LPIC-3 305-300 Practice Questions

Shared by Wilson on 21-10-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

What is cloud-init?

Options:

A- A configuration tool for customizing cloud instances
B- A containerization tool
C- A cloud management tool
D- A virtualization platform

P2P exams

Answer:

A

Explanation:

Cloud-init is a widely used instance initialization and configuration tool designed specifically for cloud environments. According to cloud-init documentation, it is responsible for customizing cloud instances at first boot using metadata and user-provided data.

Cloud-init performs tasks such as setting hostnames, configuring networking, injecting SSH keys, installing packages, and executing user-defined scripts. It is supported by most major cloud providers and works across virtual machines regardless of the underlying hypervisor.

Cloud-init is not a containerization tool, a cloud management platform, or a virtualization solution. Instead, it operates within a virtual machine to perform configuration tasks after the instance has been provisioned.

Therefore, the correct answer is A.

P2P exams

Question 2

Question Type: MultipleChoice

Which option best network interface types are valid in an LXD container configuration? (Choose three.)

Options:

A- ipsec
B- macvlan
C- bridged
D- physical
E- wifi

Answer:

B, C, D

Explanation:

LXD supports the following network interface types in an LXD container configuration1:

macvlan: Creates a virtual interface on the host with a unique MAC address and attaches it to an existing physical interface. This allows the container to have direct access to the physical network, but prevents communication with the host and other containers on the same host2.

bridged: Connects the container to an existing bridge interface on the host. This allows the container to communicate with the host and other containers on the same bridge, as well as the external network if the bridge is connected to a physical interface3.

physical: Passes an existing physical interface on the host to the container. This allows the container to have exclusive access to the physical network, but removes the interface from the host4.

The other network interface types, ipsec and wifi, are not valid in an LXD container configuration. Ipsec is a protocol for secure communication over IP networks, not a network interface type. Wifi is a wireless technology for connecting devices to a network, not a network interface type. Reference:

About networking - Canonical LXD documentation

Macvlan network - Canonical LXD documentation

Bridge network - Canonical LXD documentation

Physical network - Canonical LXD documentation

Question 3

Question Type: MultipleChoice

Which sub-command of xl changes the media inside a virtual CD-ROM drive of a Xen guest

domain?

(Specify ONLY the sub-command without any path or parameters.)

Solution:

block-attach -- or -- xl cd-insert -- or -- cd-eject -- or -- xl cd-eject -- or -- cd-insert -- or -- xl block-attach -- or -- block-detach -- or -- xl block-detach

Determine whether the given solution is correct?

Options:

A- Correct
B- Incorrect

Answer:

B

Explanation:

According to official Xen xl toolstack documentation, the correct sub-commands used to change media in a virtual CD-ROM drive are cd-insert and cd-eject. These commands are specifically designed for inserting or ejecting ISO media from a Xen guest's virtual CD-ROM device.

While commands such as block-attach and block-detach are valid xl sub-commands, they are used for attaching and detaching block devices in general and are not specific to CD-ROM media operations. The provided solution lists multiple commands, including incorrect and unrelated ones, rather than specifying the correct sub-command precisely.

Because the solution does not accurately and exclusively identify the correct xl sub-command, it is considered incorrect. Therefore, the correct determination is B.

Question 4

Question Type: MultipleChoice

What is the primary benefit of using Packer for creating machine images?

Options:

A- It provides real-time monitoring of virtual machines
B- It ensures consistency between development and production environments
C- It speeds up the development of application code
D- It simplifies the deployment of virtual machines

Answer:

B

Explanation:

Packer is an image automation tool whose primary benefit is ensuring consistency across environments. According to virtualization and DevOps documentation, Packer allows administrators to create identical machine images for development, testing, and production from a single configuration.

By using the same base image and build process, Packer eliminates configuration drift and ensures that environments behave consistently. This significantly reduces deployment issues caused by environmental differences and supports Infrastructure as Code best practices.

Packer does not monitor virtual machines or directly speed up application development. While it can indirectly simplify deployments, its main advantage is image consistency and repeatability.

Therefore, the correct answer is B.

Question 5

Question Type: MultipleChoice

What is oVirt?

Options:

A- An extension to the Linux Kernel used to provide container virtualization similar to LXC and OpenVZ
B- A comprehensive management infrastructure for Linux-based virtualization
C- An approach used to eliminate the need for virtualization called Zero-Virt
D- A Linux-based hypervisor similar to KVM and Xen
E- A library that provides access to several different virtualization technologies in a common manner

Answer:

B

Explanation:

oVirt is an open-source virtualization management platform that provides a comprehensive infrastructure for managing Linux-based virtualization environments. According to official oVirt and Red Hat documentation, oVirt acts as a management layer on top of KVM, offering centralized control over hosts, virtual machines, storage, and networking.

oVirt itself is not a hypervisor; instead, it relies on KVM as the underlying virtualization technology. It provides both a web-based management interface and REST APIs, enabling administrators to manage large-scale virtualization deployments efficiently. Features include VM lifecycle management, live migration, high availability, storage domain management, and integration with directory services.

Option E describes libvirt, not oVirt. Options A, C, and D do not match oVirt's documented purpose. Therefore, the correct answer is B.

Question 6

Question Type: MultipleChoice

Which option best tools is used to interact with XenStore?

Options:

A- xendo
B- oxs
C- xstore
D- xentore-ls
E- xl store

Answer:

D

Explanation:

XenStore is a shared configuration and state database used by Xen domains. According to Xen documentation, tools such as xenstore-ls, xenstore-read, and xenstore-write are used to interact directly with XenStore.

Although option D contains a minor typographical error (''xentore-ls'' instead of xenstore-ls), it clearly refers to the correct and documented utility. None of the other options represent standard XenStore interaction tools.

Therefore, despite the typo, D is the correct answer.

Question 7

Question Type: MultipleChoice

What kind of virtualization is implemented by LXC?

Options:

A- System containers
B- Application containers
C- Hardware containers
D- CPU emulation
E- Paravirtualization

Answer:

A

Explanation:

LXC implements system containers, which are a type of operating-system-level virtualization. System containers allow running multiple isolated Linux systems on a single Linux control host, using a single Linux kernel. System containers share the same kernel with the host and each other, but have their own file system, libraries, and processes. System containers are different from application containers, which are designed to run a single application or service in an isolated environment. Application containers are usually smaller and more portable than system containers, but also more dependent on the host kernel and libraries. Hardware containers, CPU emulation, and paravirtualization are not related to LXC, as they are different kinds of virtualization methods that involve hardware abstraction, instruction translation, or modification of the guest operating system. Reference:

1: LXC - Wikipedia

2: Linux Virtualization : Linux Containers (lxc) - GeeksforGeeks

3: Features - Proxmox Virtual Environment

Question 8

Question Type: MultipleChoice

Which utility is used to create and manage virtual machine disk images in Linux?

Options:

A- Disk utility
B- VirtualBox Manager
C- Libvirt Disk Manager
D- qemu-img

Answer:

D

Explanation:

In Linux-based virtualization environments, qemu-img is the standard utility used to create, convert, inspect, and manage virtual machine disk images. Virtualization and containerization documentation clearly identifies qemu-img as a core component of the QEMU/KVM virtualization stack. It supports multiple disk image formats such as RAW, QCOW2, VMDK, and VHD, making it highly versatile across different hypervisors and platforms.

The qemu-img tool allows administrators to perform critical disk operations, including creating new disk images, resizing existing images, converting between formats, and checking disk integrity. For example, it is commonly used to create QCOW2 images, which support advanced features like copy-on-write, snapshots, and compression.

Other options listed are incorrect because Disk Utility is a general-purpose storage management tool, VirtualBox Manager is specific to Oracle VirtualBox, and Libvirt Disk Manager is not a recognized standalone utility. Libvirt relies on qemu-img internally for disk image handling.

Official virtualization notes consistently reference qemu-img as the primary disk image management tool in Linux virtualization environments, especially when working with KVM and libvirt-based systems.

Question 9

Question Type: MultipleChoice

Which option best tasks is performed by Vagrant?

Options:

A- It automates the installation of a virtual machine according to a configuration file describing the desired VM
B- It monitors the functionality of a virtual machine and restarts the VM in case of failure
C- It is a programming interface used to create reports from collected performance data
D- It is a hypervisor optimized for embedded ARM systems
E- It migrates virtual machines automatically between host systems

Answer:

A

Explanation:

Vagrant is a tool designed to automate the creation and provisioning of development virtual machines using a declarative configuration file called a Vagrantfile. According to virtualization documentation, Vagrant simplifies setting up reproducible environments by automatically installing and configuring virtual machines based on this file.

Vagrant does not perform monitoring, reporting, hypervisor functions, or automatic VM migration. It integrates with providers such as VirtualBox, KVM, and VMware but is not itself a hypervisor.

Therefore, the correct answer is A.

Question 10

Question Type: MultipleChoice

Which option best mechanisms are used by LXC and Docker to create containers? (Choose three.)

Options:

A- Linux Capabilities
B- Kernel Namespaces
C- Control Groups

D- POSIXACLs
E- File System Permissions

Answer:

A, B, C

Explanation:

LXC and Docker are both container technologies that use Linux kernel features to create isolated environments for running applications. The main mechanisms that they use are:

Linux Capabilities: These are a set of privileges that can be assigned to processes to limit their access to certain system resources or operations. For example, a process with the CAP_NET_ADMIN capability can perform network administration tasks, such as creating or deleting network interfaces. Linux capabilities allow containers to run with reduced privileges, enhancing their security and isolation.

Kernel Namespaces: These are a way of creating separate views of the system resources for different processes. For example, a process in a mount namespace can have a different file system layout than the host or other namespaces. Kernel namespaces allow containers to have their own network interfaces, process IDs, user IDs, and other resources, without interfering with the host or other containers.

Control Groups: These are a way of grouping processes and applying resource limits and accounting to them. For example, a control group can limit the amount of CPU, memory, disk I/O, or network bandwidth that a process or a group of processes can use. Control groups allow containers to have a fair share of the system resources and prevent them from exhausting the host resources.

POSIX ACLs and file system permissions are not mechanisms used by LXC and Docker to create containers. They are methods of controlling the access to files and directories on a file system, which can be applied to any process, not just containers.

LXC vs Docker: Which Container Platform Is Right for You?

LXC vs Docker: Why Docker is Better in 2023 | UpGuard

What is the Difference Between LXC, LXD and Docker Containers

lxc - Which container implementation docker is using - Unix & Linux Stack Exchange

Question 11

Question Type: MultipleChoice

Which of the following statements in a Dockerfile leads to a container which outputs hello world? (Choose two.)

Options:

A- ENTRYPOINT 'echo Hello World'
B- ENTRYPOINT ['echo hello world']
C- ENTRYPOINT ['echo', 'hello', 'world']
D- ENTRYPOINT echo Hello World
E- ENTRYPOINT 'echo', 'Hello', 'World*

Answer:

B, C

Explanation:

The ENTRYPOINT instruction in a Dockerfile specifies the default command to run when a container is started from the image. The ENTRYPOINT instruction can be written in two forms: exec form and shell form. The exec form uses a JSON array to specify the command and its arguments, such as [''executable'', ''param1'', ''param2'']. The shell form uses a single string to specify the command and its arguments, such as ''executable param1 param2''. The shell form is converted to the exec form by adding /bin/sh -c to the beginning of the command. Therefore, the following statements in a Dockerfile are equivalent and will lead to a container that outputs hello world:

ENTRYPOINT [''echo hello world''] ENTRYPOINT [''/bin/sh'', ''-c'', ''echo hello world''] ENTRYPOINT ''echo hello world'' ENTRYPOINT [''echo'', ''hello'', ''world''] ENTRYPOINT [''/bin/sh'', ''-c'', ''echo'', ''hello'', ''world''] ENTRYPOINT ''echo hello world''

The other statements in the question are invalid or incorrect. The statement A. ENTRYPOINT ''echo Hello World'' is invalid because it uses double quotes to enclose the entire command, which is not allowed in the shell form. The statement D. ENTRYPOINT echo Hello World is incorrect because it does not use quotes to enclose the command, which is required in the shell form. The statement E. ENTRYPOINT ''echo'', ''Hello'', ''World'' is invalid because it uses double quotes to separate the command and its arguments, which is not allowed in the exec form. Reference:

Dockerfile reference | Docker Docs

Using the Dockerfile ENTRYPOINT and CMD Instructions - ATA Learning

Difference Between run, cmd and entrypoint in a Dockerfile

To Get Premium Files for 305-300 Visit

https://www.p2pexams.com/products/305-300

For More Free Questions Visit

https://www.p2pexams.com/lpi/pdf/305-300

