“Kyiv Vocation College of Communication”
Cyclical Commission of Computer Engineering

REPORT ON EXECUTION
LABORATORY WORK №1

on the discipline: "Operating Systems"

Topic: “Introduction to the working environment of virtual machines and features of the Linux operating system”

Performed by student
group RPZ-13a
Kateryna Hranat
Checked by teacher
V.S. Sushanova

Performed by student group RPZ-13a Kateryna Hranat

Kyiv 2024

Work Objectives:

1. Familiarization with hypervisors of different types and virtualization when working with operating systems.
2. Introduction to the main types of modern operating systems, a brief overview of their capabilities.

Material Support for Classes:

1. Personal computer of IBM PC type.
2. Operating system from the Windows family (Windows 7).
3. Virtual machine - VirtualBox (Oracle).
4. GNU/Linux operating system - CentOS.
5. Cisco Networking Academy website netacad.com and its online courses on Linux.

Performed by student group RPZ-13a Kateryna Hranat

Tasks for Preliminary Preparation:

1. Read the brief theoretical information for the laboratory work and create a small dictionary of basic English terms related to the classification of virtual environments.

English terms	Ukrainian terms
virtual-machine	віртуальна машина
shared hosting	спільний хостинг
dedicated hosting	виділений хостинг
operating systems	операційні системи
machine simulators	машинні симулятори
binary translation	бінарний переклад
distributions	дистрибуції
graphical user interface	графічний інтерфейс користувача
hardware	апаратне забезпечення
command line interface	командний рядок
applications	застосунки
software	програмне забезпечення

2. Reading the material from the brief theoretical information, answer the following questions:

   2.1. Characterize the concept of "hypervisor." What are their types?

The concept of a "**hypervisor**" refers to a software layer that enables the virtualization of multiple operating systems on a single physical host. There are two main types of hypervisors:

Type 1 Hypervisor (Bare Metal Hypervisor): This type runs directly on the host's hardware to control the hardware and to manage guest operating systems. It doesn't require a host operating system and is considered more efficient.
Type 2 Hypervisor (Hosted Hypervisor): This type runs on a conventional operating system just like other computer programs. It hosts guest operating systems and is generally easier to set up but may have slightly lower performance compared to Type 1.

2.2. List the main components and capabilities of hypervisors according to your variant (journal serial number), Table 1.

Variant	1, 6, 11, 16, 21	2, 7, 12, 17, 22	3, 8, 13, 18, 23	4, 9, 14, 19, 24	5, 10, 15, 20, 25
Hypervisor	VirtualBox	VMware	Xen	KVM	Hyper-V

Variant 4 - KVM
KVM, which stands for Kernel-based Virtual Machine, is a virtualization infrastructure for the Linux kernel. It allows you to turn Linux into a hypervisor, enabling it to run multiple virtual machines (VMs) concurrently. Here are the main components and capabilities of KVM:

1. **Kernel Module (kvm.ko):** KVM is implemented as a kernel module that provides the core virtualization infrastructure. It leverages hardware virtualization extensions (Intel VT-x and AMD-V) to enhance performance.

2. **QEMU (Quick Emulator):** QEMU is a user-space component that works in conjunction with KVM to emulate the hardware of the virtual machines. It handles the I/O emulation, device models, and provides a user-friendly interface for managing VMs.

3. **/lib/modules/kernel-version/kernel/arch/x86/kvm/:** This directory contains the architecture-specific KVM module (e.g., x86 in the path) and is loaded into the kernel during runtime.

4. **Kernel Virtual Machine (KVM):** KVM itself acts as a module that extends the Linux kernel to support hardware-assisted virtualization.

5. **CPU and Memory Management:** KVM manages CPU and memory resources efficiently, allowing multiple VMs to run on the same physical host. It uses the host's memory management capabilities and leverages hardware virtualization extensions for improved performance.

6. **Hardware Virtualization Extensions (Intel VT-x and AMD-V):** KVM relies on these hardware extensions to accelerate virtualization, providing direct support for virtual machine execution.

7. **Linux Kernel Integration:** KVM is tightly integrated into the Linux kernel, allowing it to take advantage of the kernel's security, scheduling, and resource management features.

8. **APIs and Interfaces:** KVM provides APIs and interfaces for interacting with and managing virtual machines. Tools like virsh, libvirt, and other management tools can be used to create, configure, and control VMs.

9. **Live Migration:** KVM supports live migration, allowing virtual machines to be moved from one host to another without downtime.

10. **Snapshot and Cloning:** KVM allows you to create snapshots of virtual machines, capturing their state at a specific point in time. Cloning enables the replication of VMs for quick deployment.

11. **Nested Virtualization:** KVM supports nested virtualization, allowing you to run virtual machines inside virtual machines.

Overall, KVM provides a robust and flexible virtualization platform for Linux, combining the advantages of hardware-assisted virtualization with the power and stability of the Linux kernel.

3. Study the materials of the online course "NDG Linux Essentials" from Cisco:

- Chapter 1 - Introduction to Linux
- Chapter 2 - Operating Systems

Complete✔

4. Complete testing in the NDG Linux Essentials course on the following topics:

Chapter 02 Exam

Complete✔

5. Prepare an initial version of the report in electronic form:

- Title page, topic, and purpose of the work
- Glossary of terms
- Answers to points 2.1 and 2.2 from the tasks for preliminary preparation

Complete✔

Performed by student group RPZ-13a Kateryna Hranat

Progress of Work:

Watch introductory videos and demonstration materials on the following topics:
1.1. GNU/Linux. Basic Information.
Access: 1.Linux для начинающих. Введение
1.2. Installing CentOS in VirtualBox.
Access: 2.Linux для начинающих. Установка CentOS в VirtualBox
1.3. Installing CentOS in Text Mode.
Access: Установка CentOS 6 в текстовом режиме
1.4. Installing the Gnome Desktop Environment in CentOS.
Access: Установка окружения рабочего стола Gnome в CentOS 6
1.5. Installing the KDE Desktop Environment in CentOS.
Access: Установка окружения рабочего стола KDE в CentOS 6
1.6. The Shell (Linux).
Access:
https://drive.google.com/open?id=0B0PV0_SM0LoDSVNPWUVRdUxaN2s
1.7. Overview of Linux Graphical Shells.
Access: Обзор оболочек Linux

After watching the videos, answer the following questions:

2.1. List the steps for deploying an operating system based on a VirtualBox virtual machine.
Steps for deploying an operating system on a VirtualBox virtual machine:

1. Booting up the VirtualBox virtual machine.
2. Creating a new virtual machine.
3. Selecting the operating system for installation.

4. Configuring the virtual machine settings (memory, disk size, processor, etc.).
5. Booting from the operating system image or inserting the appropriate media (CD/DVD or ISO file).
6. Installing the operating system by following the on-screen instructions and prompts.
7. Configuring additional parameters post-installation (optional).
8. Starting the virtual machine and verifying the correctness of the deployment.

2.2. Are there any hardware limitations when installing 32-bit and 64-bit OS?
Hardware limitations when installing 32-bit and 64-bit operating systems:

1. To install a 32-bit operating system, a processor supporting the x86 architecture is required.
2. To install a 64-bit operating system, a processor supporting the x86-64 architecture (AMD64 or Intel 64) is required.

2.3. What are the main steps when installing CentOS in text mode?
Main steps for installing CentOS in text mode:

1. Launching the installation from the installation media or image.
2. Selecting the installation mode (text mode).
3. Choosing language and regional settings.
4. Configuring the network (optional).
5. Selecting the disk for installation and partitioning its space.
6. Choosing the packages to install.
7. Setting up the user password and other basic parameters.
8. Completing the installation and rebooting the system.

2.4. How can you install the Gnome and KDE graphical shells on CentOS if it's already installed in text mode (specify necessary commands and packages)?
Встановлення графічних оболонок Gnome та KDE на CentOS:

Для встановлення Gnome:

```
sudo yum groupinstall "GNOME Desktop"
```

Для встановлення KDE:

```
sudo yum groupinstall "KDE Plasma Workspaces"
```

2.5. Provide a brief overview of the graphical interfaces used in various Linux distributions according to your variant (journal serial number).

Variant	1, 4, 7, 10, 13, 16, 19, 22, 25	2, 5, 8, 11, 14, 17, 20, 23	3, 6, 9, 12, 15, 18, 21, 24
Graphical user interface	KDE and Fluxbox	Gnome and JWM	Xfce and Fvwm

Variant 4 - KDE and Fluxbox

KDE:
KDE (K Desktop Environment) is a powerful and full-featured desktop environment for Linux and other UNIX-like operating systems.
It offers many features including desktop widgets, taskbar, file manager, user accounts, system settings, etc.
It provides a high level of customization, allowing users to change the appearance and behavior of almost all aspects of the interface.
Popular Linux distributions such as Kubuntu, openSUSE, Mageia, use KDE as their primary desktop environment.

Performed by student group RPZ-13a Kateryna Hranat

Fluxbox:

Fluxbox is a lightweight and fast window manager for UNIX-like systems.
It has a minimalist design and limited functionality compared to larger desktop environments like KDE or GNOME.
Known for its speed and efficiency, particularly on older or less powerful computers. Provides a basic set of features including window management, support for virtual desktops, and customization of keyboard shortcuts.
Typically used in lightweight Linux distributions such as Damn Small Linux, antiX, or as an alternative window manager in distributions with KDE or GNOME.

Control questions

1. Compare type 1 and type 2 hypervisors, what is the difference between them and their areas of application?

Type 1 offers better performance and security as it interacts directly with the hardware. Type 2 is more convenient for development and testing but adds an additional layer of virtualization that may impact performance.

2. Explain the concept of "GNU GPL," what is its main idea?

This is a software license that provides free access to the software's source code, allowing modification and distribution.

3. What is the essence of open-source software?

It is software available for viewing, modification, and distribution according to the requirements of an open license. The main idea is to promote collaborative development and cooperation among developers.

4. What is a distribution?

It is a set of software based on an operating system, including the OS kernel, system utilities, software packages, and other necessary software.

5. What system administration tasks can be implemented based on the Linux OS?

Network configuration, user and group management, file system management, system resource monitoring, security configuration, etc.

6. How are the OS Android and Linux related to each other?

Android is based on the Linux kernel. However, Android uses its own runtime environment, distinct from traditional Linux systems.

7. The main features and usage area of Embedded Linux?

Embedded Linux refers to embedded systems based on the Linux kernel used in devices such as mobile phones, routers, TVs, cars, etc. They can operate in limited resources and are known for their high stability and reliability.

8. How can you change the boot type of Linux: in text mode (runlevel 3) or graphical (runlevel 5)? What are the differences between CLI and GUI modes?

This can be done by modifying boot parameters in the GRUB configuration file (/boot/grub/grub.cfg). For text mode, you can change the "linux" parameter to "text", and for graphical mode, leave the "linux" parameter unchanged. The CLI (Command Line Interface) mode allows interaction with the system through the command line, while the GUI (Graphical User Interface) mode provides a graphical user interface for interaction with the system.

Conclusions:

While completing laboratory work №1, I familiarized myself with hypervisors of various types, virtualization when working with operating systems, the main types of modern OS, and briefly reviewed their capabilities.

Performed by student group RPZ-13a Kateryna Hranat

Theoretical exploration delved deeper into the concepts of hypervisors and their types, the Linux system, the VirtualBox virtual machine, CentOS, graphical interfaces like Gnome and KDE, among others. Additionally, I practiced my English language skills in writing the report. I gained a lot of new information.