The attached DRAFT document (provided here for historical purposes) has been superseded by the following publication:

Publication Number: **NIST Special Publication (SP) 800-125B**

Title: **Secure Virtual Network Configuration for Virtual Machine (VM) Protection**

Publication Date: **3/29/2016**

- Final Publication: http://dx.doi.org/10.6028/NIST.SP.800-125B (which links to http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-125B.pdf).
- Related Information on CSRC: http://csrc.nist.gov/publications/PubsSPs.html#SP-800-125-B
- Information on other NIST cybersecurity publications and programs can be found at: http://csrc.nist.gov/

The following information was posted with the attached DRAFT document:

Sep. 29, 2015

SP 800-125 B

DRAFT Secure Virtual Network Configuration for Virtual Machine (VM) Protection

NIST requests public comments on Draft Special Publication 800-125B, *Secure Virtual Network Configuration for Virtual Machine (VM) Protection*. VMs constitute the primary resource to be protected in a virtualized infrastructure, since they are the compute engines on which business/mission critical applications of the enterprise are run. Further, since VMs are end-nodes of a virtual network, the configuration of virtual network forms an important element in the security of VMs and their hosted applications. The virtual network configuration areas considered for VM protection in this document are – Network Segmentation, Network Path Redundancy, Firewall Deployment Architecture and VM Traffic Monitoring. The configuration options in each of these areas are analyzed for their advantages and disadvantages and security recommendations are provided.

<u>The specific areas where comments are solicited are:</u>

- Advantages and Disadvantages of the various configuration options in the four virtual network configuration areas.
- The Security Recommendations

The public comment period closes on **October 23, 2015**.
Send comments to: sp800-125b @nist.gov. Please use the Comment Template provided below, using the following "Type" codes for comments: E - editorial; G - general; T - technical.

DRAFT NIST Special Publication 800-125B

Secure Virtual Network Configuration for Virtual Machine (VM) Protection

Ramaswamy Chandramouli

This publication is available free of charge from:
http://dx.doi.org/10.6028/NIST.SP.XXX

COMPUTER SECURITY

National Institute of Standards and Technology
U.S. Department of Commerce

DRAFT NIST Special Publication 800-125B

Secure Virtual Network Configuration for Virtual Machine (VM) Protection

Ramaswamy Chandramouli
Computer Security Division
Information Technology Laboratory

This publication is available free of charge from:
http://dx.doi.org/10.6028/NIST.SP.XXX

September 2015

U.S. Department of Commerce
Penny Pritzker, Secretary

National Institute of Standards and Technology
Willie May, Under Secretary of Commerce for Standards and Technology and Director

Authority

This publication has been developed by NIST in accordance with its statutory responsibilities under the Federal Information Security Modernization Act (FISMA) of 2014, 44 U.S.C. § 3541 *et seq.*, Public Law (P.L.) 113-283. NIST is responsible for developing information security standards and guidelines, including minimum requirements for federal information systems, but such standards and guidelines shall not apply to national security systems without the express approval of appropriate federal officials exercising policy authority over such systems. This guideline is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130, Section 8b(3), *Securing Agency Information Systems*, as analyzed in Circular A-130, Appendix IV: *Analysis of Key Sections*. Supplemental information is provided in Circular A-130, Appendix III, *Security of Federal Automated Information Resources*.

Nothing in this publication should be taken to contradict the standards and guidelines made mandatory and binding on federal agencies by the Secretary of Commerce under statutory authority. Nor should these guidelines be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the OMB, or any other federal official. This publication may be used by nongovernmental organizations on a voluntary basis and is not subject to copyright in the United States. Attribution would, however, be appreciated by NIST.

National Institute of Standards and Technology Special Publication 800-125B
Natl. Inst. Stand. Technol. Spec. Publ. 800-125B, 27 pages (Sept 2015)
CODEN: NSPUE2

This publication is available free of charge from:
http://dx.doi.org/10.6028/NIST.SP.XXX

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by NIST, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

There may be references in this publication to other publications currently under development by NIST in accordance with its assigned statutory responsibilities. The information in this publication, including concepts and methodologies, may be used by federal agencies even before the completion of such companion publications. Thus, until each publication is completed, current requirements, guidelines, and procedures, where they exist, remain operative. For planning and transition purposes, federal agencies may wish to closely follow the development of these new publications by NIST.

Organizations are encouraged to review all draft publications during public comment periods and provide feedback to NIST. All NIST Computer Security Division publications, other than the ones noted above, are available at http://csrc.nist.gov/publications.

Public comment period: *Sept 28, 2015* through *Oct 23, 2015*

All comments are subject to release under the Freedom of Information Act (FOIA).

National Institute of Standards and Technology
Attn: Computer Security Division, Information Technology Laboratory
100 Bureau Drive (Mail Stop 8930) Gaithersburg, MD 20899-8930
Email: sp800-125b@nist.gov

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the Nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL's responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security and privacy of other than national security-related information in federal information systems. The Special Publication 800-series reports on ITL's research, guidelines, and outreach efforts in information system security, and its collaborative activities with industry, government, and academic organizations.

Abstract

Virtual Machines (VMs) are key resources to be protected since they are the compute engines hosting mission-critical applications. Since VMs are end-nodes of a virtual network, the configuration of the virtual network forms an important element in the security of VMs and their hosted applications. The virtual network configuration areas discussed in this documentation are: Network Segmentation, Network path redundancy, firewall deployment architecture and VM Traffic Monitoring. The various configuration options under these areas are analyzed for their advantages and disadvantages and a set of security recommendations are provided.

Keywords

VLAN; Overlay Network; Virtual Firewall; Virtual Machine; Virtual Network Segmentation;

Executive Summary

Data center infrastructures are rapidly becoming virtualized due to increasing deployment of virtualized hosts (also called hypervisor hosts). Virtual Machines (VMs) are the key resources to be protected in this virtualized infrastructure since they are the compute engines hosting mission-critical applications of the enterprise. Since VMs are end-nodes of a virtual network, the configuration of the virtual network forms an important element in the overall security strategy for VMs.

The purpose of this NIST Special Publication is to provide an analysis of various virtual network configuration options for protection of virtual machines (VMs) and provide security recommendations based on the analysis. The configuration areas, which are relevant from a security point of view, that are discussed in this publication are: **Network Segmentation, Network Path Redundancy, Firewall Deployment Architecture and VM Traffic Monitoring.** Different configuration options in each of these areas have different advantages and disadvantages. These are identified in this publication to arrive at a set of one or more security recommendations for each configuration area.

The motivation for this document is the trend in US Federal government agencies to deploy server virtualization within their internal IT infrastructure as well as the use of VMs provided by a cloud service provider for deploying agency applications. Hence the target audience is Chief Information Security Officers (CISO) and other personnel/contractors involved in configuring the system architecture for hosting multi-tier agency applications and for provisioning the necessary security protections through appropriate virtual network configurations. The intended goal is that the analysis of the various configuration options (in terms of advantages and disadvantages) provided in this report, along with security recommendations, will facilitate making informed decisions with respect to architecting the virtual network configuration. Such a configuration is expected to ensure the appropriate level of protection for all VMs and the application workloads running in them in the entire virtualized infrastructure of the enterprise.

Table of Contents

1 Introduction – Virtualized Infrastructures & Virtual Machine

A significant trend in the buildup of modern data centers is the increasing deployment of virtualized hosts. A virtualized host is a physical host with a server virtualization product (i.e., the hypervisor) running inside and hence capable of supporting multiple computing stacks each with different platform configuration (O/S & Middleware). The individual computing stack inside a virtualized host (also called hypervisor host) is encapsulated in an entity called virtual machine (VM). A VM being a compute engine has resources assigned to it – such as processors, memory, storage etc and these are called virtual resources. A VM's computing stack consists of O/S (called Guest O/S), Middleware (optional) and one or more application programs. Invariably, the application programs loaded into a VM are server programs (e.g., webserver, DBMS) and hence the whole process of deploying a virtualized host with multiple VMs running inside it, is called Server Virtualization.

A data center with predominant presence of hypervisor/virtualized hosts is said to have a virtualized infrastructure. The hypervisor product inside a virtualized host has the capability to define a network for linking the various VMs inside a host with each other and to the outside (physical) enterprise network. This network is called a Virtual Network, since the networking appliances are entirely software-defined. The core software-defined components of this virtual network are: Virtual Network Interface Cards (vNICs) inside each VM and the virtual switches (vSwitch) defined to operate inside the hypervisor kernel. The virtual switches, in turn, are connected to the physical network interface cards (pNICs) of the virtualized host to provide a communication path for applications (including Guest O/S) running inside VMs to interact with computing/storage elements in the physical network of the data center.

Being the communication pathway for VMs, the virtual network and the associated configuration parameters play a critical role in ensuring the security of the VM as a whole and in particular the mission-critical applications running inside them. The virtual network configuration areas, which are relevant from a security point of view, that are discussed in this documentation are: **Network Segmentation, Network Path Redundancy, Firewall Deployment Architecture and VM Traffic Monitoring.** Different configuration options in each of these areas have different advantages and disadvantages. The purpose of this document is to analyze these advantages and disadvantages from a security viewpoint and provide one or more security recommendations.

1.1 Out of scope

Based on the material discussed so far, it should be clear that this document is seeking to address only network-level protections for a VM. Two other areas that need to be addressed for ensuring the overall security of the VM and the applications hosted on them are – Host-level protection and VM data protection. These two areas are outside the scope of this document. Most of the host-level protection measures needed for a VM such as robust authentication, support for secure access protocols (e.g., SSH) are no different than the ones for their physical counterparts (i.e., physical servers). There are only a few host-level operations that are specific to VM that need secure practices (e.g., re-starting VMs from snapshots). The VM data protection measures have also been not included within the scope of this document since data associated with a VM are generally stored under well-established storage networking technologies (e.g., iSCSI, Fiber Channel etc).

1.2 Organization of this Publication

The organization of the rest of this publication is as follows:
Section 2 – discusses five network segmentation approaches for virtualized infrastructures
Section 3 – discusses the technique for creating network path redundancy in virtual networks
Section 4 – discusses three types of firewall usage for control of virtual network traffic
Section 5 – discusses two configuration approaches for capturing traffic for VM monitoring.

2 Network Segmentation Configurations for VM Protection

There is a viewpoint among security practitioners that network segmentation is a purely network management technique and not a security protection measure. However, many practitioners consider network segmentation as an integral part or at least a preliminary step of a defense-in-depth network security strategy. There are some standards such as PCI DSS 3.0 that calls forth for network segmentation as a security requirement for data protection.

The five network segmentation approaches discussed in this section are organized in their increasing order of scalability. The main motivation for network segmentation is to achieve logical separation for applications of different sensitivity levels in the enterprise. The initial approach to achieve this is by hosting all applications of a given sensitivity level in one VM and hosting all VMs of the same sensitivity level (based on hosted applications) in a given virtualized host (Section 2.1). This is strictly not a network segmentation approach (since it does not involve configuration of a network parameter) but is still included as one of the network segmentation approach since the objective of providing VM protection is met. Sections 2.2 & 2.3 discuss approaches for creating virtual network segments inside a virtualized host using virtual switches and virtual firewalls respectively. Truly scalable (data center wide) approaches for creating virtual network segments that span multiple virtualized hosts are discussed in sections 2.4 & 2.5 based on VLAN and overlay networking technologies respectively.

2.1 Segmentation based on Virtualized Hosts

When enterprise applications of different sensitivity levels were starting to be hosted in VMs, the initial network-based protection measure that was adopted was to locate applications of the different sensitivity levels and their hosting VMs in different virtualized hosts. This isolation between applications was extended into the physical network of the data center by connecting these hypervisor hosts to different physical switches and regulating the traffic between these physical switches using firewall rules. Alternatively, virtualized hosts carrying application workloads of different sensitivity levels were mounted in different racks so that they are connected to different Top of the Rack (ToR) switches.

2.1.1 Advantages

The most obvious advantage of the segmentation of VMs using the above approach is simplicity of network configuration and ease of subsequent network monitoring since traffic flowing into and out of VMs hosting workloads of different sensitivity levels are physically isolated.

2.1.2 Disadvantages

The basic economic goal of full hardware utilization will not be realized if any virtualized host is utilized for hosting VMs of a single sensitivity level as there may be different numbers of applications in each sensitivity level. This will also have an impact on the workload balancing for the data center as a whole. This solution will also hamper the flexibility in VM migration as the target hypervisor host should be of the same sensitivity level (or any other classification criteria used – e.g., same department) as the source host.

2.2 Segmentation using Virtual Switches

An alternative to segmenting VMs by virtualized hosts is by connecting VMs belonging to different sensitivity levels to different virtual switches within a single virtualized host. The isolation of traffic between VMs of different sensitivity levels has to be still achieved by connecting the different virtual switches to different physical switches with their respective pathways going through different physical NICs of the virtualized host. Finally, of course, the traffic flow between these physical switches has to be regulated through the usual mechanisms such as the firewall.

2.2.1 Advantages

Segmenting the population of VMs using virtual switches as opposed to hosting them in different virtualized hosts promotes better utilization of hypervisor host resources while still maintaining ease of configuration. Further, by design, all hypervisor architectures prevent connection between virtual switches within a hypervisor platform, thus providing some security assurance.

2.2.2 Disadvantages

Connecting a single virtualized host to two different physical switches may present difficulty in the case of certain environments such as rack mounted servers. The flexibility in VM migration may still be hampered due to non-availability of ports in the virtual switches of the same sensitivity level (based on the sensitivity level of the migrating VM) in the target hypervisor host.

2.3 Network Segmentation using Virtual Firewalls

When Internet-facing applications (especially web applications) are run on (non-virtualized) physical hosts, a separate subnet called DMZ is created using physical firewalls. Similarly when VMs hosting web servers running internet-facing applications are deployed on a virtualized host, they can be isolated and run in a virtual network segment that is separated from a virtual network segment that is connected to the enterprise’s internal network. Just as two firewalls – one facing the internet and the other protecting the internal network – are needed in a physical network, there are two firewalls needed inside a virtualized host to create a virtual network equivalent of a DMZ. The major difference in the latter case, is that, the two firewalls have to run in a virtual network and hence these firewalls are software firewalls run as a virtual security appliance on dedicated (usually hardened) VMs. A configuration for DMZ inside a virtualized host is shown in Figure 1.

As one can see from Figure 1, there are 3 virtual switches – VS1, VS2 and VS3 inside the virtualized host. The uplink port of VS1 is connected to the physical NIC –pNIC1 that is connected to a physical

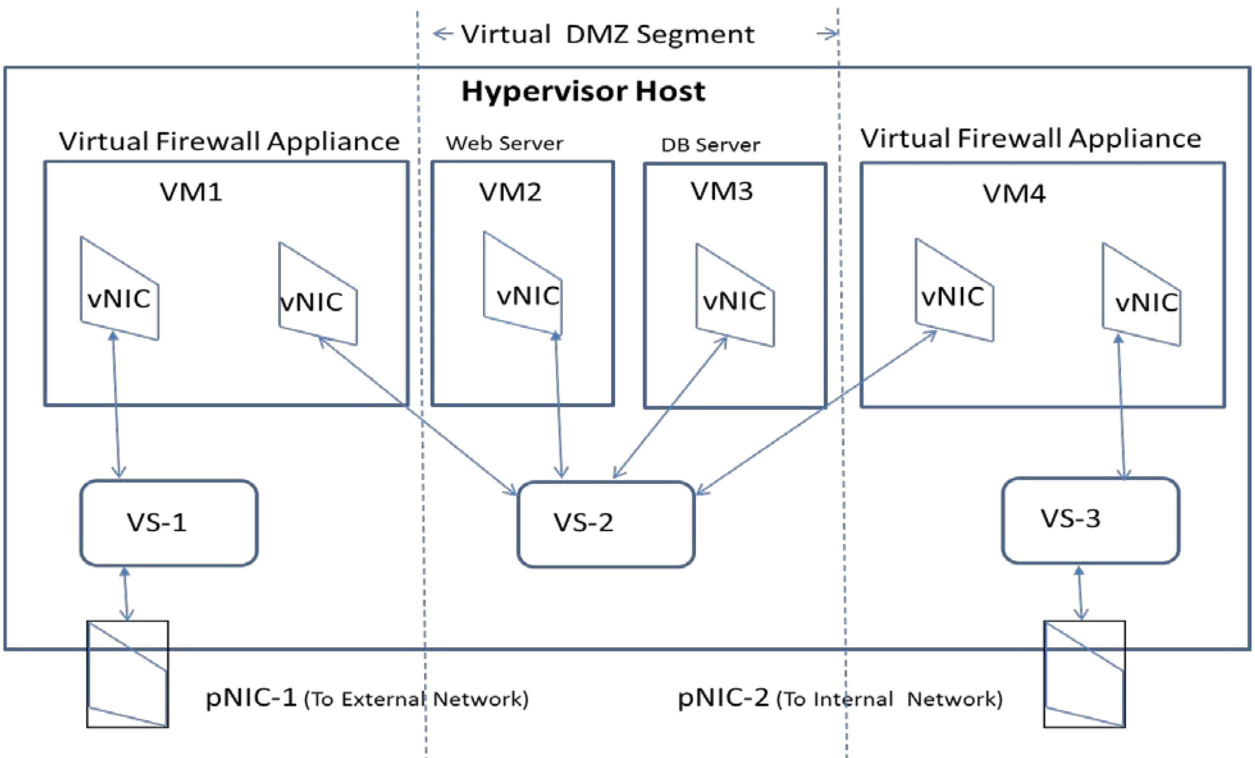

Figure 1 – Virtual Network Segmentation using Virtual Switches & Virtual Firewalls

switch in the external network. Similarly the uplink port of VS3 is connected to the physical NIC – pNIC2 that is connected to a physical switch in the data center’s internal network. The firewall appliances running in VM1 and VM4 respectively play the role of internet-facing firewall and internal firewall respectively. This is due to the fact that VM1 acts as the traffic control bridge between the virtual switches VS1 and VS2 while VM4 acts as the traffic control bridge between the virtual switches VS2 and VS3. What this configuration has done is to create an isolated virtual network segment based on the virtual switch VS2 (DMZ of the virtual network), since VS2 can only communicate with the internet using firewall in VM1 and with the internal network using the firewall in VM4. Hence all VMs connected to the virtual switch VS2 (in our configuration the VMs – VM2 & VM3) run in this isolated virtual network segment as well, with all traffic into and from them to/from external network controlled by firewall in VM1 and all traffic into and from them to/from internal network controlled by firewall in VM4.

Looking at the above virtual network configuration from a VM point of view (irrespective of whether they run a firewall or a business application), we find that VMs VM1 and VM4 are multi-homed VMs with at least one of the vNICs connected to a virtual switch whose uplink port is connected to a physical NIC. By contrast, the VMs VM2 & VM3 are connected only to a Internal-only virtual switch (i.e., VS2 - that is not connected to any physical NIC. A virtual switch that is not connected to any physical NIC is called an "Internal-only Switch") and hence we can state that VMs connected only to Internal-only switches enjoy a degree of isolation as they run in an isolated virtual network segment.

2.3.1 Advantages

- Virtual firewalls come packaged as Virtual Security Appliances on purpose-built VMs and hence are easy to deploy.
- Since virtual firewalls run on VMs, they can be easily integrated with virtualization management tools/servers and hence can be easily configured (especially their security rules or ACLs) as well.

2.3.2 Disadvantages

- The VMs hosting the virtual firewall appliance compete for the same hypervisor resources (i.e., CPU cores, memory etc) as VMs running business applications.
- The span of the protected network segment that is created is limited to a single virtualized host. Migration of the VMs in the protected network segment (for load balancing or fault tolerance reasons) to another virtualized host is possible only if the target host has identical virtual network configuration. Creating virtualized hosts with identical virtual network configuration may limit full utilization of the overall capacity of the hosts. On the flip side, it may constrain VM migration flexibility.

2.4 Network Segmentation using VLANS in Virtual Network

VLANs were originally implemented in data centers where nodes were configured to operate in Ethernet-switched modes for ease of control and network management (e.g., broadcast containment). Being a network segmentation technique, it provided value as a security measure because of the traffic isolation effect. In a data center with all physical (non-virtualized) hosts, a VLAN is defined by assigning a unique ID called VLAN tag to one or more ports of a physical switch. All hosts connected to those ports then become members of that VLAN ID. Thus a logical grouping of servers (hosts) is created, irrespective of their physical locations, in the large flat network of a data center (since the 6-byte MAC address of the host's NICs do not reflect its topological location (the switch/router to which it is connected)). An example of a VLAN Configuration is shown in Figure 2.

The concept of VLAN can be extended and implemented in a data center with virtualized hosts (in fact inside each virtualized host) using virtual switches with ports or port groups that support

VLAN tagging and processing. In other words, VLAN IDs are assigned to ports of a virtual switch inside a hypervisor kernel and VMs are assigned to appropriate ports based on their VLAN membership. These VLAN-capable virtual switches can perform tagging of all packets going out of a VM with a VLAN tag (depending upon which port it has received the packet from) and can route an incoming packet with a specific VLAN tag to the appropriate VM by sending it through a port whose VLAN ID assignment equals the VLAN tag of the packet. Corresponding to the VLAN configuration of the various virtual switches inside a virtualized host, link aggregation should be configured on links linking the physical NICs of these virtualized hosts to the physical switch of the data center. This is necessary so that these links can carry traffic corresponding to all VLAN IDs configured inside that virtualized host. Further, the ports of the physical switch which forms the termination point of these links should also be configured as trunking ports (capable of receiving and sending traffic belonging to multiple VLANs). A given VLAN ID can be assigned to ports of virtual switches located in multiple virtualized hosts. Thus we see that the combined VLAN configuration consisting of the configuration inside the virtualized host (assigning VLAN IDs to ports of virtual switches or virtual NICs of VMs) and the configuration outside the virtualized host (link aggregation and port trunking in physical switches) provide a pathway for VLANs defined in the physical network to be carried into a virtualized host (and vice versa), thus providing the ability to isolate traffic emanating from VMs distributed throughout the data center and thus a means to provide confidentiality and integrity protection to the applications running inside those VMs.

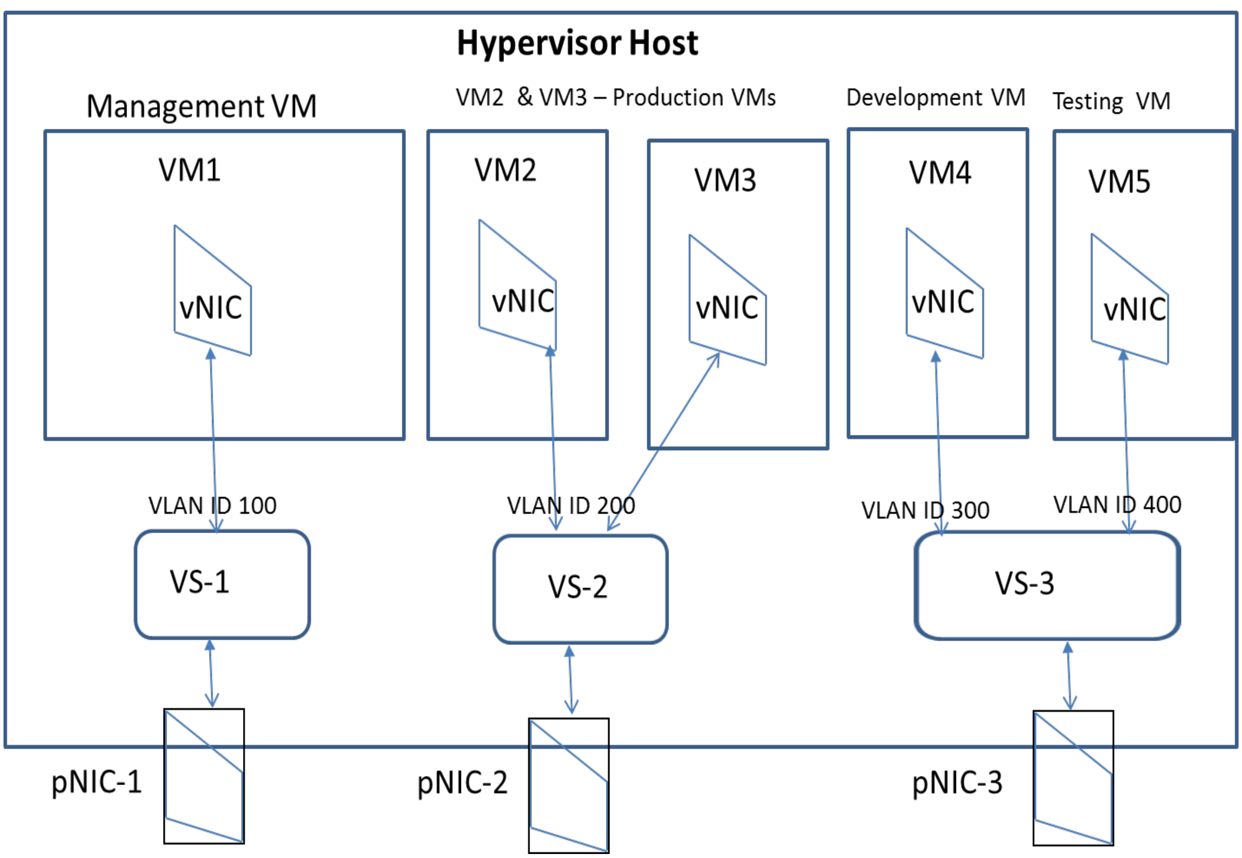

Figure 2 – An Example VLAN Configuration

Thus a logical group of VMs is created with the traffic among the members of that group being isolated from traffic belonging to another group. The logical separation of network traffic provided by VLAN configuration can be based on any arbitrary criteria. Thus we can have:

(a) Management VLAN for carrying only Management traffic (used for sending management/configuration commands to the hypervisor),
(b) VM Migration VLAN for carrying traffic generated during VM migration (migrating VMs from one virtualized host to another for availability and load balancing reasons,
(c) Logging VLAN for carrying traffic used for Fault Tolerant Logging,
(d) Storage VLAN for carrying traffic pertaining to NFS or iSCSI storage,
(e) Desktop VLAN for carrying traffic from VMs running Virtual Desk Infrastructure software and last but not the least,
(f) a set of production VLANs for carrying traffic between the production VMs (the set of VMs hosting the various business applications). These days, enterprise application architectures are made up of three tiers: Webserver, Application and Database tiers. A separate VLAN can be created for each of these tiers with traffic between them regulated using firewall rules. Further in a cloud data center, VMs may belong to different consumers or cloud users, and the cloud provider can provide isolation of traffic belonging to different clients using VLAN configuration. In effect what is done is that one or more logical or virtual network segments are created for each tenant by making VMs belonging to each of them being assigned to/connected to a different VLAN segment. In addition to confidentiality and integrity assurances (referred to earlier) that is provided by logical separation of network traffic, different QoS rules can be applied to different VLANs (depending upon the type of traffic carried), thus providing availability assurance as well. An example of VLAN-based virtual network segmentation inside a hypervisor host is given in Figure 2.

In summary, we saw that network segmentation using VLAN logically groups devices or users, by function, department or application irrespective of their physical location on the LAN. The grouping is obtained by assignment of an identifier called VLAN ID to one or more ports of a switch and connecting the computing units (physical servers or VMs) to those ports.

2.4.1 Advantages

- Network segmentation using VLANs is more scalable than approaches using virtual firewalls (section 2.3). This is due to the following:
  (a) The granularity of VLAN definition is at the port level of a virtual switch. Since each virtual switch can support around 64 ports, the number of network segments (in our context VLANs) that can be defined inside a single virtualized host is much more than what is practically possible using firewall VMs.
  (b) Network segments can extend beyond a single virtualized host (unlike the segment defined using virtual firewalls) since the same VLAN ID can be assigned to ports of virtual switches in different virtualized hosts. Also the total number of network segments that can be defined in the entire data center is around 4000 (since the VLAN ID is 12 bits long).

2.4.2 Disadvantages

- The configuration of the ports in the physical switch (and their links) attached to a virtualized host must exactly match the VLANs defined on the virtual switches inside that virtualized host. This results in tight coupling between virtual network and some portion of the physical network of the data center. The consequence of this tight coupling is that the port configuration of the physical switches has to be frequently updated since the VLAN profile of the attached virtualized host may frequently change due to migration of VMs between VLANs and between virtualized hosts as well as due to change in profile of applications hosted on VMs. More specifically, the MAC address to VLAN ID mapping in the physical switches may go out of synch, resulting in some packets being flooded through all ports of the physical switch. This in turn results in increased workload on the some hypervisors due to processing packets that are not targeted towards any VM it is hosting at that point in time.
- The capability to define network segment spanning virtualized hosts may spur administrators to create a VLAN segment with a large span for providing greater VM mobility (for load balancing and availability reasons). This phenomenon called VLAN sprawl may result in more broadcast traffic for the data center as a whole and also has the potential to introduce configuration mismatch between the VLAN profile of virtualized hosts and their associated physical switches (discussed earlier).

2.5 Network Segmentation using Overlay-based Virtual Networking

In the Overlay-based virtual networking, isolation is realized by encapsulating an Ethernet frame received from a VM as follows. Out of the three encapsulation schemes (or overlay schemes) – VXLAN, GRE and STT, let us now look at the encapsulation process in VXLAN through components shown in Figure 3. First, the Ethernet frame received from a VM, that contains the MAC address of destination VM is encapsulated in two stages: (a) First with the 24 bit VXLAN ID (virtual Layer 2 (L2) segment) to which the sending/receiving VM belongs and (b) two, with the source/destination IP address of VXLAN tunnel endpoints (VTEP), that are kernel modules residing in the hypervisors of sending/receiving VMs respectively. The source IP address is the IP address of VTEP that is generating the encapsulated packet and the destination IP address is the IP address of VTEP in a remote hypervisor host sitting anywhere in the data center network that houses the destination VM. Thus, we see that VXLAN encapsulation enables creation of a virtual Layer 2 segment that can span not only different hypervisor hosts but also IP subnets within the data center.

Both encapsulations described above that are used to generate a VXLAN packet are performed by a hypervisor kernel module called the overlay module. One of the key pieces of information that this overlay module needs is the mapping of the MAC address of the remote VM to its corresponding VTEP's IP address (i.e., the IP address of the overlay end node in the hypervisor host hosting that remote VM). The overlay module can obtain this IP address in two ways: either by flooding using IP learning packets or configuring the mapping information using a SDN controller that uses a standard protocol to deliver this mapping table to the overlay modules in each hypervisor host. The second approach is more desirable since learning using flooding results in unnecessary network traffic in the entire virtualized infrastructure. The VXLAN based network segmentation can be configured to provide isolation among resources of multiple tenants of a cloud data center as follows. A particular tenant can be assigned two or more VXLAN segments (or IDs). The tenant can make use of multiple VXLAN segments by assigning VMs hosting each tier (Web, Application or Database) to the same or different VXLAN segments. If VMs belonging to a client are in different VXLAN segments, selective connectivity can be established among those VXLAN segments belonging to the same tenant through suitable firewall configurations, while communication between VXLAN segments belonging to different tenants can be prohibited.

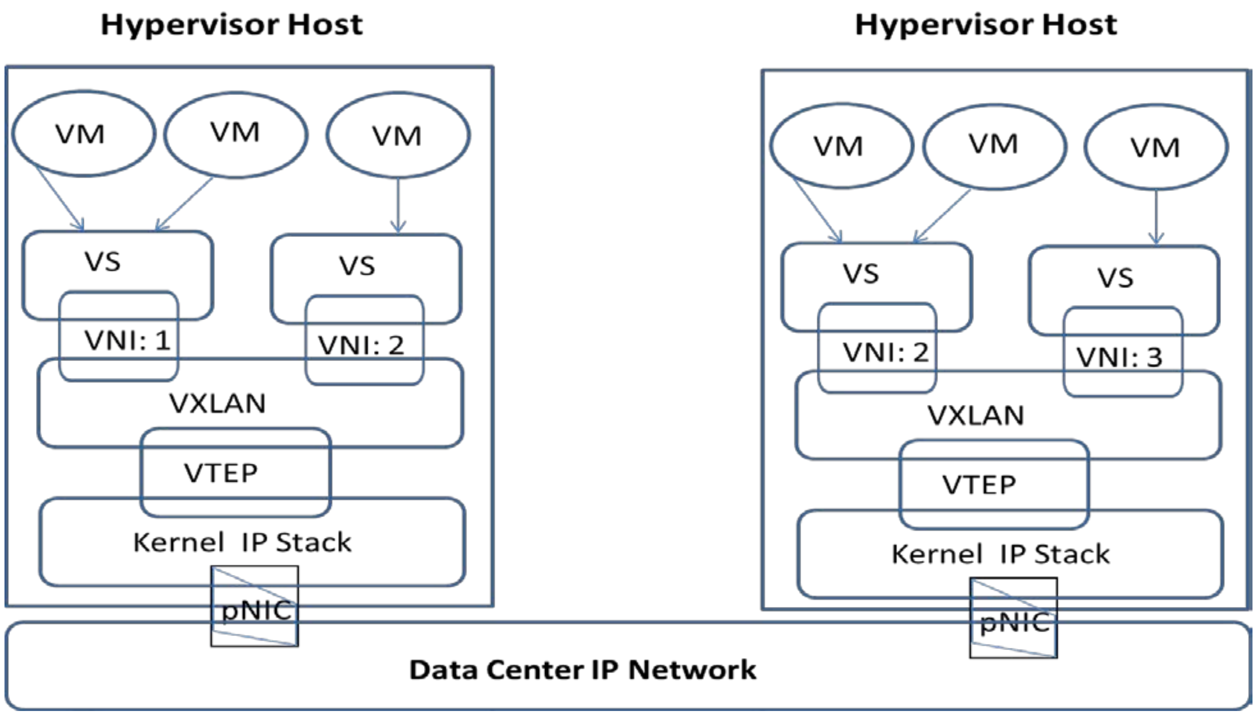

Figure 3 – Virtual Network Segmentation using Overlays (VXLAN)

2.5.1 Advantages of Overlay-based Network Segmentation

- The overlay-based network segmentation is infinitely scalable compared to the VLAN-based approach due to the following:
  - (a) A VXLAN network identifier (VNID) is a 24 bit field compared to the 12 bit VLAN ID. Hence the namespace for VXLANs (and hence the number of network segments that can be created) is about 16 million as opposed to 4096 for VLANs.
  - (b) Another factor contributing to scalability of the overlay scheme is that the encapsulating packet is an IP/UDP packet. Hence the number of network segments that can be defined is limited only by the number of IP subnets in the data center and not by the number of ports of virtual switches as in the case of VLAN-based network segmentation.
- In a data center that is offered for IaaS cloud service, isolation between the tenants (cloud service subscribers) can be achieved by assigning each of them at least one VXLAN segment (denoted by a unique VXLAN ID). Since VXLAN is a logical L2 layer network (called overlay network) running on top of a physical L3 layer (IP) network inside the data center, the latter is independent of the former. In other words, no device of the

physical network has its configuration dependent on the configuration in any part of virtual network. The consequence of this feature is that it gives the freedom to locate the computing and/or storage nodes belonging to a particular client in any physical segment of the data center network. This freedom and flexibility in turn, helps to locate those computing/storage resources based on performance (high performance VMs for data/compute intensive workloads) and load balancing considerations. This results in greater VM mobility and hence its availability.

2.5.2 Disadvantages of Overlay-based Network Segmentation

- A given network segment (a particular VXLAN ID) can exist in any virtualized host in the data center. Hence routing packets between any two VMs requires large mapping tables (in the overlay-network end points) in order to generate encapsulated packets - since the MAC address of the destination VM could be located in any IP subnet and in any virtualized host in the data center. Building these mapping tables using just flooding technique is inefficient. Hence a control plane needs to be deployed in the virtualized infrastructure to populate the mapping tables for use by overlay packet generation module in the hypervisor. This creates an additional layer of control and adds to the complexity of network management.

2.6 Security Recommendations for Network Segmentation

VM-VN-R1: In all VLAN deployments, the switch (physical switch connecting to virtualized host) port configuration should be VLAN aware – i.e., its configuration should reflect the VLAN profile of the connected virtualized host.

VM-VN-R2: Large data center networks with hundreds of virtualized hosts and thousands of VMs and requiring many segments should deploy an overlay-based virtual networking because of scalability (Large Namespace) and Virtual/Physical network independence.

VM-VN-R3: Large overlay-based virtual networking deployments should always include either centralized or federated SDN controllers using standard protocols for configuration of overlay modules in various hypervisor platforms.

3. Network Path Redundancy Configurations for VM Protection (Multipathing)

Configuring multiple communication paths for a VM to communicate is essential for ensuring the availability aspect of security and hence any network configuration for achieving this can also be looked upon as an integral part of network-based protection for VMs.

Before we look at the various options available for configuring multiple communication paths for VMs, we have to look at the scope of this configuration area based on the state of network technology. First is that the physical network configuration in the data center will be largely unaffected by the presence of virtualized hosts except some tasks such as VLAN configuration of ports in the physical switches connecting to the virtualized hosts as well as configuring the associated links as trunk links. Hence our configuration options relating to network path redundancy for VMs are confined to the virtual network inside the virtualized hosts including their physical NICs. Secondly the virtual network configuration features provided in most hypervisor offerings involve a combination of load balancing and failover policy options. From a network path redundancy perspective, we are only interested in the failover policy options.

3.1 NIC Teaming Configuration for Network Path Redundancy

Hypervisor offerings may differ in the policy configuration options that they provide for providing network path failover, but they have to provide a common configuration feature called NIC teaming or NIC bonding. NIC teaming allows administrators to combine multiple physical NICs into a NIC team for virtual network load balancing and NIC failover capabilities in a virtualized host. The members of the NIC team are connected to the different uplink ports of the same virtual switch. The NIC team can be configured both for failover purpose and load balancing purpose. Failover capability requires at least two physical NICs in the NIC team. One of them can be configured as "Active" and the other as "Standby". If an active physical NIC fails or traffic fails to flow through it, the traffic will start flowing (or be routed) through the standby physical NIC thus maintaining continuity of network traffic flow from all VMs connected to that virtual switch. This type of configuration is also called active-passive NIC bonding.

Some hypervisor offerings allow NIC teaming functionality to be defined at the VM-level. NIC teaming feature at the VM-level enables administrators to create a NIC team using virtual NICs of a VM enabling the VM's NICs to perform the same NIC team functionality inside the VM, just like their physical NIC counterparts do at the virtualized host level.

3.2 Policy Configuration Options for NIC Teaming

Then the next task is set the policy options relating to NIC teaming and this is the task for which configuration options available in different hypervisors are different. Again, we are interested in those options relating to failover and not load balancing since the explicit objective of the latter is to improve network performance rather than network availability. The different policy options for network failover pertain to different ways in which the NIC team detects NIC/link failure and perform failover.

One policy option available for network failover detection looks for electrical signals from the physical NIC itself for detecting the physical NIC failure or the failure of the link emanating from the physical NIC. Another option available is to set up the functionality to send beacon probes (Ethernet broadcast frames) on a regular basis to detect both link failure and configuration problems.

3.3 Security Recommendations for Configuring Network Path Redundancy

The following recommendations seek to improve the fault tolerance (redundancy) already provided by NIC teaming.

VM-MP-R1: It would be preferable to use physical NICs that use different drivers in the NIC team. The failure of one driver will only affect one member of the NIC team and will keep the traffic flowing through the other physical NICs of the NIC team.

VM-MP-R2: If multiple PCI buses are available in the virtualized host, each physical NIC in the NIC team should be placed on a separate PCI bus. This provides fault tolerance against the PCI bus failure in the virtualized host.

VM-MP-R3: The network path redundancy created within the virtual network of the virtualized host should also be extended to the immediate physical network links emanating from the virtualized host. This can be achieved by having the individual members of the NIC team (i.e., the two or more physical NICs) connected to different physical switches.

4 VM protection through Traffic Control using Firewalls

The primary use of a firewall is for traffic control. In a virtualized infrastructure, traffic control for VM protection is to be exercised for the following two scenarios.

- Traffic flowing between any two virtual network segments (or subnets)
- All traffic flowing into and out of a VM

There are several use cases where traffic flowing between two VMs (or groups of VMs) need to be controlled, regardless of whether the VMs are resident within the same virtualized host or in different virtualized hosts. The following are some of them:

- The total set of applications in an enterprise may be of different sensitivity levels. It is impractical to segregate them by running each category (applications of the same sensitivity level) in different virtualized hosts. Hence a given virtualized host may contain VMs of different sensitivity levels (assuming that all applications hosted in a VM are of the same sensitivity level). Hence there is the need to control traffic between VMs within the same virtualized host (inter-VM intra-host traffic).
- Most large scale enterprise applications are designed with three-tier architecture – Web Server, Application Logic and Database tiers. There may be multiple VMs associated with each tier and generally for reasons of load balancing and security, VMs hosting applications belonging to a particular tier are generally assigned to the same network segment or subnet though spanning across multiple virtualized hosts. This type of configuration gives rise to the presence of Web Server subnet (segment), Database Server subnet etc. However, for any enterprise application to function, the webserver tier of the application needs to talk to the corresponding application logic tier which in turn may need to communicate with database tier of that application. Hence it is obvious that a VM hosting a web server tier and housed in

the subnet-A needs controlled connectivity to a VM hosting an application logic tier and housed in another subnet-B. Since a subnet itself can multiple virtualized hosts, it is needless to say that VMs belonging to different application tiers (on a dedicated subnet) may be located in different virtualized hosts and the traffic between them controlled as well (inter-VM inter-host traffic).

- In some enterprises, networks are segmented based on departments in an enterprise (this applies even if the underlying infrastructure is virtualized), the need for exchanging data selectively between applications belonging to two different departments (say marketing and manufacturing), may require communication between a VM in the marketing segment and a VM in the manufacturing segment.

The common requirement in all the use cases discussed above is that all inter-VM traffic must be subjected to policy-based inspection and filtering. Inter-VM traffic is initiated when a VM generates communication packets that are sent through a virtual NIC of that VM to the port of a virtual switch defined inside the hypervisor kernel. If the target VM resides inside the same virtualized host, these packets are forwarded to another port in the same virtual switch. The target VM (dedicated to it) may either be connected to the same virtual switch or the connection to the target VM may go through another VM that acts as a bridge between virtual switches of the two communicating VMs. If the target VM resides in another virtualized host, these packets are sent to the uplink ports of that virtual switch to be forwarded to any of the physical NIC of that virtualized host. From there these packets travel through the physical network of the data center and on to the virtualized host where the target VM resides. The packets again travel through the virtual network in that virtualized host to reach the target VM. Hence it is clear that since VMs are end-nodes of a virtual network, the originating and ending network in any inter-VM communication are virtual networks. Hence a software-based virtual firewall either functioning in a VM or in the hypervisor kernel would be a natural mechanism to control inter-VM traffic. However, since connection between any two virtual segments (in different virtualized hosts at least) goes through a physical network, a physical firewall can also be deployed to control inter-VM traffic between VMs in different virtualized hosts. Hence this was one of the earliest approaches adopted for controlling inter-VM traffic. A physical firewall configuration to control inter-VM traffic is analyzed for its pros and cons in section 4.1. A subnet-level (VM-based) virtual firewall based approach for controlling inter-VM traffic is discussed in section 4.2 and its advantages and disadvantages are analyzed.

So far our discussion of firewall for traffic control function is about the first scenario where we are dealing with traffic flowing between two virtual network segments. Let us now look at the second scenario where traffic flowing into and out of a particular VM needs to be controlled. This situation arises when fine grained policies that pertain to communication packets emerging from and into a particular VM are needed. To enforce these policies, a mechanism to intercept packets between the virtual NIC of a VM to the virtual switch within the hypervisor kernel is needed. Such a mechanism is provided by another class of virtual firewalls called NIC-level or Hypervisor-mode firewall. The advantages and disadvantages of this class of virtual firewalls are discussed in section 4.3.

A brief overview of the three classes of firewalls referred above (physical firewall, subnet-level virtual firewall and kernel-based virtual firewall) is given below to facilitate analysis of their advantages and disadvantages.

- Physical Firewalls: This class of firewalls can perform their function either in hardware or software. The distinguishing feature is that no other software runs in the server platform where the firewall is installed – in other words the hardware of the server is dedicated to running only one application – the firewall application.
- Virtual Firewalls: This class of firewalls is entirely software-based running either in a dedicated VM or as a hypervisor kernel module. They are distinguished from physical firewalls by the fact that they share the computing, network and storage resources with other VMs within the hypervisor host where they are installed. The two sub-classes of virtual firewalls are:
  (a) Subnet-level virtual firewall: These run in a dedicated VM which is usually configured with multiple virtual NICs. Each virtual NIC is connected to a different subnet or security zone of the virtual network. Since they communicate with the virtual network only through the virtual NICs of the VM platform, they are agnostic to the type of virtual network.
  (b) NIC-level firewall: These firewalls are logically placed in between the virtual NIC of VMs and the virtual switch inside the hypervisor kernel. They function as loadable (hypervisor) kernel module using the hypervisor's introspection API. Thus they can intercept every packet coming into and out of an individual VM. Subsequent filtering of packets can be performed either in the hypervisor kernel itself or in a dedicated VM. In the latter case, the portion of the firewall functioning as a kernel module performs the function of just intercepting and forwarding the traffic to a VM-based module and the actual filtering of traffic is done in the VM-based module (just as a VM-based subnet-level virtual firewall does).

4.1 Physical Firewalls for VM Protection

In this early scheme, the inter-VM virtual network traffic inside a virtualized host is routed out of that virtual network (often called network in the box) on to the physical network (via the physical network interface cards (pNICs) connected to the uplink ports of the virtual switches to which VM are connected). On this network is installed a firewall with filtering rules pertaining to traffic flowing out of and into each VM on the virtualized host. The VLAN traffic emerging out of the virtualized host is inspected by this firewall and is then either dropped or passed back into the virtual network and on to the target VM.

4.1.1 Advantages & Disadvantages

The advantage of this early scheme is the leveraging of mature, sophisticated firewall rules and other capabilities of the firewall technology. However, the use of physical firewalls for inspection and filtering of virtual network traffic carries a number of disadvantages:

- The performance penalty due to increased latency involved in routing the virtual network traffic to the physical network outside the virtualized host and then back to the virtual network inside the virtualized host. This phenomenon is known as hairpinning.

- The error-prone manual process involved in maintaining the state information about various VMs as the composition of VMs inside a virtualized host may keep on changing due to VM migrations.
- The physical firewall may lack integration with virtualization management system. This in turn may hamper automation of provisioning and update of firewall rules that may be continuously changing due to change in profiles (due to type of application workloads) of VMs.

4.2 Virtual Firewalls – Subnet-level

The disadvantages and limitation of physical firewalls motivated the development of virtual firewalls. Virtual firewalls are entirely software-based artifacts and packaged as a virtual security appliance and run on specially prepared (hardened) VMs. The first generation of virtual firewalls operated in bridge mode – that is just like their physical counterpart, they can be placed at a strategic location within the network – in this case the virtual network of a virtualized host. Many of the offerings of this firewall are stateful and application types. In addition, many of them offer additional features such as NAT, DHCP, Site-to-Site IPsec VPN as well as load balancing for selective protocols such as TCP, HTTP & HTTPS. The advantages and limitations of subnet-level virtual firewall are as follows:

4.2.1 Advantages of Subnet-level Virtual Firewalls

- Avoids the need to route virtual network traffic inside the hypervisor host to physical network and back.
- The effort required to deploy is as easy as deploying any other VM.

4.2.2 Disadvantages of Subnet-level Virtual Firewalls

- The speed of packet processing is dependent on several factors such as number of CPU cores allocated to the VM hosting the firewall appliance, the TCP/IP stack of the O/S running the appliance and the switching speed of hypervisor switches.
- In virtualized hosts containing VMs running I/O intensive applications, there could be heavy hypervisor overhead. Even otherwise, since it functions in a VM, it takes away some of the CPU and memory resources of the hypervisor that could otherwise be used for running production applications.
- Since the virtual firewall is itself a VM, the integrity of its operation depends upon its relationship to application VMs. Uncoordinated migration of VMs in the hypervisor could alter this relationship and affect the integrity of its operation.
- Traffic flowing into and out of all portgroups and switches connected with the zones associated with the firewall are redirected to the VM hosting the firewall, resulting in unnecessary traffic (a phenomenon called Traffic Trombones).
- Firewall rules and state associated with a VM do not migrate automatically when a VM is live-migrated to another virtualized host. Hence that VM may lose its security protection, unless the same rules are reconfigured in the environment of the target virtualized host.

4.3 Virtual Firewalls – Kernel-based

Kernel-based virtual firewalls were designed to overcome the limitation of Subnet-level virtual firewalls. It comes packaged as a Loadable Kernel Module (LKM) – which means it is installed and run in the hypervisor kernel.

4.3.1 Advantages of Kernel-based Virtual Firewalls

- Much higher performance compared to a Subnet-level virtual firewall because of the fact that packet processing is done not using the VM-assigned resources (virtual CPUs & virtual memory) but using the hardware resources available to the hypervisor kernel.
- Since it is running as a hypervisor kernel module, its functionality cannot be monitored or altered by a rogue VM with access to virtual network inside the hypervisor host.
- It has the greatest visibility into the state of the VM including virtual hardware, memory, storage and applications besides the incoming and outgoing network traffic in each VM.
- It has direct access to all virtual switches and all the network interfaces of those switches. Hence the scope of its packet monitoring and filtering functionality not only includes inter-VM traffic but also traffic from VM to the physical network (through the physical NICs of the hypervisor host).
- Since it is a hypervisor kernel module, packet filtering functions operate between the Virtual Network Interface Cards (vNICs) of each VM and the hypervisor switch. The firewall rules (or ACLs) and state are logically attached to the VM interface and hence these artifacts move with the VM when it migrates to another virtualized host, thus providing continuity of security protection for the migrated VM.

4.3.2 Disadvantages of Kernel-based Virtual Firewalls

- Can have integration problem with some virtualization management tools having access to only VMs or virtual networks. This is due to the fact that this class of firewalls runs as a managed kernel process and is therefore neither a VM-resident program nor a component of the virtual network (such as a virtual switch or a virtual NIC) of the virtualized host.

4.4 Security Recommendations for Firewall Deployment Architecture

VM-FW-R1: In virtualized environments with VMs running delay-sensitive applications, virtual firewalls instead of physical firewalls should be deployed for traffic flow control, because, in the latter case, there is latency involved in routing the virtual network traffic to outside the virtualized host and back into the virtual network.

VM-FW-R2: In virtualized environments with VMs running I/O intensive applications, Kernel-based virtual firewalls should be deployed instead of Subnet-level virtual firewalls, since in the former, packet processing is performed in the kernel of the hypervisor at native hardware speeds.

VM-FW-R3: For both Subnet-level and Kernel-based virtual firewalls, it is preferable if the firewall integrates with a virtualization management platform rather than being accessible only through a standalone console. The former capability will enable provisioning of uniform firewall rules to multiple firewall instances easier than ones with

the latter capability – thus reducing the chances of configuration errors.

VM-FW-R4: For both Subnet-level and Kernel-based virtual firewalls, it is preferable if the firewall supports rules using higher-level components or abstractions (e.g., security group) in addition to the basic 5-tuple (Source/Destination IP address, Source/Destination Ports, Protocol etc).

5. VM Traffic Monitoring

Firewalls only ensure that inter-VM traffic conforms to some organizational information flow and security rules. However, to identify any traffic coming into or flowing out of VMs as malicious or harmful and to generate alerts or take preventive action, it is necessary to set up traffic monitoring capabilities to monitor all incoming/outgoing traffic of a VM.

To analyze communication packets going into or coming out of a VM, a functionality to copy those packets (incoming or outgoing) and send them to a network monitor application (also called analyzer application) is needed. This functionality is called port mirroring. The purpose of a network monitoring application is to perform security analysis, network diagnostics and generation of network performance metrics. In tune with the theme of this document, we only focus on the configuration options available in hypervisor to turn on the port mirroring functionality. Depending upon the hypervisor offering, this configuration option may exist as either a VM-configuration feature or virtual switch port configuration feature with the common goal being to set up a VM traffic monitoring capability.

5.1 Enabling VM Traffic Monitoring using VM Network Adapter Configuration

In some hypervisor offerings, the network monitoring application runs as a VM-based application. Hence this VM and its virtual NIC becomes the destination VM/vNIC (analyzer VM) to which traffic must be sent for analysis. The VM whose incoming/outgoing traffic is to be monitored (monitored VM) becomes then the source VM/vNIC. Thus the values “Source” and “Destination” are assigned to the “mirroring mode” configuration parameter of the network adapters (vNICs) respectively of the monitored VM and analyzer VM.

5.2 Enabling VM Traffic Monitoring using Virtual Switch Port Configuration

There are two ways that a virtual switch can be configured to enable visibility into traffic flowing into and out of a particular VM for use by a networking monitoring tool such as IDS or Sniffers. They are:

- In the earlier versions of a virtual switch, the only configuration option available was to set a particular VM port group into promiscuous mode. This will allow any VM connected to that port group to have visibility into the traffic going into or coming out of all VMs connected to that port group.
- In the latter versions of a virtual switch, the traffic flowing into and out of the port of a virtual switch (to which the monitored VM is connected) can be forward to another specific port. The target or destination port can be another virtual port or an uplink port. The

flexibility this provides is that the network monitoring application can be located either in a VM or in the physical network outside the virtualized host.

5.3. Security Recommendations for VM Traffic Monitoring

Based on the available configuration options in various hypervisor platforms, the following are some recommendations for VM Traffic Monitoring options.

VM-TM-R1: Traffic Monitoring for a VM should be applied to both incoming and outgoing traffic.

VM-TM-R2: If traffic visibility into and out of a VM is created by setting the promiscuous mode feature, care should be taken to see that this is activated only for the required VM port group and not for the entire virtual switch

VM-TM-R3: Port mirroring feature that provides choices in destination ports (either the virtual port or uplink port) facilitates the use of network monitoring tools in the physical network which are generally more robust and feature rich compared to VM-based ones.

6. Summary

With the increasing percentage of virtualized infrastructure in enterprise data centers (used for in-house applications as well as for offering external cloud services), the VMs hosting mission-critical applications becomes a critical resource to be protected. VMs just like their physical counterparts (i.e., physical servers) can be protected through host-level and network-level security measures. In the case of VMs, since they are end-nodes of virtual network, the virtual network configuration forms a critical element in their protection. Four virtual network configuration areas are considered in this publication - Network Segmentation, Network Path Redundancy, Firewall Deployment Architecture and VM Traffic Monitoring. The various configuration options under these areas are analyzed for their advantages and disadvantages and a set of security recommendations are provided

Appendix A - Acronyms

DMZ – Demilitarized Zone (A network segment created as a buffer between an enterprise's external and internal network)

DHCP – Dynamic Host Configuration Protocol

NAT – Network Address Translation

pNIC – Physical Network Interface Card

VLAN – Virtual Local Area Network

VM – Virtual Machine

vNIC – Virtual Network Interface Card

VPN – Virtual Private Network

VXLAN – Virtual Extended Local Area Network

Appendix B - Bibliography

[1] R. Chandramouli, “Analysis of Network Segmentation Techniques in Cloud Data Centers,” *2015 International Conference on Grid & Cloud Computing and Applications (GCA’15)*, Las Vegas, USA, July 27-30, 2015.

[2] R. Chandramouli**, “**Deployment-driven Security Configuration for Virtual Networks**,”** *Sixth International Conference on Networks & Communications (NETCOM - 2014),* Chennai, India, Dec 27-28, 2014.

[3] Introduction to Virtualized Networking [Web site], http://www.ipspace.net/Introduction_to_Virtualized_Networking [accessed 8/15/15].

[4] Overlay Virtual Networking [Web site], http://www.ipspace.net/Overlay_Virtual_Networking [accessed 8/15/15]

[5] Virtual Firewalls [Web site], http://www.ipspace.net/Virtual_Firewalls [accessed 8/15/15]

[6] VXLAN Technical Deep Dive http://www.ipspace.net/VXLAN_Technical_Deep_Dive [accessed 7/15/15]

[7] D. Shackleford, *Virtualization Security – Protecting Virtualized Environments*, Wiley Publishing Inc, Indianapolis, IA, USA, 2013

[8] PCI DSS : Virtualization Guidelines [accessed 9/10/15] https://www.pcisecuritystandards.org/documents/Virtualization_InfoSupp_v2.pdf

[9] *Windows Server 2012 Hyper-V: Deploying the Hyper-V Enterprise Server Virtualization Platform* – Zahir Hussain Shah, Packt Publishing Ltd, Birmingham, UK – March 2013.

[10] *Mastering VMware vSphere 6* – Nick Marshall, John Wiley & Sons, Indianapolis, Indiana, USA,2015.