwitch 来解决上述问题，而网络厂商则希望采取硬件方式，通过外部交换机来处理该问题，不过这是十年之后才出现的，以 IEEE 802.1BR 和 IEEE 802.1Qbg 为代表。本文主要关注基于软件的虚拟交换机方式。如图 1 所示：

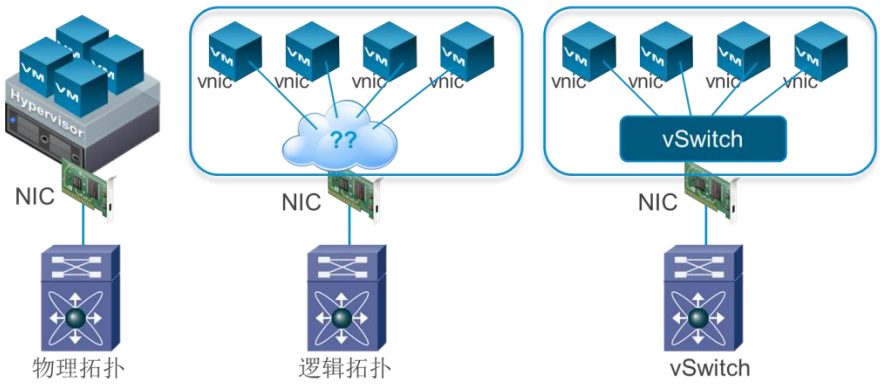

图 1 虚拟网络逻辑拓扑及 vSwitch

探索 vSwitch

vSwitch 是虚拟机管理程序 Hypervisor 中内置的组件，模拟二层交换功能，对内提供虚拟机的接入端口，对外与服务器物理网卡相连。同二层交换机一样，vSwitch 提供 IEEE 802.1Q VLAN 标记功能以及 MAC 地址表用于转发以太帧。vSwitch 是基于单个 vSphere 主机来配置和管理的，其配置文件以文本方式存储在/etc/vmware/esx.conf 中。有趣的是 vSwitch 是支持 CDP 协议的，/etc/vmware/esx.conf 配置中有关 CDP 的配置如下：

```
/net/vswitch/child[0000]/cdp/status = "both"
-- 将缺省的"listen"修改为"both"，那么，上层交换机可以通过 CDP 协议发现 vSwitch

Device ID: CiscoUCSC220.n1kdemo
Entry address(es):
Platform: VMware ESX,  Capabilities: Switch
Interface: GigabitEthernet1/0/7,  Port ID (outgoing port): vmnic0
Holdtime : 138 sec
Version :
Releasebuild-1331820
advertisement version: 2
以下输出在此省略……
```

vSwitch 与物理交换机最大的不同在于：在物理环境中，物理服务器的网络连接参数，如 VLAN、ACL，是由物理交换机来配置和掌管的；而 vSwitch 和虚拟机网卡的连接参数，是通过网络标签（Port-Group）配置在虚拟机上的，由虚拟机掌管连接参数，如图 2 所示：

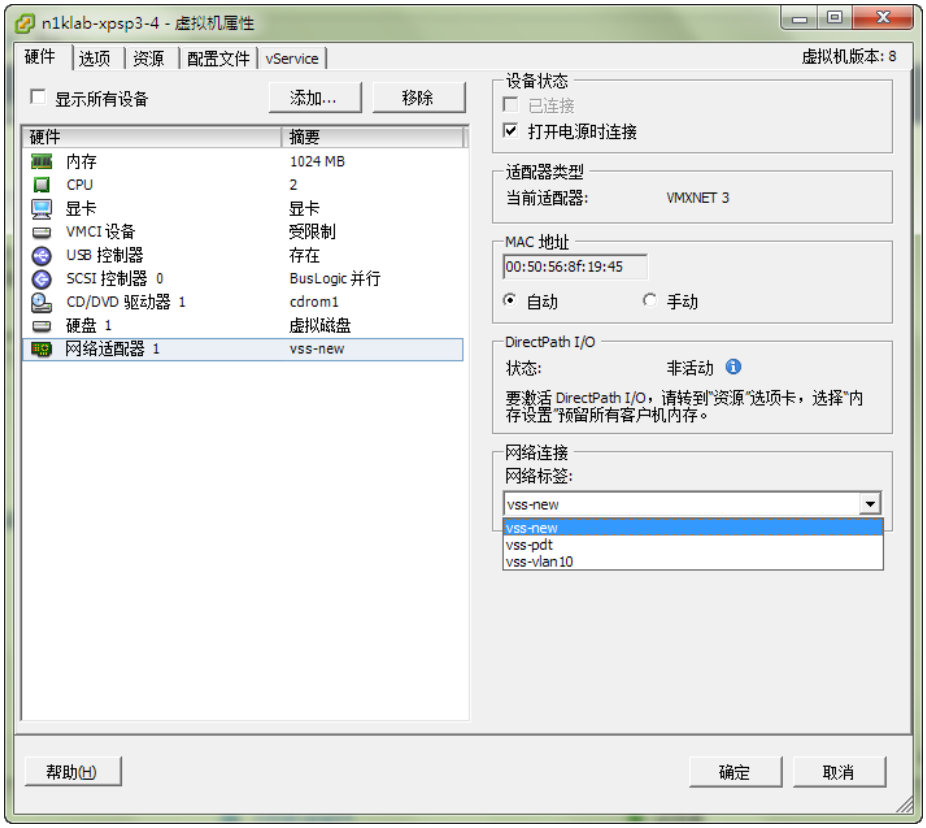

图 2 vSphere 客户端中编辑虚机属性

网络标签定义了 VLAN、安全策略、流量整形以及上游链路负载均衡等配置信息，其中安全策略主要是指网卡是否配置为混杂模式，以及是否接受虚拟机 vNIC MAC 地址改变，并不是传统意义上的安全访问控制 ACL。虚拟机需要漂移，在所有虚拟服务器中，对应的网络标签配置需保持一致，并且，服务器物理网卡与物理交换机互联的端口配置需为 Trunk 模式，以支持多 VLAN 环境。

基于软件的 vSwitch 是通过 CPU 转发的，转发性能同 CPU 性能及内存总线带宽直接相关。在最新发布的 vSphere 5.5 平台下，笔者在一台 2 路 Intel Xeon E5-2640 CPU 服务器内安装了两台 64 位的 CentOS 虚拟机。虚拟网卡选择 VMXNET3 型号，此为 VMware 新一代的网卡，是半虚拟化网卡，优化了性能，支持 10GE，巨型帧等特性。通过 iperf 测试了两者的吞吐量，测试结果如图 3 所示：

```
[root@centos64-lab etc]# iperf -c 10.0.50.35 -w 2048000 -m -t 20
------------------------------------------------------------
Client connecting to 10.0.50.35, TCP port 5001
TCP window size: 3.91 MByte (WARNING: requested 1.95 MByte)
------------------------------------------------------------
[  3] local 10.0.50.30 port 49314 connected with 10.0.50.35 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-20.0 sec  54.1 GBytes  23.3 Gbits/sec
[  3] MSS size 8960 bytes (MTU 9000 bytes, unknown interface)
```

图 3 iperf 测试虚拟机吞吐量

诚然，笔者调大 TCP 滑动窗口值以及网卡 MTU 为 9000，均会提升传输效率，但测试结果竟然达到 23.3Gbps 之多。这是由于在同一服务器内的两个虚拟机之间的通信，不受物理网卡速率的限制，操作系统按最大可能的吞吐量进行报文传输。另一个有趣的现象是：32 位 Windows XP 安装 vmxnet3 网卡，网卡速率显示为 1.4Gbps，在物理环境中，哪儿有这样的网卡？

上文的 vSwitch 是 VMware 早期的标准交换机，至今也是大部分服务器虚拟化环境中使用的虚拟交换机。如前文所述，如果要支持虚机漂移（vMotion），所有服务器 vSwitch 配置需要保持同步，否则，在 vCenter 进行虚机漂移时，提示网络配置冲突，而不能进行迁移。因此，这涉及到一个很大的维护管理的开销问题。并且，标准 vSwitch 的功能有限，仅能满足基本的连通性，像 PVLAN、安全控制、端口镜像分析、链路聚合协议 LACP 等均不支持。

2009 年，VMware 正式发布了分布式虚拟交换机（Distributed Virtual Switch DVS），解决了集中管理的问题。简言之，分布式虚拟交换机可同时在多个服务器上创建和维护网络标签（Port-Group），并且集中在 vCenter 进行管理，解决了网络标签配置一致性的问题，从而很好的支持虚机漂移。部署 DVS 需要安装 vCenter，并且 vSphere 需要企业加强版许可。如图 4 所示：DVS 在 vCenter 中的呈现的拓扑，左侧为虚拟机连接图示，右侧为物理网络连接图示。

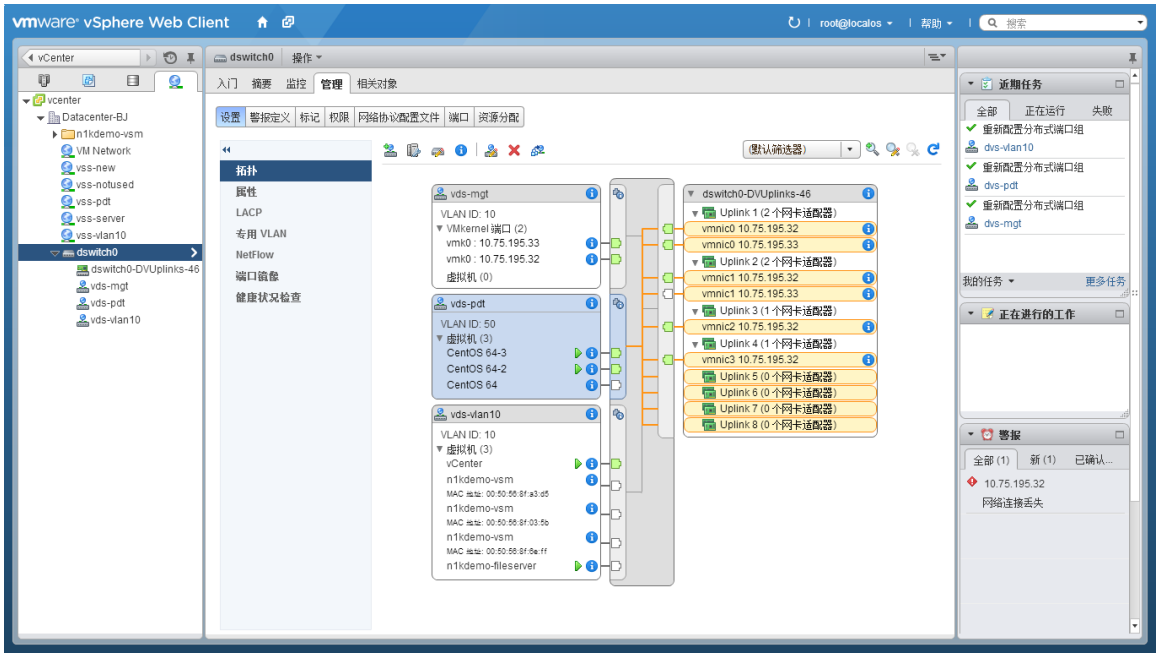

图 4 在 vCenter 中管理 DVS

DVS 固然解决了集中管理的问题，然而，这个虚拟交换机具体由谁来管理呢？一般用户的做法是由采购虚拟化软件的部门管理，管理职责往往会落到服务器部门。在这种物理网络和虚拟网络各自独立管理的情况下，如何保持虚拟网络和物理网络之间的无缝衔接？网络团队若只能管理到物理交换机，面对物理服务器配置大量的 Trunk 接口，而对下层的虚拟世界一无所知。很显然需要部门间的大量沟通、协调的工作量，很多信息需要通过邮件和电子表格传递，导致较低的效率。若虚拟网络管理权限交给网络团队，网络团队需通过 vCenter 来管理 DVS，一方面是管理方式发生转变，与物理环境管理体验不一致，效果仅仅是将“部门间”的沟通变为“部门内”的沟通，无实质性改善；另一方面是需要同服务器部门共用 vCenter，也容易导致冲突。简言之，服务器团队和网络团队的边界模糊化了，需要跨部门紧密协作才能健康运转。

Cisco Nexus 1000v ：无缝衔接物理与虚拟网络

VMware DVS 提供了第三方集成接口，Cisco 和 VMware 通过紧密协作，在 VMWorld 2008 发布了 Nexus 1000v，并于 2009 年 5 月正式发货（First Commercial Ship FCS）。利用 DVS 提供的 API，Nexus 1000v 将思科数据中心网络操作系统（NX-OS）的智能延伸至虚拟网络，是业内第一个第三方纯软件的虚拟交换机。Nexus 1000v 是完全基于 802.1Q 的标准交换机，纯软件解决方案，可无缝融入现有的物理环境，下文以 VMware 版本的 Nexus 1000v 进行说明。

Nexus 1000v 被发明出来的主要使命是重建网络和服务团队的责任边界，将网络延伸至虚拟机层，从而实现网络管理与服务器管理互不干扰的运作模式。从管理上来看，Nexus 1000v 对服务器

管理员提供与 DVS 一致的管理界面，对网络管理员提供 NX-OS 的管理体验（CLI 或者 GUI）。下文将向您详细介绍 Nexus 1000v 如何实现这种运作模式。

Nexus 1000v 由两个基本组件组成，虚拟引擎模块（Virtual Supervisor Module VSM）和虚拟以太网模块（Virtual Ethernet Module VEM），其架构如图 5：

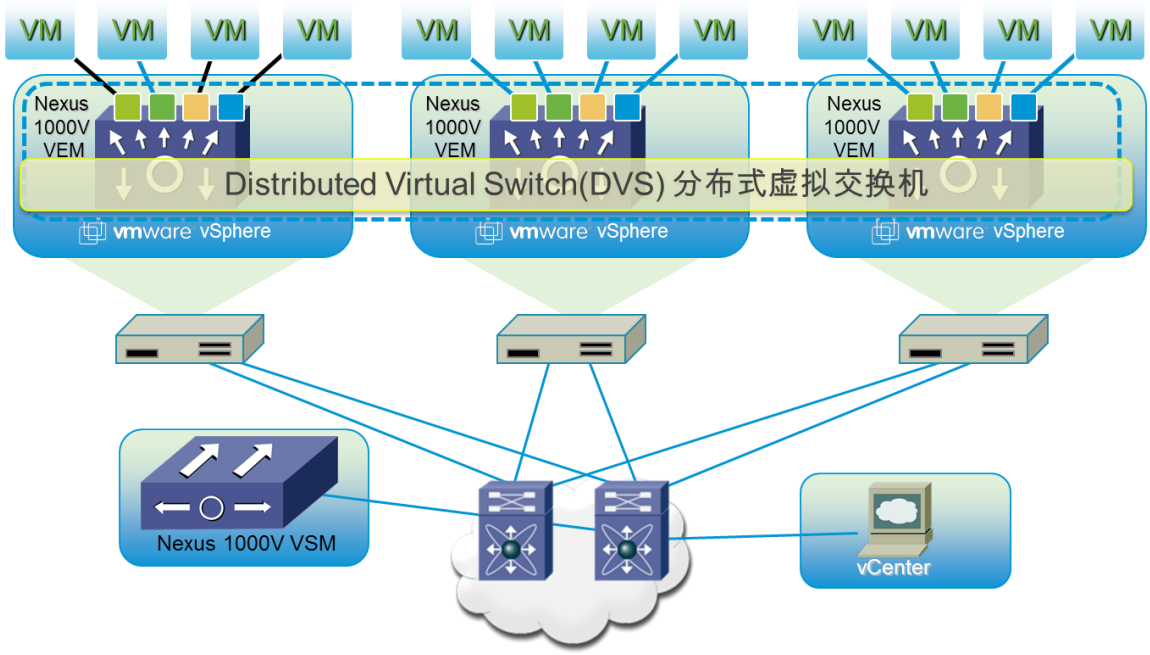

图 5 Nexus 1000v 架构

虚拟引擎模块（Virtual Supervisor Module VSM）：

- 提供 Nexus 1000v 虚拟交换机的控制和管理平面；
- 运行独立版本的 NX-OS 系统，提供 CLI 界面，负责与 vCenter 集成；
- 控制 VEM，当前 4.2(1)SV2(2.1a)版本支持 128 个 VEM，将来计划支持更多；
- 可作为虚机安装或者采用独立的硬件（Nexus 1110-X 和 Nexus 1110-S）；
- 虚拟机可通过 ISO 文件 Boot 或者通过 vCenter 导入 OVA 虚机模板手工安装，也可以通过基于 Java 的自动安装程序安装，具体参考 Nexus 1000v 的安装指南。

虚拟以太网模块（Virtual Ethernet Module VEM）：

- 通过 VIB 安装包安装在每一个 vSphere 主机上（Agent），作为 vSphere 主机上运行的进程，替代 VMware 的 vSwitch 或 DVS，为每个虚拟机提供接入端口；
- 作为 Nexus 1000v 的数据平面，转发虚机与虚机之间以及虚机和外部资源之间的流量，转发路径并不经过 VSM；
- 充分利用 DVS 提供的 API，将 NX-OS 的高级特性，如 QoS、PVLAN、SPAN、Netflow 等物理网络的特性延伸到虚拟网络。

事实上，完全可以将 Nexus 1000v 同 Nexus 7000 进行类比，VSM 如同 Nexus 7000 的控制引擎，VEM 如同 Nexus 7000 的 I/O 板卡。VSM 与 VEM 之间的通信机制同 N7K 引擎与板卡通信机制一样，都采用 NX-OS 的 MTS（Message and Transaction System）和 AIPC（Asynchronous Inter-Process Communication），只不过 N7K 有物理的背板连接，而 N1KV 是通过 L2/L3 网络进行通信的，当然，其通道是加密的。N1KV“虚拟机箱”和模块化交换机类比关系如图 6 所示：

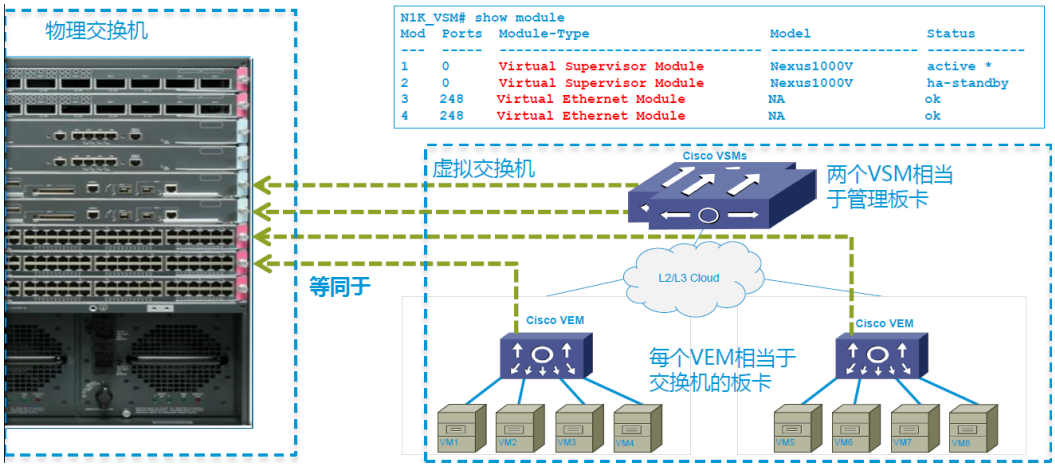

图 6 Nexus 1000v “虚拟机箱”

每一个 VEM 在 VSM 的管理界面中呈现出一块线卡，VSM 使用 vSphere 的 Server UUID 来唯一标识 VEM。VEM 向 VSM 注册时，向 VSM 申请槽位号，VSM 作出响应，为 VEM 分配槽位号，1、2 槽位预留给主备 VSM 引擎，3-130 提供给 VEM。Show module 查看各槽位信息，如下：

```
n1kdemo-vsm# show module
Mod  Ports  Module-Type                          Model               Status
---  -----  ---------------------------------   ------------------  ------------
1    0      Virtual Supervisor Module           Nexus1000V          active *
2    0      Virtual Supervisor Module           Nexus1000V          ha-standby
3    332    Virtual Ethernet Module             NA                  ok
4    332    Virtual Ethernet Module             NA                  ok
Mod  Sw                   Hw
---  ------------------   ------------------------------------------------
1    4.2(1)SV2(2.1)       0.0
2    4.2(1)SV2(2.1)       0.0
3    4.2(1)SV2(2.1)       VMware ESXi 5.5.0 Releasebuild-1331820 (3.2)
4    4.2(1)SV2(2.1)       VMware ESXi 5.5.0 Releasebuild-1331820 (3.2)
Mod  Server-IP        Server-UUID                          Server-Name
---  ---------------  ------------------------------------  --------------------
1    10.75.195.35     NA                                    NA
2    10.75.195.35     NA                                    NA
3    10.75.195.32     4c4c4544-0058-5310-8036-b9c04f425831  10.75.195.32
4    10.75.195.33     6fcb2715-08f2-7d4c-a412-bacc3d2a663e  10.75.195.33
```

槽位信息在 running-config 中也有所体现，配置是自动生成的：

```
vem 3
  host id 4c4c4544-0058-5310-8036-b9c04f425831
vem 4
  host id 6fcb2715-08f2-7d4c-a412-bacc3d2a663e
```

如前文所述，VSM 和 VEM 之间通过 L2/L3 网络进行通信，VSM 虚拟机有三个网卡，按照顺序分别是：

- 网卡 1—控制端口（control 0）：将 NX-OS 的 AIPC 协议通过以太网延伸，VSM 通过 MTS over AIPC 对 VEM 进行配置；VSM 主备之间的同步信息也通过 Control 接口传递。控制接口有两种模式，L2 和 L3，其中 L2 模式要求 VSM 和 VEM 在同一 VLAN，Cisco 推荐采用 L3 模式。L3 采用 UDP 4785 端口进行通信，通道加密。
- 网卡 2—管理端口（mgmt0）：用于系统管理以及 VSM 和 vCenter 之间的通信，并且，在 L3 控制模式，缺省 mgmt0 也负责与 VEM 之间的通信，此时，控制口仅用于主备 VSM 之间的心跳通信，而数据口则空闲；
- 网卡 3—数据端口（packet）：用于 VEM 需要将报文送回 VSM 进行后续处理时的通信，例如 CDP 报文和 IGMP。

如果采用 Nexus 1110 硬件设备，也适用上述原理，区别是硬件设备可以为 VSM 提供专有的硬件资源，如 CPU、内存和网卡。VSM 的初始配置，在虚拟机启动后，通过 Setup 命令，按照提示进行初始化配置。缺省会选择 L3 控制模式，通过 mgmt0 作为控制接口，也可选择 control0 作为控制接口，具体参考 Nexus 1000v 的安装指南。

VEM 是 Nexus 1000v 分布式交换机的板卡，其转发数据是各自独立的：

- 每个 VEM 都能独立学习 MAC 地址，拥有独立的 MAC 地址表，相互间不同步；
- 虚拟机的 MAC 地址是静态的，并且在虚拟机在线期间都不会过期；
- 在 VSM 能查看所有的 MAC 地址表，会看到有重复条目，其中显示为 Static 的是本地 VEM，显示 Dynamic 的是从其他外部学到的；
- 每个 VEM 的每个 VLAN 可最多支持 4096 个 MAC 地址，每个 VEM 共支持 32000 个 MAC；
- 在同一个 VEM 上的虚拟机通信在 VEM 本地交换；
- 在不同的 VEM 或在网段以外的通信都由上游交换机处理；
- VSM 不在数据层面——当配置应用到 VEM 之后，VEM 可以在没有 VSM 存在的情况下单独工作，即所谓 headless（无头）状态。 在 headless 状态下无法进行管理作业，但只要 VEM 不失去其配置就可以无限制地持续工作，数据传送也就不会受到影响，新的配置无法更新到 VEM，但在 VSM 恢复后可以进行配置同步；
- VEM 通过 vSphere 主机的 vmkernel 接口与 VSM 进行通信；
- 在当前的 4.2(1)SV2(2.1a)版本 VSM 和 VEM 之间最大通信时延可达 100ms，支持跨数据中心部署。

另一个与物理交换机的不同点是：VEM 不支持 STP，其防环机制可以用图 7 表示：

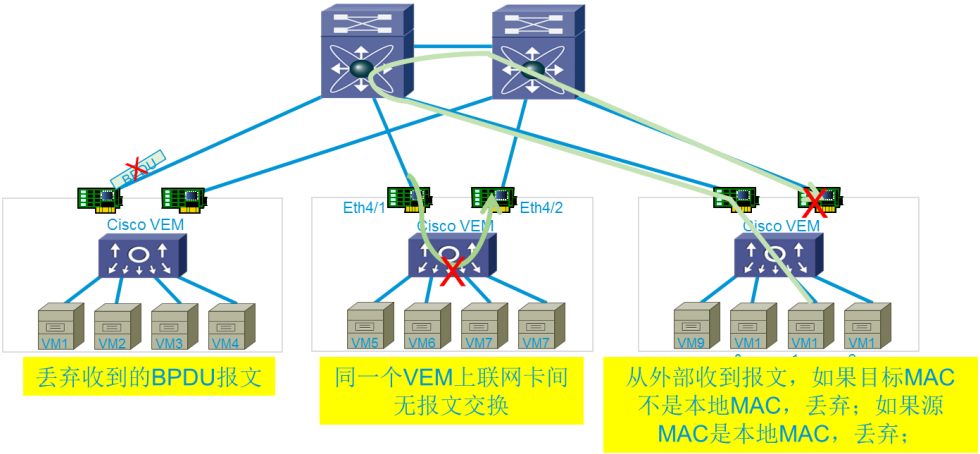

图 7 VEM 环路防范机制

对于上游交换机，其与 vSphere 主机互联的接口，建议配置为边缘端口，并关闭 BPDU，开启 BPDU 过滤。Nexus 1000v 支持 LACP，如果上层交换机采用 vPC 或者 VSS 技术，可以实现跨机箱链路聚合。

仅有 VSM 和 VEM 这两个组件，Nexus 1000v 还不能正常工作，还需与 vCenter 进行紧密集成才可形成完整的解决方案。vCenter 对外提供 API 接口（VMware Virtual Infrastructure Methodology (VIM) API），第三方可通过插件的方式向其注册，与其集成。VSM 与 vCenter 之间通过 API over SSL 通信；Nexus 1000v 与 vCenter 的注册由 VSM 主动发起，而不是在 vCenter 中添加 DVS 的来进行配置。vCenter 需要导入 VSM 的 XML 插件（从 http://<VSM-IP-address>下载），建立互信关系，方可完成 Nexus 1000v 的注册。在 VSM 上添加以下配置即可完成与 vCenter 的注册：

```
svs connection vcenter
  protocol vmware-vim
  remote ip address 10.75.195.31 port 80
  vmware dvs datacenter-name Datacenter-BJ
  connect
命令执行后，在 VSM 产生如下 Log 日志：
n1kdemo-vsm %VMS-5-CONN_CREATE: Connection 'vcenter' created.
n1kdemo-vsm %VMS-5-CONN_CONNECT: Connection 'vcenter' connected to the vCenter Server.
n1kdemo-vsm %VMS-5-DVS_CREATE: dvswitch 'n1kdemo-vsm' created on the vCenter Server.
```

一旦 VSM 与 vCenter 连接成功，N1K 就会出现在 vCenter 的网络清单中，与 DVS 呈现给虚拟化/服务器管理人员的管理接口、界面一致，在 VSM 上可通过命令查看连接状态：

```
n1kdemo-vsm# show svs connections
connection vcenter:
    ip address: 10.75.195.31
    remote port: 80
    protocol: vmware-vim https
    certificate: default
    datacenter name: Datacenter-BJ
    admin: n1kUser(user)
    max-ports: 8192
    DVS uuid: 08 a3 29 50 2b b5 8e b0-4d 45 d8 f0 63 9e bb be
    config status: Enabled
    operational status: Connected
    sync status: Complete
    version: VMware vCenter Server 5.5.0 build-1312298
    vc-uuid: 5000A09F-DE82-4C47-AD03-5902B4805300
```

Nexus1000v 采用端口配置模板（Port-Profile）进行配置。Port-Profile 是一组命令集合，用于定义一系列相同类型端口的属性，可被应用到物理或虚拟接口上，当其被应用于特定的端口配置后，这些端口就继承了这个 Port-Profile 所定义的端口属性。应用到物理端口的 Port-Profile 是 Ethernet 类型，面向虚拟机网卡的虚拟端口的 Port-Profile 是 vEthernet 。Port-Profile 包含以下命令集：

```
VLAN
Private VLAN (PVLAN)
Virtual Extensible LAN (VXLAN)
Access control list (ACL)
Quality of service (QoS)
Catalyst Integrated Security Features (CISF)
Virtual Service Domain (VSD)
Port channel
Port security
Link Aggregation Control Protocol (LACP)
LACP Offload
NetFlow
Virtual Router Redundancy Protocol (VRRP)
Unknown Unicast Flood Blocking (UUFB)
```

Port-Profile 不仅仅在 Nexus 1000v 上才有，在其他 Nexus 系列交换机上也都支持。您可能会发现该 Port-Profile 与 VMware Port-group 是非常类似的，并且 Port-Profile 配置包含更多的特性。实际上，Nexus 1000v 上所配置的 Port-Profile 与 vCenter 中对应 DVS（Nexus 1000v）的确具有一一对应的关系。

Nexus 1000v 的 Port-Profile 配置如下所示：

```
n1kdemo-vsm# show run port-profile
port-profile type ethernet n1k-uplink
  vmware port-group
  switchport mode trunk
  switchport trunk allowed vlan 1-3967,4048-4093
  no shutdown
  system vlan 10,20
  state enabled
port-profile type vethernet n1k-l3-ctrl
  capability l3control
  !capability l3control 表示需要赋予该 vethernet L3 Control 的能力，应用到 VEM 对应的 vSphere 主机的 vmkernel 接口上。
  vmware port-group
  switchport mode access
  switchport access vlan 20
  no shutdown
  system vlan 20
  state enabled
port-profile type vethernet n1k-production
  vmware port-group
  switchport mode access
  switchport access vlan 50
  no shutdown
  state enabled
命令执行后，在 VSM 上产生如下日志记录：
n1kdemo-vsm %VMS-5-DVPG_CREATE: created port-group 'n1k-uplink' on the vCenter Server.
n1kdemo-vsm %VMS-5-DVPG_CREATE: created port-group 'n1k-l3-ctrl' on the vCenter Server.
n1kdemo-vsm %VMS-5-DVPG_CREATE: created port-group 'n1k-production' on the vCenter
Server.
```

在 Nexus 1000v 的 Port-Profile 配置中，添加 vmware port-group 以及 state enable 命令后，VSM 通过 API 在 vCenter 对应的 DVS（n1kdemo-vsm）上创建对应的 Port-group。

在 vCenter 网络清单中，n1kdemo-vsm 拓扑如图 8 所示：

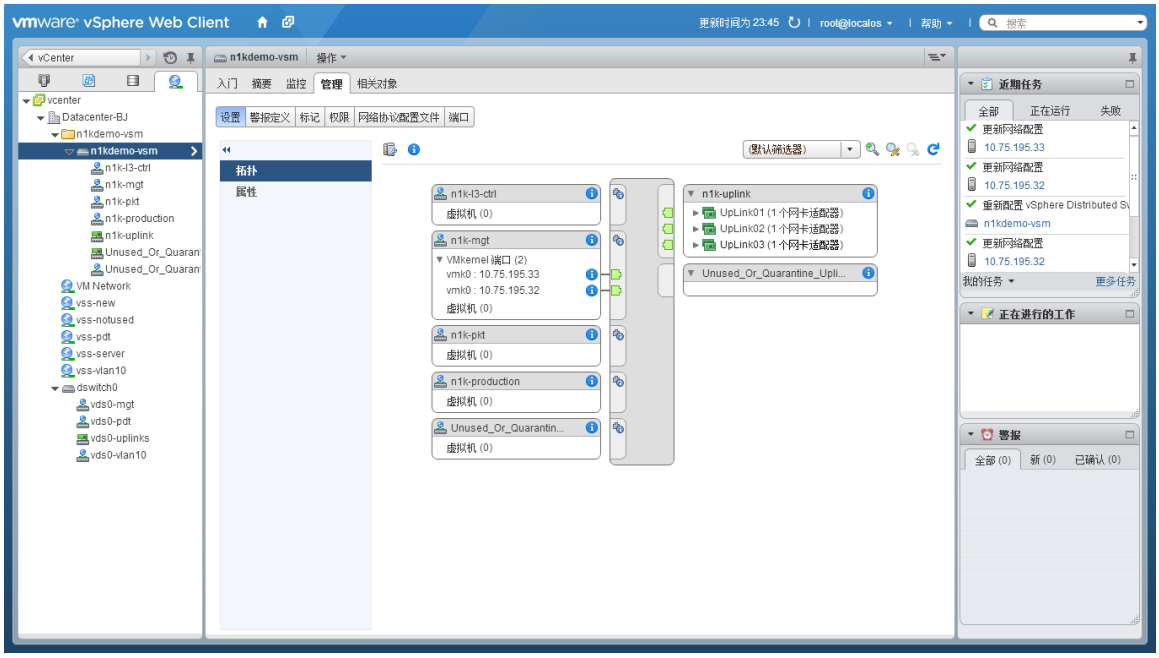

图 8 在 vCenter 中的 Nexus 1000v 拓扑

仔细对比图 4 和图 8，您会发现，前者在 vCenter 中有很多配置选项，而后者仅有迁移选项，原因是涉及网络的配置全都在 VSM 上完成，再通过 API 推送到 vCenter 上。因此，Nexus 1000v 通过 API 完成与 vCenter 的通信，从而使得网络团队与虚拟化/服务器管理团队之间实现可靠、高效、无缝的衔接，同时又保障各自之间的相互独立，互不影响。

这里有一个很有趣的“先有鸡，还是先有蛋？”的问题，即一个新的 VEM 初次安装，需要添加到 Nexus 1000v 中，它未被 VSM 配置，不知道应该如何与 VSM 通信。该问题需要由 System VLAN 和 Opaque Data 来解决。

System VLAN 指在 VEM 在被 VSM 配置完成前就应该允许其通信的 VLAN，拥有 System Vlan 的 Port-profile 叫做“system port-profiles”。通常，Nexus 1000v Control、 Packet、Management VLAN，vSphere 管理接口以及承载 iSCSI 的 vmkernel 接口需要配置为 System VLAN。在上文中的“n1k-uplink”和“n1k-l3-ctrl”这两个 Port-Profile 均配置了 System VLAN。

Opaque data 由 VSM 生成，在 VSM 向 vCenter 连接注册时推送给 vCenter，包含了 VEM 需要初始化上行端口所需要的最小信息，一旦上行端口配置完成，VEM 即可和 VSM 通信完成其他配置。Opaque data 主要包含：域配置(Domain ID, Control VLAN, Packet VLAN)、DVS 名、VSM 版本、System 配置[System VLANS]、Port-profile 名字、VSM IP 地址、VSM MAC 地址等。然后由 vCenter 将 Opaque data 推送到 VEM。以下是分别在 VSM 和 VEM 上看到的信息，两者是通过 Opaque data 实现配置的。

```
n1kdemo-vsm# show svs domain
SVS domain config:
  Domain id:     10
  Control vlan:   NA
  Packet vlan:    NA
  L2/L3 Control mode: L3
```

```
  L3 control interface: control0
  Status: Config push to VC successful.
  Control type multicast: No

VEM 安装完毕后，可通过 vemcmd 查看 VEM 的各种信息，用以辅助管理员进行管理和排障。
该命令在 VSM 上，也可以通过 module vem 4 execute vemcmd show card 执行
~ # vemcmd show card
Card UUID type  2: 6fcb2715-08f2-7d4c-a412-bacc3d2a663e
Card name: CiscoUCSC220
Switch name: n1kdemo-vsm
Switch alias: DvsPortset-3
Switch uuid: 08 a3 29 50 2b b5 8e b0-4d 45 d8 f0 63 9e bb be
Card domain: 10
Card slot: 4
VEM Tunnel Mode: L3 Mode
L3 Ctrl Index: 49
L3 Ctrl VLAN: 20
VEM Control (AIPC) MAC: 00:02:3d:10:0a:03
VEM Packet (Inband) MAC: 00:02:3d:20:0a:03
VEM Control Agent (DPA) MAC: 00:02:3d:40:0a:03
VEM SPAN MAC: 00:02:3d:30:0a:03
Primary VSM MAC : 00:50:56:a9:3b:65
Primary VSM PKT MAC : 00:50:56:a9:3f:80
Primary VSM MGMT MAC : 00:50:56:a9:d7:18
Standby VSM CTRL MAC : ff:ff:ff:ff:ff:ff
Management IPv4 address: 10.75.195.33
Management IPv6 address: 0000:0000:0000:0000:0000:0000:0000:0000
Primary L3 Control IPv4 address: 10.0.20.35

vCenter 中也可直接查看 Opaque Data，在 https://vc ip address/mob 中可以查看，路径为“content” > “rootFolder(group-dxx)” > “datacenter-xxx” > “networkFolder(group-nxx)” > “group-nxx(vsm-name)” > “dvs-xxx(vsmname)” > “config” > “com.cisco.svs.switch.config”
```

在实际操作中，我们通过在 vCenter 中添加主机到 Nexus 1000v 中将 Opaque data 推送到 VEM，此操作与向 DVS 添加和管理主机没有不同。如图 9 所示：

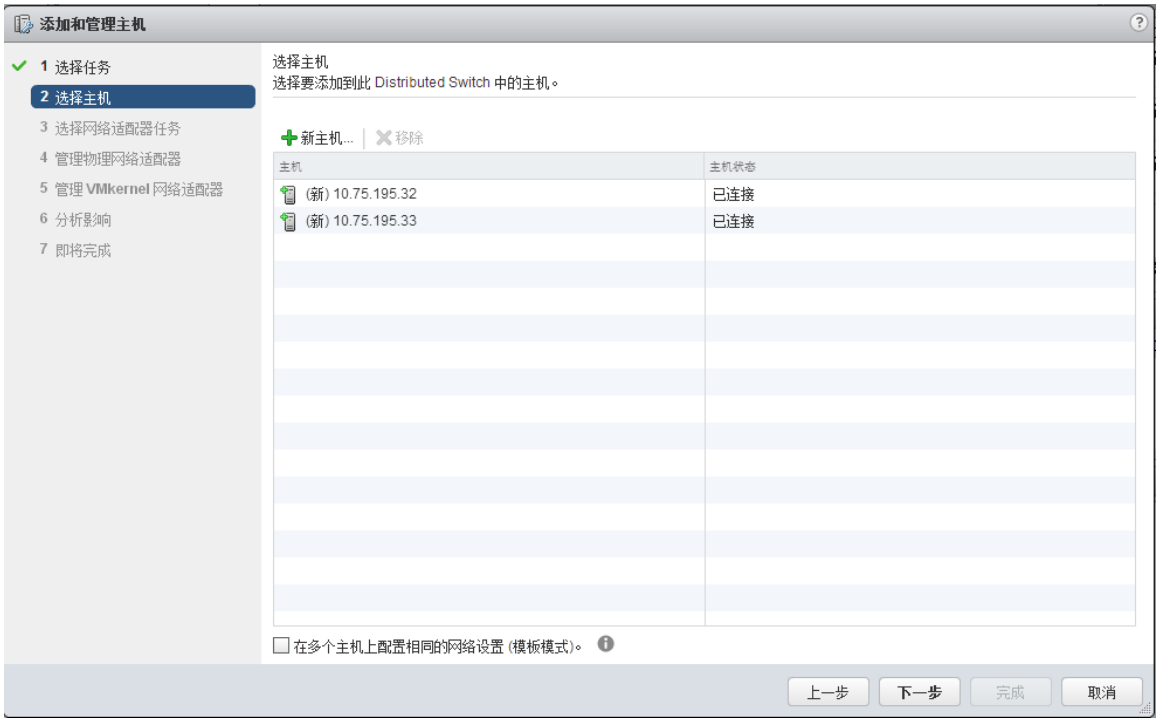

图 9 新增 VEM 到 Nexus 1000v 中

选择“新主机”，分配上联接口，并分配网络标签“n1k-uplink”，将用于 VEM 通信的 vmkernel 接口（使用管理 vmk 或者新增一个 vmk 接口）迁移到“n1k-l3-ctrl”网络标签。一步步操作完毕后，VEM 收到 Opaque data，进行初始化，与 VSM 取得联系。VSM 发现 VEM，对其完成其他配置，作为新的板卡安装到 Nexus 1000v 中。

在此过程中，VSM 会产生如下日志：

```
%VEM_MGR-2-VEM_MGR_DETECTED: Host CiscoUCSC220 detected as module 4
%VEM_MGR-2-MOD_ONLINE: Module 4 is online
%VIM-5-IF_ATTACHED: Interface Ethernet4/1 is attached to vmnic0 on module 4
%VIM-5-IF_ATTACHED: Interface Ethernet4/2 is attached to vmnic1 on module 4
```

同时，Nexus 1000v 的 running-config 中会自动添加以下信息：

```
interface Ethernet4/1
  inherit port-profile n1k-uplink
interface Ethernet4/2
  inherit port-profile n1k-uplink
```

Nexus 1000v 就绪后，虚拟化/服务器管理团队可将虚拟机向 Nexus 1000v 进行迁移，此操作与向 DVS 迁移没有不同。可以在网络清单中使用迁移向导（Wizard）进行迁移，也可以编辑虚拟机的网络标签进行迁移。图 10 展示了通过迁移向导向 Nexus 1000v 迁移的一个步骤：

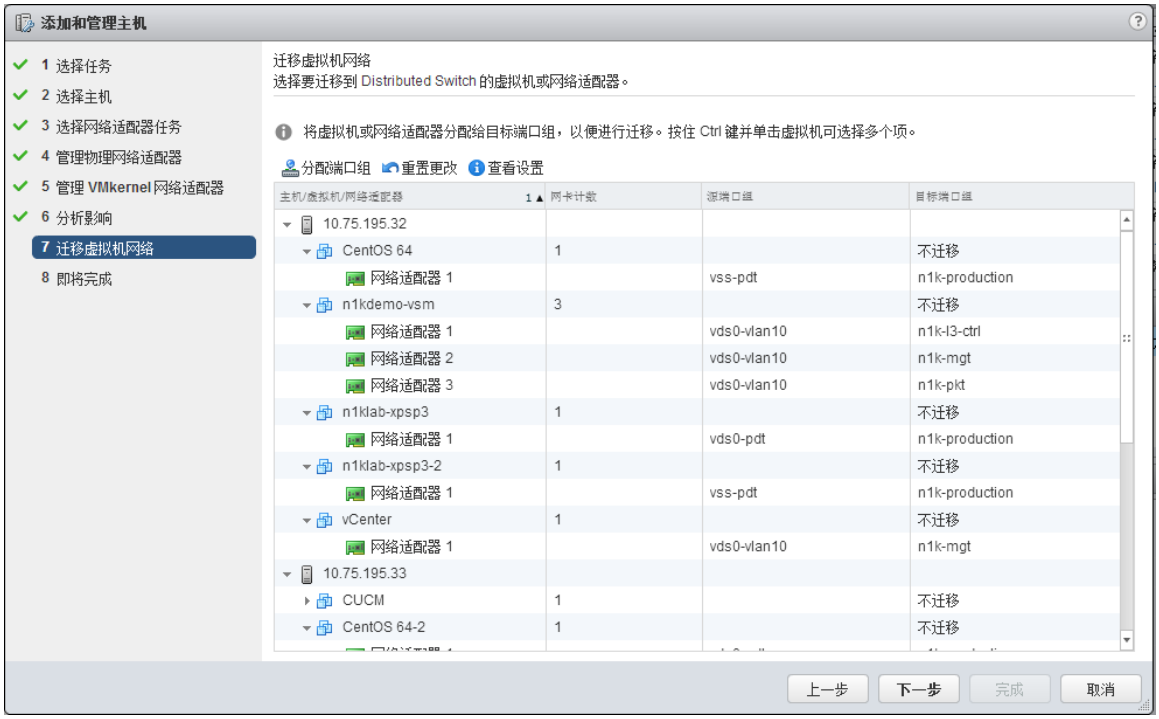

图 10 通过迁移向导将虚拟机迁移到 Nexus 1000v

在虚拟化/服务器管理团队将虚拟机迁移到 Nexus 1000v 后，VSM 上也有相应的日志出现，提醒网络管理员：

```
n1kdemo-vsm %VIM-5-IF_ATTACHED: Interface Vethernet5 is attached to Network Adapter 1
of CentOS 64-3 on port 3 of module 4 with dvport id 256
n1kdemo-vsm %ETHPORT-5-IF_UP: Interface Vethernet5 is up in mode access
//日志解读：CentOS 64-3 这个虚拟机的第一个网卡连接到了 VEM 4 的第 3 个端口，dvport id 为 256，dvport
是 vCenter 分配的，是全局的。VSM 自动生成一个 Interface Vethernet5 与其相对应。
```

Running-config 会自动生成以下 vethernet 接口配置，通常情况下，网络管理员不直接配置该 vethernet 接口，但是在必要时也可以对该 vethernet 进行 shutdown 操作。

```
interface Vethernet5
  inherit port-profile n1k-production
  description CentOS 64-3, Network Adapter 1
  vmware dvport 256 dvswitch uuid "08 a3 29 50 2b b5 8e b0-4d 45 d8 f0 63 9e bb be"
  vmware vm mac 0050.568F.4410
```

dvswitch uuid 是 Nexus 1000v 在 vCenter 中的唯一标识号，dvport id 256、vethernet 5 接口，MAC 绑定关系在虚拟机进行 vmotion 过程均中保持不变，因此可以实现策略跟随，并且，端口上的统计数值也可以跟随。虚机进行 vMotion 迁移，vethernet 改变的仅仅是其 VEM 的槽位号（代表物理服务器）。

```
vMotion 之前：
n1kdemo-vsm# show int vethernet 19
Vethernet19 is up
  Port description is n1klab-xpsp3-2, Network Adapter 1
  Hardware: Virtual, address: 000c.292d.4272 (bia 000c.292d.4272)
  Owner is VM "n1klab-xpsp3-2", adapter is Network Adapter 1
  Active on module 3
  VMware DVS port 129
  Port-Profile is n1k-production
  ……
vMotion 以后：
n1kdemo-vsm# show int vethernet 19
Vethernet19 is up
  Port description is n1klab-xpsp3-2, Network Adapter 1
  Hardware: Virtual, address: 000c.292d.4272 (bia 000c.292d.4272)
  Owner is VM "n1klab-xpsp3-2", adapter is Network Adapter 1
  Active on module 4
  VMware DVS port 129
  Port-Profile is n1k-production
  ……

n1kdemo-vsm# show int virtual

--------------------------------------------------------------------------------
Port        Adapter        Owner                     Mod Host
--------------------------------------------------------------------------------
Veth1       vmk1           VMware VMkernel           3   DELLPET310
Veth2       vmk1           VMware VMkernel           4   CiscoUCSC220
Veth3       vmk0           VMware VMkernel           3   DELLPET310
Veth4       vmk0           VMware VMkernel           4   CiscoUCSC220
Veth5       Net Adapter 1  CentOS 64-3               4   CiscoUCSC220
Veth6       Net Adapter 1  vcenter                   4   CiscoUCSC220
Veth7       Net Adapter 1  CentOS 64-2               4   CiscoUCSC220
Veth8       Net Adapter 1  n1kdemo-vsm-1             3   DELLPET310
Veth9       Net Adapter 1  n1kdemo-vsm-2             3   DELLPET310
Veth10      Net Adapter 1  n1klab-xpsp3              3   DELLPET310
Veth11      Net Adapter 3  n1kdemo-vsm-1             3   DELLPET310
Veth12      Net Adapter 3  n1kdemo-vsm-2             3   DELLPET310
Veth13      Net Adapter 2  n1kdemo-vsm-1             3   DELLPET310
Veth14      Net Adapter 2  n1kdemo-vsm-2             3   DELLPET310
```

当服务器管理员将虚拟机迁移到 Nexus 1000v 后，物理及虚拟网络的拓扑如下，笔者在测试中，将 vSphere 所有的 vmnic、vmk 以及虚拟机 vNIC 全部链接在 VEM 上。迁移过程中，要注意先后顺序，确保 vCenter 与 vSphere 主机保持连接，请迁移完虚拟机后，再将 vSwitch0 的上连接口迁移到 Nexus 1000v。

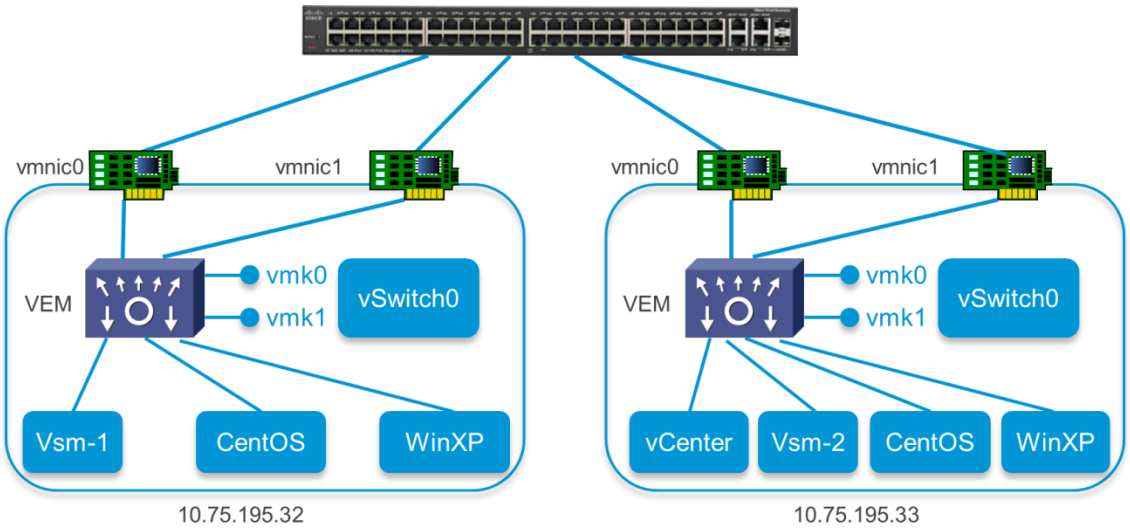

图 11 实验环境中迁移完成后的拓扑图

Nexus 1000v 还提供一个 vTracker 工具，帮助提供虚拟网络环境信息， vTracker 从 vCenter、CDP 收集相关信息，是一个很好的跟踪、排错工具。该功能通过 feature vtracker 命令开启：

网络管理员可以通过 vTracker 获取虚拟机的信息，如：虚拟名字、操作系统、状态、CPU、内存分配及利用率，跟踪 vMotion 的记录等等，让网络管理员不再对底层的虚拟机部署情况一无所知。此工具对虚拟化/服务器管理团队是透明的，他们不会增加任何工作量。

```
n1kdemo-vsm# show vtracker vm-view info
Module 4:
   VM Name:                CentOS 64-2
   Guest Os:               CentOS 4/5/6 (64-bit)
   Power State:            Powered On
   VM Uuid:                420fb7c2-32ad-ff7f-a433-adc3d6bc5331
   Virtual CPU Allocated:  2
   CPU Usage:              0 %
   Memory Allocated:       2048 MB
   Memory Usage:           0 %
   VM FT State:            Unknown
   Tools Running status:   Running
   Tools Version status:   current
   Data Store:             datastore1 (1)
   VM Uptime:              8 days 23 hours 30 minutes 43 seconds

Show vtracker vm-view vnic 可以查看每一个虚拟机有关网卡的详细信息，如 IP 地址、Mac 地址、虚机名称、
VLAN 等。

n1kdemo-vsm# show vtracker vm-view vnic
* Network: For Access interface - Access vlan, Trunk interface - Native vlan,
           VXLAN interface - Segment Id.
--------------------------------------------------------------------------------
Mod VM-Name          VethPort     Drv Type  Mac-Addr        State Network Pinning
    HypvPort         Adapter      Mode      IP-Addr
--------------------------------------------------------------------------------
3   n1klab-xpsp3     Veth10       Vmxnet3   0050.568f.07d8  up    50      Eth3/1
    260              Adapter 1    access    10.0.50.98
4   CentOS 64-2      Veth7        Vmxnet3   0050.568f.0208  up    50      Eth4/2
    257              Adapter 1    access    10.0.50.30
4   CentOS 64-3      Veth5        Vmxnet3   0050.568f.4410  up    50      Eth4/2
    256              Adapter 1    access    10.0.50.35
4   vcenter          Veth13       Vmxnet3   000c.2952.d552  up    10      Eth4/2
    68               Adapter 1    access    10.75.195.31
```

最后，我们以图 12 来示意网络、服务器团队之间的工作流，进行总结：

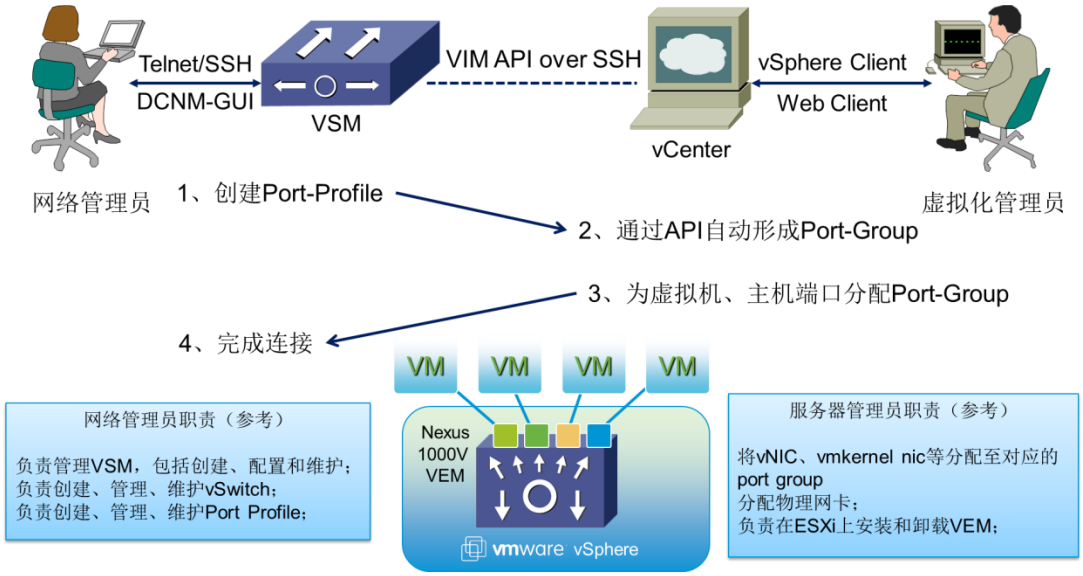

图 12 Nexus 1000v 的运作模式图示

结语：Cisco Nexus 1000v ——不仅仅是一个虚拟交换机

Cisco Nexus 1000v 起始于 Cisco 和 VMware 的紧密合作，至今已有 5 年的历史。Nexus 1000v 不断创新，已不仅仅是一个虚拟交换机，Cisco 将其定位为云网络平台的基础。Nexus 1000v 目前已扩展至其他虚拟化平台，支持 Microsoft Hyper-V，并即将支持 Linux KVM，均提供相同的 NX-OS 特性和管理体验。同时，为满足多种云之间的互联需求，Nexus 1000v 也有 Inter-Cloud 版本，可以提供企业数据中心与公有云之间高度安全的二层连接，并保持同样的策略。Nexus 1000v 也率先支持 VXLAN，在传统 VLAN 上进行了扩展，支持数以百万计的逻辑网络，能够让工作负载在多个数据中心和云基础架构间迁移。

利用 vPath（Virtual Service Data Path），Nexus 1000v 还支持服务链（Service Chaining）功能，可在报文数据路径中插入各种虚拟服务，包括 VSG（Virtual Security Gateway）提供基于分区的安全服务，第三方 Citrix NetScaler 1000V 提供虚拟负载均衡服务，ASA 1000v 提供云防火墙安全服务。vPath 是 Nexus 1000v VEM 内嵌的智能模块。vPath 将原始二层报文进行封装，提供 VEM 到 VSN（Virtual Service Node）之间的隧道，是一种 Overlay 技术。vPath 隧道支持 VLAN 模式、IP 模式和 VXLAN 模式，使得 VSN 可以部署在网络中的任意位置。vPath 的转发优先于正常的二层交换，并且支持可编程，可使 VEM 直接对后续流量进行重定向、丢弃或允许等操作，解决流量集中到 VSN 上处理的性能和可扩展性问题。

看到这里，您会发现 Nexus 1000v 实际上与当前炙手可热的软件定义的网络（SDN）有很多共同点：

- 控制平面（VSM）和数据平面（VEM）分离；

- 控制逻辑集中，集中在 VSM 上控制；
- 支持开放接口（Nexus 1000v 支持 Restful API）。

在了解了 Nexus 1000v 的架构和原理后，相信您会更容易理解虚拟网络解决方案。实际上，业内基于硬件的解决方案，如 802.1BR 和 802.1Qbg，同样也需要在 Hypervisor 或者 Hypervisor 的操作系统上安装 Agent 程序，但是这些 Agent 程序并不提供标准交换功能，其作用是将虚拟机的报文交给上层物理交换机处理。SDN 如果要部署到虚拟网络，也需要安装 Agent，例如 Openflow Agent。

Nexus 1000v 是成熟并且完全可落地的解决方案，同时，Nexus 1000v 还在不断演进，在 Cisco 最新发布的 ACI（Application Centric Infrastructure）架构中，同样能看到 Nexus 1000v 的影子，其名称改为 AVS（Application Virtual Switch）。

参考资料

1、Virtual Machine Networking: Standards and Solutions

http://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps9902/whitepaper_c11-620065.html

2、Cisco Press《Data Center Virtualization Fundamentals》Gustavo Alessandro Andrade Santana

http://www.ciscopress.com/store/data-center-virtualization-fundamentals-understanding-9781587143243

3、Cisco Nexus 1000V Series Switches Deployment Guide Version 3

http://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps9902/guide_c07-556626.html