

13.3.1 Virtual Networking Concepts

Click one of the buttons to take you to that part of the video.

Virtual Networking Concepts 0:00-0:32

A hypervisor presents a virtual hardware environment that's used by a virtual machine. One component of that virtual hardware is a virtual network interface card, or virtual NIC. There's no way to connect a physical network cable to a virtual NIC. So how do you connect a virtual machine to a physical network?

We're going to answer that question by describing three virtual networking concepts: virtual switches, bridging, and overlay networks. We're also going to discuss NAT and other virtual switch modes, as well as multi-homed virtual machine considerations.

Virtual Switch 0:33-1:05

Linux-based hypervisors, such as Xen, KVM, and QEMU, provide a software-based virtual network switch. A virtual network switch works just like a physical network switch. The only difference is that you don't use physical cables to connect to the switch. When you create the virtual switch in the hypervisor, you configure it with a software connection to the physical NIC of the host. When you create a VM, you configure it with a software connection to the virtual switch. If other VMs residing on the hypervisor need to communicate with this VM, you can connect them to the same switch.

Linux Bridging 1:06-1:48

Historically, a network bridge divided a network into two segments. This increased the performance of the network by isolating the traffic on each side of the bridge. Only necessary traffic was forwarded across the bridge. Eventually, multi-port bridges evolved into what is now a network switch. A Linux bridge is a virtual switch, and it behaves like a network switch. It also offers additional features, such as NAT.

To implement Linux bridging, you install the bridge-utils software package. This package contains the brctl command line utility. We can use brctl to create and configure virtual switches. GUI-based virtual machine managers provide a graphical interface to create virtual switches and connect both the host and virtual machines to them.

NAT Mode 1:49-2:52

By default, a Linux bridge or virtual switch operates in NAT mode. You may want to use this mode when there's no need for the external network to be aware of your VMs, but the VMs need access to the external network. In NAT mode, your guest VMs use your host's IP address to communicate with the external network. Computers on the external network can't initiate communications to your guest VMs because they only know the host's IP address. When your VM sends a network packet to an address on external network, your host substitutes its own IP address for the VM's IP address and sends the packet. The host keeps track of this communication. When it receives a packet from the external network that's a response to the original message, it forwards it back to the initiating VM.

When a virtual switch operates in bridged mode, all your VMs are configured with IP addresses on the same subnet as your host computer. The virtual switch in this mode acts just like a physical switch and forwards network frames accordingly. This means that all physical machines on the same physical network will be aware of your virtual machines.

Routed Mode 2:53-3:11

When a virtual switch is in routed mode, the host acts as a gateway. This mode may be useful if your virtual machines are located in a firewall's DMZ. You have to configure your VMs with IP addresses in a different subnet than the external network. This may require you to configure static routes that point to your VM's subnet.

Isolated Mode 3:12-3:31

When a virtual switch is in isolated mode, your VMs only communicate with each other. They're disconnected from the external network and even from the host computer. The only way you can access them is using the hypervisor's VM console. This mode may be useful when you're

testing the performance of a new system and don't want any external distractions.

Multi-Homed Considerations 3:32-4:04

Physical computers can have multiple NICs connected to multiple subnets to increase performance, use the computer as a network gateway, or use the computer as a network firewall. Virtual machines can also be configured as a multi-homed system with multiple virtual NICs. Depending on your requirements, you could connect each NIC to a separate virtual switch or connect multiple NICs to a single virtual switch. And, depending on your needs, you could link each physical NIC on the host machine to a different virtual switch, or you could link several physical NICs to one virtual switch.

Overlay Networks 4:05-5:21

So, you can network VMs residing on a single host machine by connecting them to the same virtual switch. But what if you wanted VMs on different host machines to be networked in a way that doesn't expose them to the external network? One solution is to implement overlay networking.

An overlay network is a virtual network that's built on top of an existing network. One way to understand how this is done is to think about network packets arranged within other packets that are assembled and dissected as they're moved over physical networks between the sender and the receiver. In this way, you can have multiple layers of abstraction. A virtual network can be overlayed on another virtual network, which can be overlayed on top of a physical network.

Overlay networking is especially important in scalable cloud technologies, where virtual resources like VMs and containers dedicated to one customer reside in the same datacenter as VMs and containers dedicated to other customers. It would be a nightmare to reconfigure physical networks to accommodate even simple customer changes.

Overlay networking makes a network more complex, and it decreases network performance. But the flexibility of adding, moving, and removing virtual assets at different physical locations often makes overlay networking very appealing and well worth the drawbacks.

Summary 5:22-5:38

That's it for this lesson. In this lesson, we covered virtual networking concepts. We described virtual switches and how they're created using the `brctl` utility. We introduced the different Linux bridging modes, including NAT mode, bridged mode, and isolated mode. We considered how multi-homed VMs are configured, and we finished by discussing overlay networking.

Copyright © 2022 TestOut Corporation All rights reserved.