# Virtual Switching

## Best Practice Document

Produced by the RENATER-led working group on Virtual Switching

Authors: C. Foll (Université Lille 3/GIP RENATER), A. Méré (Université Paris Sud/GIP RENATER), J-P. Feuillerat (DSI CNRS/GIP RENATER)

December 2013

| | |
|---|---|
| Document No: | GN3plus-NA3-T2-R3.4 |
| Version / date: | V1.0, 2 December 2013 |
| Original language : | French |
| Original title: | “Commutation virtuelle” |
| Original version / date: | V1.0, 2 December 2013 |
| Contact: | cbp@listes.renater.fr |

RENATER bears responsibility for the content of this document. The work has been carried out by a RENATER led working group on virtual switching as part of a joint-venture project within the HE sector in France.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 605243, relating to the project 'Multi-Gigabit European Research and Education Network and Associated Services (GN3plus)'.

# Table of Contents

# Table of Figures

# Executive Summary

The increase in the use of virtual machines in our campuses has forced the network to adapt. It must meet the same standards for access to the virtual machines as those required for access to physical servers: security, reliability, performance.

Virtual switching has evolved to the point where today it offers a very varied range of solutions meeting the needs of system and network administrators. It even goes beyond what is offered by physical switching (port hot-swapping with all security policies, high port density and scalability, etc.).

This document discusses the problems connected with moving from physical switching to virtual switching in both technical and organisations terms. It presents a range of available solutions and the advantages/inconveniences associated with each.

An implementation solution is also presented.

# 1 What is a virtual switch?

## 1.1 History

From the year 2000, virtualisation has progressively established itself, starting with the workstation, followed by the development infrastructures, before coming into widespread production use in many campuses.

From a few machines running on a physical server in the same IP addressing plan, we have moved to dozens or even hundreds of virtual machines belonging to multiple VLANs, and having requirements in terms of security and quality of service equivalent to physical machines.

## 1.2 Role and interest of virtual switches

The purpose of virtual switches is to ensure Layer 2 connectivity between different virtual machines.

These can be limited to traffic exchanges within the virtual infrastructure. In this case, the “physical” switching infrastructure does not handle the traffic, which is completely invisible to them.

They can also send traffic to the physical infrastructure. In this case, it may involve the need to perform a routing operation or use of the 802.1Qbg or 802.1Qbh standards, as discussed in the remainder of the document.

Technically, an aggregated VLAN is connected to servers hosting the hypervisors; the technical teams responsible for the virtualisation infrastructure create virtual switches associated with different VLANs, before attaching the virtual machines to them.

Virtualisation offers great flexibility and simplicity for carrying out certain tasks for which the network teams are responsible, such as the migration of servers and the associated integration tasks.

It reduces costs by limiting the number of physical switches. This limitation also simplifies the network architectures, making them easier to administer.

This growth of “virtual switching” is thus understandable. Nonetheless, it brings other problems in its wake.

## 1.3 Problems caused by virtual switching

### 1.3.1 Organisational

At the beginning, management of the virtual switch infrastructure is often carried out by the system teams who have control of the hypervisor. This can create problems if the organisation has not been thought through in advance. Indeed:

- The network team loses visibility and its prerogatives over part of the network.
- The network team can no longer audit and verify the consistency of all the active elements of the architecture.
- Part of the network security management is transferred to the system team which may not have experience in it.
- The network team can feel dispossessed of some of its work. This can generate conflicts. This problem should not be ignored as, in addition to the bad atmosphere it creates, it can result in a breakdown of responsibilities and, ultimately, difficulties in resolving a failure.

### 1.3.2 Functional

The active equipment has lots of functionality that cannot be found in what is offered by default in the virtualisation infrastructures:

- Monitoring:

  In general, functions such as sFlow/NetFlow or SNMP are not available on the virtual switches supplied by default with the hypervisors. This makes global monitoring of the switching infrastructure impossible. Part of the network therefore cannot be monitored by the teams responsible for the monitoring.

- Traffic analysis:

  RSPAN-type functionality, which allows analysis of all traffic passing through a port or a VLAN, is also often missing, making it more difficult to use a complex IDS to analyse traffic between virtual machines, for example.

- QoS:

  Quality of service functions are not generally implemented on the standard virtual switches. This can create significant problems for virtualised hosting of some services, such as those related to ToIP or videoconferencing.

- Security functions:

  Protection mechanisms against Address Resolution Protocol (ARP) cache poisoning or DHCP snooping attacks currently installed on switches are not always present on virtualised architectures. This is also the case for private VLANs that allow traffic between machines on the same VLAN to be blocked, or access lists, which are often not available on basic virtual switches.

### 1.3.3 Security

The security problems caused by virtual switching follow from the two previous points.

**Organisational**

If the separation of tasks between the teams has not been properly planned and decided, security can prove to be a delicate issue in the context of virtualisation. Side effects on security include operational error of the system team creating a problem for the network team, and a team not trained in networking having to manage active elements.

Among the many risks produced are:

- The creation of a virtual machine connected to several VLANs by mistake, allowing the security policy to be bypassed.
- A denial of service at the network level rendering the virtualisation infrastructure inoperative as a consequence of bad configuration.
- The absence of rules for QoS or filtering.

Errors of this type are not specific to virtualised environments, but the risk is increased if the team tasked with configuration of the virtual switches is only trained in system administration.

**Functional**

As we have seen, many security functions are missing from the “basic” switches offered by the virtualisation infrastructure. Consequently, attackers can carry out attacks more easily and with greater impact than if they had gained control of a physical machine.

It should be noted, however, that as the bond between virtual machines and virtual switches is particularly strong, some security functions are only possible in the virtual world, in particular, the ability to block passage for network cards in “promiscuous” mode or sharing-related functions and the limitation of I/O networks between virtual machines. As we will see in what follows, these deficiencies can be overcome by subscribing to an additional licence, particularly QoS functions and those related to protection against network attacks.

### 1.3.4 Troubleshooting

Since some functionality is generally missing (e.g. monitoring, traffic analysis), the detection and analysis of problems on the network can prove particularly difficult. All the more so since, as the virtual machines and switches often share the same equipment (for the hypervisor), the analysis and resolution of performance problems can be rendered more complex. This is particularly the case when it comes to isolating what is inherent to problems associated with the system and problems associated with the network. Even worse, a major system change can degrade network performance or vice versa.

# 2 The different solutions

## 2.1 Logical solutions

### 2.1.1 Bridges managed at the system level

Use of the “bridge network” function integrated into the system was the first solution usually chosen to allow virtual machines access to the network while using their own IP. It is indeed the simplest operating method and the easiest to set up.

When the hypervisor is a Linux server, the use of a specific package is required (with Debian this is “bridge-utils”). It allows the creation of a bridge interface “brX” linking a physical interface “ethx” with a logical interface “tapX” to which a virtual machine will be connected.

The initial versions of Xen used this mechanism and it is still used by Kernel-based Virtual machine (KVM). This operating method allows Layer 2 connectivity to be achieved between one or more virtual machines and a physical network and remains fully appropriate for use limited to a few machines or a test architecture. Its use in a more extended architecture (large number of VMs, VLANs, etc.) is not advisable because the management and supervision of many bridges is complex.

### 2.1.2 Virtual Ethernet Bridges (VEB)

These take the form of a software package linked to the hypervisor (without additional cost) offering the functionality of an entry-level, physical switch. There is no use of the system switching functions of the host server as in the previous case, but switching management is offered at the hypervisor level.

If virtual machines connected to a VEB need to communicate amongst themselves, the exchange remains internal to the hypervisor’s VEB. If the destination is external, the exchange will then pass through the server’s physical interface.

Although more advanced than the bridge systems, VEBs do not include all the functions expected by a switch, particularly in terms of monitoring, and management of service quality or security. They require management methods that are unique to themselves and can often only use a single type of hypervisor.

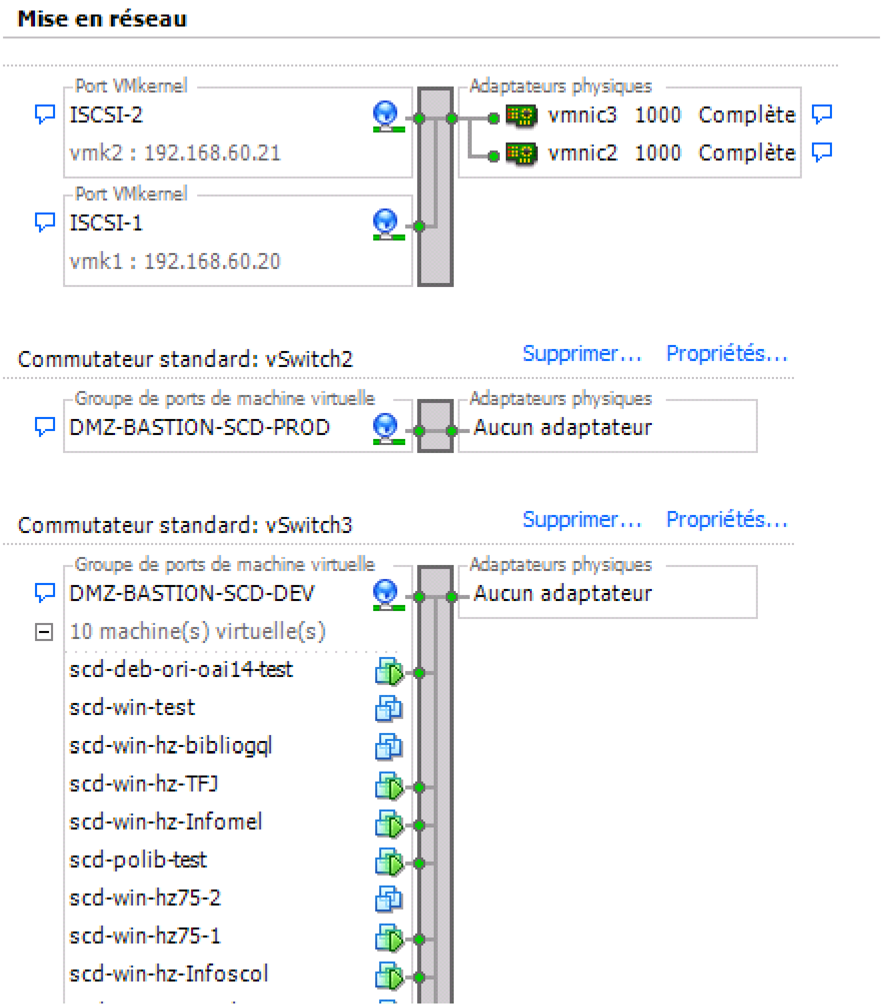
This type of switch is also often administered by the team responsible for the hypervisor, resulting in a loss of visibility for the network team.

When the number of virtual machines is limited and few advanced Layer 2 functions are required, this solution is without doubt a good one. It allows rapid communication between machines on the same VEB.

This type of solution will nonetheless be difficult to set up in architectures with a large number of physical servers which will require a distributed architecture instead. The configuration of the port groups will have to be replicated manually on each virtual switch so that a VM can find its virtual port configuration when moved from one server to another.

### Example with VMware vSphere

When a standard vSphere licence is purchased, the user will have a VEB named Vswitch that offers a set of basic functions (Layer 2 forwarding, VLANs, etc.).



Mise en réseau

Port VMkernel
ISCSI-2
vmk2 : 192.168.60.21

Adaptateurs physiques
vmnic3 1000 Complète
vmnic2 1000 Complète

Port VMkernel
ISCSI-1
vmk1 : 192.168.60.20

Commutateur standard: vSwitch2 Supprimer... Propriétés...

Groupe de ports de machine virtuelle
DMZ-BASTION-SCD-PROD

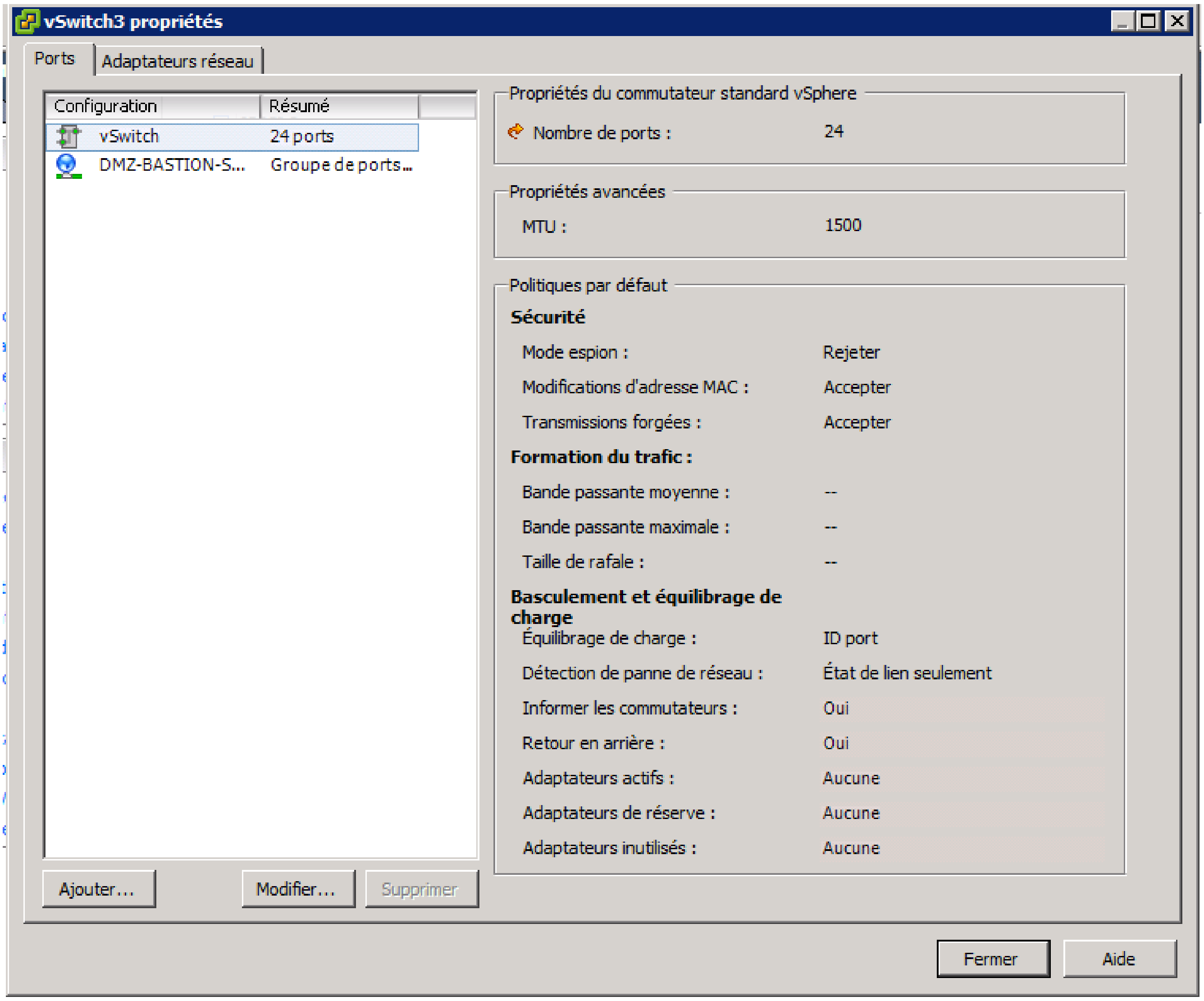
Adaptateurs physiques
Aucun adaptateur

Commutateur standard: vSwitch3 Supprimer... Propriétés...

Groupe de ports de machine virtuelle
DMZ-BASTION-SCD-DEV
10 machine(s) virtuelle(s)
scd-deb-ori-oai14-test
scd-win-test
scd-win-hz-bibliogql
scd-win-hz-TFJ
scd-win-hz-Infomel
scd-polib-test
scd-win-hz75-2
scd-win-hz75-1
scd-win-hz-Infoscol

Adaptateurs physiques
Aucun adaptateur

Figure 2.1: Connection of the virtual machines to the vSwitches (one vSwitch being created per VLAN)



| Configuration | Résumé |
|---|---|
| vSwitch | 24 ports |
| DMZ-BASTION-S... | Groupe de ports... |

Propriétés du commutateur standard vSphere

Nombre de ports : 24

Propriétés avancées

MTU : 1500

Politiques par défaut

**Sécurité**

Mode espion : Rejeter

Modifications d'adresse MAC : Accepter

Transmissions forgées : Accepter

**Formation du trafic :**

Bande passante moyenne : --

Bande passante maximale : --

Taille de rafale : --

**Basculement et équilibrage de charge**

Équilibrage de charge : ID port

Détection de panne de réseau : État de lien seulement

Informer les commutateurs : Oui

Retour en arrière : Oui

Adaptateurs actifs : Aucune

Adaptateurs de réserve : Aucune

Adaptateurs inutilisés : Aucune

Figure 2.2: Options available for each vSwitch

As can be seen in Figure 2.2, the options are basic, and more appropriate for entry-level switches for workstations rather than equipment installed for server access in a data centre.

### 2.1.3 Virtual Distributed Switches (VDS)

“Virtual Distributed Switches” (VDS) make use of the concept of the Virtual Ethernet Bridge, i.e. it is a software switch implemented by the hypervisor, but with considerably extended functionality. The main difference with VEBs is the possibility of deploying complex network configurations on multiple physical servers.

All of the distributed virtual switches are brought together in a single “virtual chassis” managed by a supervisor, where in reality each module corresponds to a host server. From then on, all the configuration is centralised in a single location. This allows the network team to simply define the port profiles, which will be moved automatically to all the virtual switches and made available to the system team, which will be able to assign them to the virtual machines.

This separation between the “network” and “system” management is also an advantage of some VDSs. The network administrator manages the switch as if it was a Layer 2 physical switch, while the system administrator simply assigns a usage profile defined by the network team to a virtual machine.

Both types of administrator will find the functionality of a traditional Layer 2 physical switch (port monitoring, quality of service, security – private VLAN, DHCP snooping, ARP inspection...), both for the network  interfaces of the virtual machines and the physical interfaces of the host servers, and will be able to create port profiles tailored for each type of virtual machine.

An additional advantage is that any shift of the VM to another host server will be completely transparent at the network level.

This model is therefore ideal when there are different system and network teams, each of which wants to control their own sphere, where there are significant problems related to monitoring and security, and once there are several host servers each serving a large number of virtual machines.

**Example with VMware vSphere**

With vSphere, the most expensive vSphere licence, the “Enterprise Plus”, has to be acquired to obtain advanced switching functionality. The retail price of this licence is almost 2.5 times more than the standard version [vSphere_pricing]

This licence also opens up the possibility of using third-party software switches (Cisco Nexus 1000v, IBM 5000V, etc.) or the “advanced” VMware switches called “Distributed Switch” with the following functionality:

- Monitoring (SNMPv3 and NetFlow).
- Traffic analysis with RSPAN.
- Isolation of the virtual machines with PVLAN.

## 2.2 Mixed solutions (physical/logical)

The solution that involves having the VMs communicate via physical elements is being promoted by hardware manufacturers. They argue that this type of solution frees up the server from handling the processing associated with the virtual switch operation, that no virtual switch has all of the functionality of a physical switch and that it is simpler to integrate with other equipment (such as Intrusion-Prevention Systems and firewalls).

However, we have not come across feedback related to this type of solution being used in the teaching/research community, nor have we been able to test it.

### 2.2.1 Virtual Ethernet Ports Aggregators

Virtual Ethernet Ports Aggregators (VEPA) adopt the 802.1 Qbg (Edge Virtual Bridging) standard and are offered by HP infrastructure in particular.

The aim of these connectors is to emulate as closely as possible the normal functioning of the physical network, allowing the virtual machines to be managed as physical servers. Packet-switching between two physical servers is performed by physical switches, so the VEPA module forces the traffic coming from the VMs to go from the physical interface of the server hosting the hypervisor to the physical switch.

The advantages of this type of operation are apparent:

- Setting up virtual machines does not involve any change in the work of the network team.
- The monitoring and administration equipment is the same.
- The organisation does not have to be modified, given that the boundaries between the system and network teams do not change.

The CPU of the server is no longer used to switch traffic and apply rules to the traffic flow (for ACL, QoS, etc.)

The use of VEPA can thus seem to be the solution for existing infrastructure based on HP. It nonetheless requires that the physical equipment supports this type of operation. For that it has to accept sending traffic coming from a physical port to this same port (802.1br standard). Often that only involves a system update.

In addition, if you want to separate the traffic coming from each VM, the switches and network cards must accept the QinQ. (802.1ad). The switches and network cards also need to be powerful and fast to match the bandwidth and latency performance that a VEB could achieve between virtual machines hosted on the same server.

One possible implementation of this type of solution is to use a VEPA-compatible virtual switch such as the HP5900v, integrating it into a VSphere hypervisor in the same manner as a distributed switch.

### 2.2.2 VN-Tags

The VN-Tag solutions are based on the 802.1Qbh standard and are offered by Cisco.

The objective is the same as for VEPA: to send the traffic back to the physical switches. A tag is added to the traffic coming from each VM so that it can be identified by the switch. This allows its ports to be configured as if the virtual machine was directly connected to it.

The problem is that this solution, although potentially a future standard, requires both Nexus switches from Cisco and a Cisco UCS server infrastructure.

### 2.2.3 “Single Root I/O Virtualization” compatible network cards

Another solution is the use of SR-IOV-compatible cards. These cards present themselves at the hypervisor or system level as a group of different instances of this same card (the number depending on the card). This type of solution allows the processor load of a VEB to be shifted to dedicated hardware equipment.

SR-IOV can be used with VSphere 5.1, for example. Nonetheless this method of implementation does not allow the use of many advanced functions (such as vmotion, vshield, netflow).

It has also been put forward by Microsoft for Hyper-V under Windows server 2012 and can be used with XEN and KVM.

Examples of SR-IOV compatible cards are given in [SR-IOV_en] and [SR-IOV-fr].

# 3 Choosing the solution

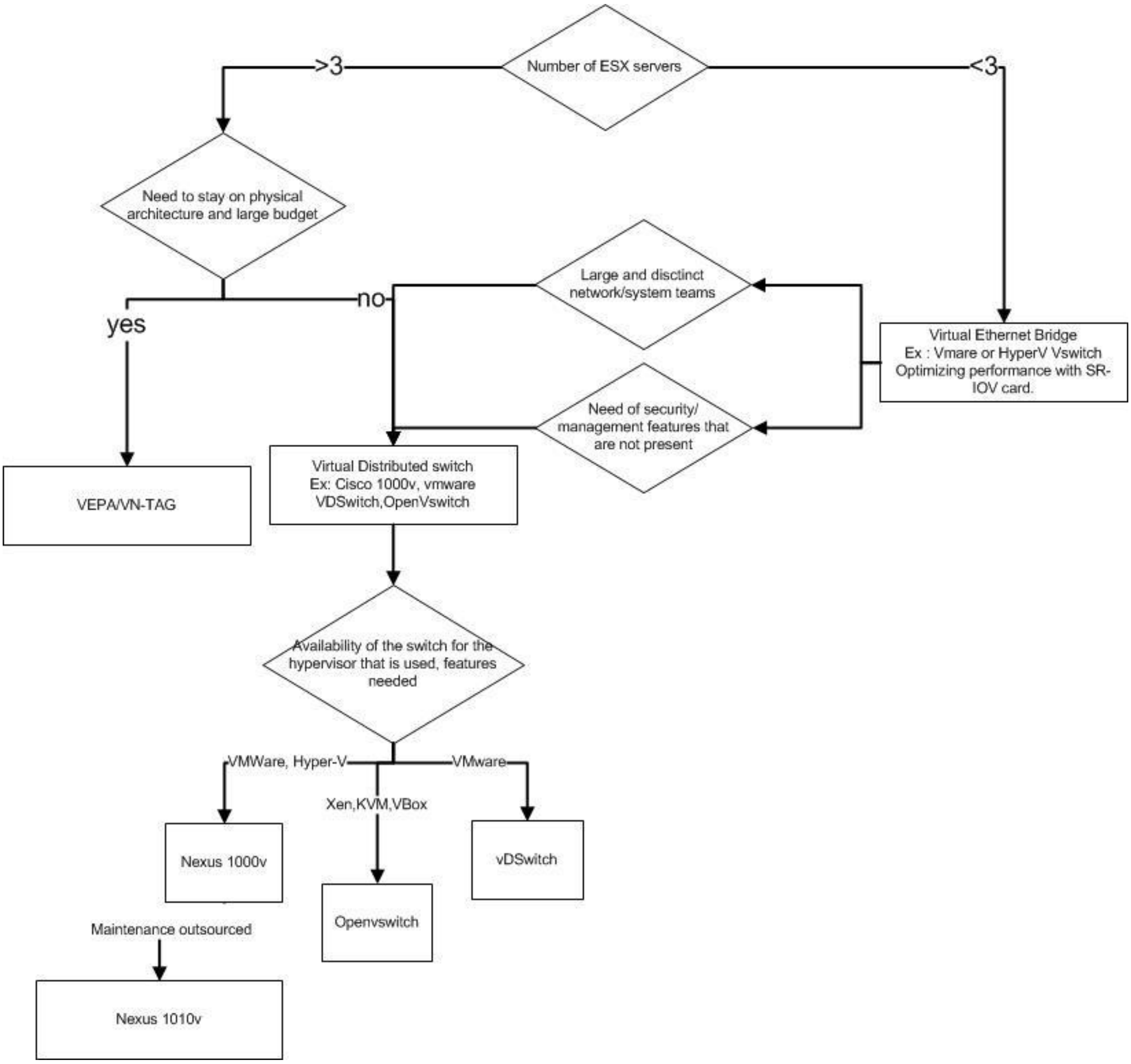

Number of ESX servers
>3
<3
Need to stay on physical architecture and large budget
yes
no
Large and disctinct network/system teams
Virtual Ethernet Bridge
Ex : Vmare or HyperV Vswitch
Optimizing performance with SR-IOV card.
Need of security/ management features that are not present
VEPA/VN-TAG
Virtual Distributed switch
Ex: Cisco 1000v, vmware VDSwitch,OpenVswitch
Availability of the switch for the hypervisor that is used, features needed
VMWare, Hyper-V
Xen,KVM,VBox
VMware
Nexus 1000v
Openvswitch
vDSwitch
Maintenance outsourced
Nexus 1010v

Figure 3.1: Decision flow chart for choosing a virtual switching solution

# 4 Feedback

## 4.1 Context

The solution presented has been set up at the Direction des systèmes d'information (DSI) of the Centre National de la Recherche Scientifique (CNRS) to meet several needs.

First of all an increase in the number of projects requiring ever more servers and thus a greater expenditure (on purchasing, hosting), together with a shortage of staff to cover the administration and monitoring of these additional items of equipment.

The use of virtualisation appeared to be the best solution for dealing with this problem, although initially the CNRS's use of virtualisation was limited to a few dozen virtual machines.

The CNRS were using the hypervisor VMware which was administered by the system team who had set up the VEB, integrated into the solution (vswitch). An 802.1q link was configured on the physical switch so the system team could allocate the VLAN of their choice to the virtual machines.

This mode of operation reached its limits with the rapid increase in the number of virtual machines. In particular, it became necessary to pass on the security rules applied to a virtual machine when it moves around, without cutting off the user traffic.

One solution could have been to move towards the Virtual Distributed Switch from VMWare but the network team already had a Cisco skillset and had mastered the administration tools from this manufacturer. The possibility of putting Access Control Lists (ACLs) on Cisco virtual switches and some QoS functions were also reasons behind the choice.

## 4.2 Architecture

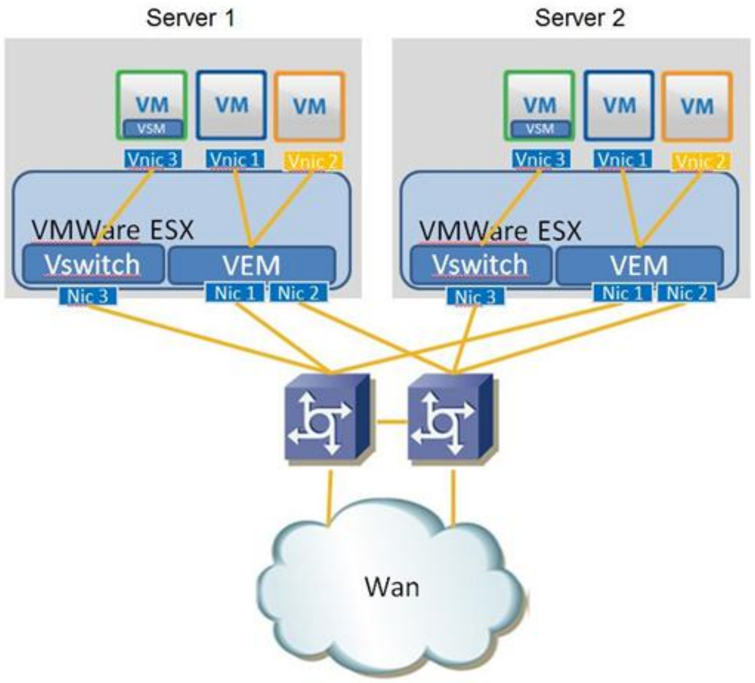


Figure 4.1: Distributed virtual switch architecture

The architecture put in place thus uses a distributed virtual switch. A program linked to the hypervisor is installed on each VMware ESX product. It lets the interfaces of the virtual machines communicate with the physical interfaces of the server. This program is called Virtual Ethernet Module (VEM).

Another program permits the monitoring of all these VEMs distributed across different ESX servers. This is the supervisor (VSM - Virtual Supervisor Module) and it is this module that hosts all of the configuration.

The term “Nexus 1000v” thus refers to the totality of this (or these) VSM(s) and all of the VEM modules distributed across each of the ESX servers.

Updating it will thus require updating the VSM at the same time (corresponding to updating network equipment), with an update of the VEMs on each ESX which corresponds to adding a module in a VMware system.

The supervisor VSM that monitors all of the VEMs can either run on a virtual machine (i.e. an entirely virtualised infrastructure), or be moved to one or two physical devices (active/passive): the Nexus 1010 or 1110. These can also handle other applications (network analysis, distributed software firewall, etc.) directly linked to the Nexus 1000v. A physical appliance option, in particular, prevents any impact on the distributed switch when there is a disruption or incident on the server hosting the VSM.

In the solution presented, this type of equipment is not available, so the two supervisors in active /passive are therefore present in the form of two virtual machines distributed across two ESXs. With four physical network cards on each ESX, it was decided to dedicate one of them for access to the VSM. Using this hybrid mode

(VEB, VDS) the VSMs were placed behind the VEMs that they manage, a potentially poor configuration that could render them inaccessible. The risk is limited because a loss of the VSMs does not necessarily involve cutting-off the traffic passing through the VEMs, but it was thought this mode of operation was more appropriate.

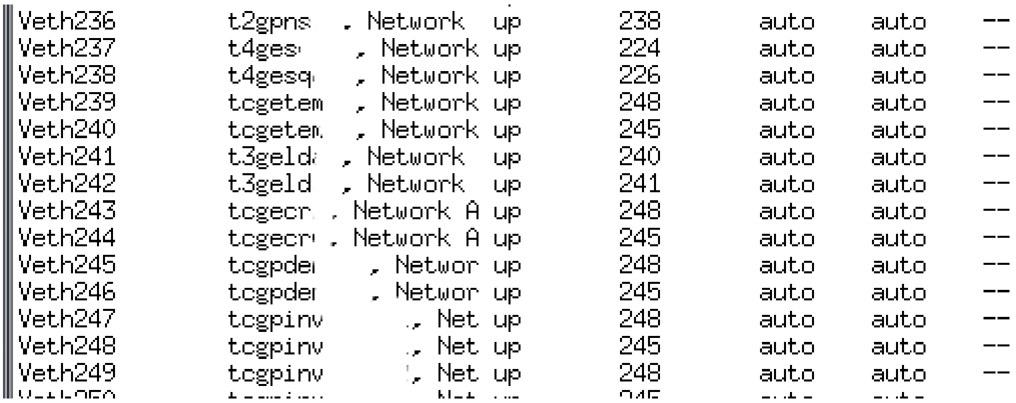
## 4.3 Use

Using a Nexus virtual switch is equivalent to using a traditional switch. Configuration of the interfaces is carried out via the port profiles.

Each port profile brings together a set of information shared by all of the ports that make it up. This information could include VLANs, PVLANs, QoSs, ACLs, etc.

So, once this port profile has been defined by the network team in the VSM, it will be available for the system team in the vSphere interface, so it can be assigned to any given machine. This illustrates the separation of tasks between teams.

In the port profile corresponding to the uplinks (i.e. to the physical switches) MAC-pinning technology was used for the EtherChannel. This mode of operation has the advantage of being usable with any type of physical switch. The assignment of a VM's MAC address to a physical server interface is by round-robin.



```
Veth236        t2gpns   . Network   up      238       auto    auto    --
Veth237        t4ges     , Network up      224       auto    auto    --
Veth238        t4gesq    , Network up      226       auto    auto    --
Veth239        tcgetem   , Network up      248       auto    auto    --
Veth240        tcgetem   , Network up      245       auto    auto    --
Veth241        t3geld  , Network   up      240       auto    auto    --
Veth242        t3geld  , Network   up      241       auto    auto    --
Veth243        tcgecr . Network A  up      248       auto    auto    --
Veth244        tcgecr . Network A  up      245       auto    auto    --
Veth245        tcgpde     , Networ up      248       auto    auto    --
Veth246        tcgpde     . Networ up      245       auto    auto    --
Veth247        tcgpinv       ., Net up     248       auto    auto    --
Veth248        tcgpinv       ., Net up     245       auto    auto    --
Veth249        tcgpinv       ', Net up     248       auto    auto    --
```

Figure 4.2: VMs shown as if connected to the physical ports of a Cisco switch (Nexus 1000v)

# Conclusion

The solutions offered for networking virtual machines are many and varied. This variety is both an advantage and a disadvantage. It is difficult to be completely sure that a choice will be the most effective and durable, given that recent solutions are often not well tested. New solutions are proposed constantly, replacing those that, not long before, were all the rage. Often they depend on a particular hypervisor, particular servers, particular physical switches and a particular organisation. In the range of solutions that have been presented here, you will no doubt find the one that is most appropriate if you have defined all your requirements in advance.

# References

[1Cisco-Nexus] Cisco Nexus 1000v Series Switch Deployment Guide with Cisco Unified Computing System
http://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps9902/guide_c07-704280.html

[SR-IOV_en] Using SR-IOV with Intel Ethernet Server Adapters (English)
http://www.intel.com/support/network/adapter/pro100/sb/CS-031492.htm

[SR-IOV_fr] Using SR-IOV with Intel Ethernet Server Adapters (French)
http://www.intel.com/support/fr/network/adapter/pro100/sb/CS-031492.htm

[3SR-IOV_primer] PCI-SIG SR-IOV Primer: An Introduction to SR-IOV Technology
http://www.intel.com/content/www/us/en/pci-express/pci-sig-sr-iov-primer-sr-iov-technology-paper.html

[2VMware_concepts] VMware Virtual Networking Concepts
http://www.vmware.com/files/pdf/virtual_networking_concepts.pdf

[vSphere_pricing] http://www.vmware.com/products/datacenter-virtualization/vsphere/pricing.html

# Glossary

| | |
|---|---|
| **ACL** | Access Control List. The list of IP addresses authorised to pass through network equipment or not. |
| **ARP** | Address Resolution Protocol |
| **IDS** | Intrusion Detection System. A mechanism that allows suspicious traffic circulating on the network to be detected. |
| **NETFLOW** | Protocol allowing information about traffic exchanges to be collected |
| **PVLAN** | Private Virtual Local Area Network |
| **QoS** | Quality of Service |
| **RSPAN** | Remote Switched Port ANalyser. Functionality allowing data passing through one port to another port to be copied. |
| **SNMP** | Simple Network Management Protocol |
| **SR-IOV** | Single Root I/O Virtualisation. Specification allowing a network card to appear as a set of physical cards. These can then be assigned to virtual machines. |
| **VDS** | Virtual Distributed Switches Advanced virtual switch allowing, among other things, management of several virtual switches distributed across different servers. |
| **VEB** | Virtual Ethernet Bridge Basic virtual switch allowing virtual machines to be linked with one another and a physical network. |
| **VEM** | Virtual Ethernet Module A module of the Nexus 1000v which, when integrated into the servers, performs the switching of the virtual machines. |
| **VEPA** | Virtual Ethernet Port Aggregators Mechanism allowing traffic to be sent from virtual machines to a physical switch. |
| **VSM** | Virtual Supervisor Module. Software allowing VEMs to be monitored; this is the core of the Nexus 1000v. |

Complete BPDs available at http://services.geant.net/cbp/Pages/Home.aspx

campus-bp-announcements@terena.org