13.2.4 Virtual Machine Facts

Virtual machines allow you to run multiple operating systems on the same physical hardware using a hypervisor. As a Linux administrator, you should be familiar with virtual machine concepts that are common to all hypervisor implementations.

This lesson covers the following topics:

- Virtual machine templates
- Virtual machine template formats
- Cloud and container templates
- Bootstrapping techniques
- Virtual machine disk storage
- Virtual machine management tools

Virtual Machine Templates

No matter which hypervisor you use, you can create a virtual machine very quickly by defining the hardware environment and save those settings. However, the process of installing the operating system can take several minutes and requires user input.

- The hypervisor uses your settings to instantiate, or create a new instance of, a hardware environment.
- You boot the VM in this environment.
- You install the operating system and then install the applications you need.

To avoid installing the operating system and applications every time you create a new VM, you can create a template.

- A template is a set of files that make up a VM that already has the operating system and applications installed.
- To create a new VM, copy the template to a new location and then tell the hypervisor to run the VM from there.

Virtual Machine Template Formats

Hypervisors from different vendors have different template formats.

Template Format	Description
VM	Microsoft has a tool called System Center Virtual Machine Manager, or SCVMM, that can be used to manage Microsoft Hyper-V virtual machines. Templates saved by SCVMM are called VM templates. This can be confusing, since a virtual machine is refferred to as a VM.
VMware	VMware is a hypervisor vendor. A VMware template consists of a *.vmx configuration file and *.vmdk virtual disk files. It may include other VMware-specific files.
OVF and OVA	The OVF standard was formed by an industry working group comprised of over 160 companies and organizations, and it's widely accepted. - OVF stands for open virtualization format. - OVF refers to both the packaging standard and a virtual machine package stored in an OVF package. - An OVF package consists of a directory that contains virtual machine files created with the OVF format. - Virtual machine templates can be exchanged as OVF packages. - Virtual machine templates are often sent as an OVA file, which is an OVF directory saved as a tar archive file.

Cloud and Container Templates

In the same way that a single virtual machine can be used as a template, a cloud environment, including virtual machines, storage, and networking, can be modularized as a template.

- A cloud provider can quickly instantiate a set of cloud resources using a template.
- Many cloud providers use a JSON file or a YAML file to encapsulate these templates.

A container image can also be saved as a template, which can be copied and used to create multiple containers, each running the same applications.

- The Open Container Initiative, or OCI, has published standards for container images.
- A popular container engine, Docker, has also published a standard.
- Both the OCI and Docker standards rely on the JSON file format.

Bootstrapping Techniques

One challenge with creating templates is that all virtual machines created using the same template will have the same settings. This problem can be solved with bootstrapping.

- Bootstrapping is the automated process of provisioning a virtual machine with unique settings and configurations.
- Bootstrapping customizes a virtual machine when the virtual machine first boots without requiring user input.

Two popular bootstrapping technologies that automate the provisioning of new Linux virtual machines are Cloud-init and a combination of Anaconda and Kickstart.

Bootstrapping Technology	Description
Cloud-init	Cloud-init is implemented by: ▪ Installing the cloud-init application on a Linux template ▪ Customizing the /etc/cloud/cloud.cfg file. ▪ Clone the template to create a new VM. ▪ Supply metadata (hostname, default locale, SSH host keys, etc.) by mounting a cloud-init optical drive or through a cloud-init interface. Cloud-init is also used by many cloud providers that provide a cloud-init dashboard to create custom Linux machines.
Anaconda and Kickstart	Anaconda is an installation program that's used by Fedora, Red Hat Enterprise Linux, and other distributions. ▪ Anaconda identifies the computer's hardware, creates a file system, and provides a user interface that guides the installation process. ▪ Anaconda installations can be scripted with kickstart for unattended installations. Using kickstart, a Linux administrator creates a single file containing the answers to all the questions that would be asked during a typical installation. This file can be kept on a server and read by individual computers during the Linux installation.

Virtual Machine Disk Storage

One of the configurations that can be made during virtual machine creation as well as during bootstrapping is adding virtual storage. A virtual disk is a file or set of files maintained by the host hypervisor that appears as a physical disk to the guest operating system. You can choose to allocate or provision virtual disk space in one of two ways.

Provisioning Method	Description
Thick provisioning	If you choose thick provisioning, the complete amount of storage capacity is pre-allocated on the hypervisor's physical storage device. For example, if the disk size is to be 10 GB, the full 10 GB plus any overhead is set aside for the virtual disk right from the start. This physical disk space will be unavailable for use by any other virtual machine.
Thin provisioning	Many administrators choose to use thin provisioning. A thin provisioned virtual disk consumes only the space that it needs initially, and then grows according to demand. The benefits of thin provisioning are that the disk is provisioned quickly and storage space is saved. However, the down side is that overprovisioning disks – assigning virtual disks more space than is physically available – causes the virtual machine to fail if the physical storage space becomes full. Fortunately, most hypervisors will inform you if a physical disk starts to fill up.

Another virtual disk setting offered by hypervisors is the disk mode that controls what happens when the virtual machine is shut down.

Disk Mode	Description
Persistent	Disk volumes in persistent mode behave like conventional disk drives. All data that's written to the virtual disk is permanent so that it's available when the virtual machine starts again.
Non-Persistent	When disks are configured in non-persistent mode, any data written to the disk is lost when the virtual machine is powered off. Non-persistent mode is useful for software testing or for doing demonstrations. ▪ Keep your virtual disk in persistent mode until your virtual machine is customized properly. ▪ Shut down the virtual machine and switch the virtual disk to non-persistent mode. ▪ Every time the virtual machine is shut down, the virtual disk reverts to the state it was in when non-persistent mode was set.

Virtual Machine Management Tools

There are three popular toolsets used by Linux-based hypervisors to manage virtual machines: libvirt, virsh, and vmm.

Management Tool	Description
libvirt	The libvirt software is an open-source application programming interface (API) that's used for creating, monitoring, migrating, starting, and stopping virtual machines. If you're a software developer, you can use the functions and routines provided by the libvirt API to interface with a hypervisor to create and control virtual machines.
virsh	The libvirt software also provides a command line tool named virsh for controlling virtualization. The virsh tool has many arguments that are commands in and of themselves. For example: ▪ The **virtsh list** command displays the virtual machines running on the hypervisor. ▪ The **virsh start** command starts a virtual machine. ▪ The **virsh shutdown** command cleanly shuts down a virtual machine.
vmm	Virtual Machine Manager (vmm) is a popular graphical tool for managing virtual machines on a Linux host. It's also known as virt-manager since the graphical interface can be started using the **virt-manager** command. It also comes with command line tools: ▪ The **virt-install** command is used to provision operating sytems into VMs. ▪ The **virt-clone** command clones existing VMs. ▪ The **virt-convert** command converts OVF VMs to run with libvert.