“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної та програмної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №2**

з дисципліни: «Операційні системи»

**Тема: «Знайомство з інтерфейсом та можливостями ОС Linux»**

Виконали студенти

групи КСМ-03Б

Команда: Кучмій-Зікеєв

М.О., Зінченко І.О. та Звєрьков А.Г.

Перевірив викладач

Сушанова В.С.

Київ 2022

**Мета роботи:**

1. Знайомство з інтерфейсами ОС Linux.

2. Отримання практичних навиків роботи в середовищах ОС Linux та мобільної

ОС — їх графічною оболонкою, входом і виходом з системи, ознайомлення зі структурою робочого столу, вивчення основних дій та налаштувань при роботі в системі.

**Матеріальне забезпечення занять**

1. ЕОМ типу IBM PC.

2. ОС сімейства Windows (Windows 10).

3. Віртуальна машина – Virtual Box (Oracle).

4. Операційна система GNU/Linux – CentOS.

5. Сайт мережевої академії Cisco netacad.com та його онлайн курси по Linux

**Завдання для попередньої підготовки.**

***Готував матеріал студент Звєрьков А.***

1.

|  |  |
| --- | --- |
| Command | Explanation |
| pwd | Use the pwd command to find out the path of the current working directory (folder) you’re in. |
| cd | Navigates through the Linux files and directories, use the cd command. |
| ls | The ls command is used to view the contents of a directory. |
| cat | One of the most frequently used commands in Linux. It is used to list the contents of a file on the standard output (sdout). |
| mv | The primary use of the mv command is to move files, although it can also be used to rename files. |
| mkdir | Use mkdir command to make a new directory — if you type mkdir Music it will create a directory called Music. |
| rmdir | If you need to delete a directory, use the rmdir command. However, rmdir only allows you to delete empty directories. |
| locate | You can use this command to locate a file, just like the search command in Windows. |
| find | Similar to the locate command, using find also searches for files and directories. |
| grep | It lets users search through all the text in a given file. |
| sudo | Short for “SuperUser Do”, this command enables you to perform tasks that require administrative or root permissions. |
| df | Use df command to get a report on the system’s disk space usage, shown in percentage and KBs. |
| kill | Terminates program if it is unresponsive. |
| ping | Use the ping command to check your connectivity status to a server. |
| history | Shows the history of previously used commands. |
| man | Shows instructions how to use CLI |
| zip, unzip | Use the zip command to compress your files into a zip archive, and use the unzip command to extract the zipped files from a zip archive. |
| useradd | useradd is used to create a new user, while passwd is adding a password to that user’s account. |

2. Дати визначення термінам:

• CLI (Command Line Interface) mode — is a command line program that accepts text input to execute operating system functions. However, CLI is still used by software developers and system administrators to configure computers, install software, and access features that are not available in the graphical interface. The command line remains an important tool for IT professionals, software developers, sys admins, network administrators and many others who prefer a more precise and reproduceable interface to their systems.

• GUI (Graphical User Interface) — a type of user interface through which users interact with electronic devices via visual indicator representations. The actions in a GUI are usually performed through direct manipulation of the graphical elements. There are many different GUIs and each has its own features and functions.

• Virtual Terminals are similar to Terminal. The only difference is that you cannot use the mouse with the Virtual Terminals. Therefore, you need to know the keyboard shortcuts. Virtual Terminals enable a number of users to work on different programs at the same time on the same computer. This is the reason they are one of the most distinguished features of Linux.

**Хід роботи.**

1.1 ***Готував матеріал студент Звєрьков А.***

1.2 ***Готував матеріал студент Кучмій-Зікеєв М.***

***1.3 Готував матеріал студент Кучмій-Зікеєв М.***

2. ***Готував матеріал студент Зінченко І.***

**2. Work in a mobile OS environment**.

***2.1. Describe the main menu of your mobile OS, what GUI does it use?***

The iOS user interface is based on direct manipulation using Multi-Touch gestures. An external accelerometer is used by several shake response applications, which also follows the same cancellation order or rotation of the 3D instrument, which is a common transfer order between book mode and landscape mode. , IOS also differs from its competetors in the following factors:

• Fast update. iPhone and iPad customers don't have to wait for developers to provide updates to host them after a new version of iOS is released. Many customers have yet to purchase iOS 4.2, which Safari announced last year. IOS applications are released frequently and become available for all devices with the necessary compatibility.

• Quality of the program. The app is almost always more attractive and comfortable on iPhone and iPad. For example, you can compare two types of comics: compare the appearance of the icons with the same application, which looks the same: Android icons look like "collective farms".

• Your own ecosystem. All devices sync with each other, with an iCloud account required when registering a device and enhancing its features.

***2.2. Describe the mobile phone component settings menu.***

Our mobile phone uses many components to ensure the comfort it offers. It is not only about the software and the way of using the iOS, but also about the components that make up the hardware. Mobile phones can be very different from each other. iOS As an operating system, it is a very important part of it, given its ability to customize and the possibilities it offers to make the different experiences of all users unique. However, it would be a mistake to focus on the operating system as the only differentiator, since software is only half the equation. The second half, no less important, is hardware. CPU, GPU, RAM, internal storage, connectivity... Most generational jumps are justified based on the hardware and the improvements they offer. For most people, that means bigger numbers for the camera or other memory, but for manufacturers it's important to consider the hardware elements and their compatibility with the software. The IOS interface is standardized and does not change from one version to another. If you set it up the way you want, you won't succeed. You can change a little, unless you escape from prison, which will be discussed below.

***2.3. Using keyboard shortcuts to perform special actions.***

If you use the Magic Keyboard with iPhone, you can use keyboard shortcuts to activate VoiceOver commands.You can also use VoiceOver Help to learn about the keyboard layout and the actions associated with different key combinations. VoiceOver Help speaks keys and keyboard commands as you type them without performing an associated action.

Enable help - VoiceOver VO-K Disable Help - VoiceOver Esc (Escape) Select next or previous item - VO-Right Arrow or VO-Left Arrow Activate the selected item - VO-Spacebar Touch and hold the selected item - VO-Shift-M Reading elements from the current position - VO-A Reading the screen from above - VO-B Pause or resume reading – Controls Copy the last spoken text to the clipboard - VO-Shift-C Text search - VO-F Mute or unmute VoiceOver - VO-S Go to home screen - VO-H Go to status bar - VO-M Open the App Switcher - VO-H-H Open the Item Selector - VO-I Change the label of the selected item - VO-/ Start, stop or pause an action - VO-Hyphen

***2.4. Sign in and shut down the device. Features of battery power settings.***

Login to the system is done using Face ID or Touch ID, depending on the version of the phone. Or, if the phone is completely new, personal settings are made for each user, and data is transferred from the previous phone to the new one. The operation is terminated by pressing and holding down one power key located on the right side of the phone, after pressing and holding this key, a small menu is displayed on the screen with options to turn off the phone or call an emergency ambulance. iPhone has an internal lithium-ion rechargeable battery that keeps your device running optimally for the time being. Compared to traditional batteries, lithium-ion batteries are lighter, charge faster, last longer and have a higher specific capacity for longer battery life. The battery icon in the upper right corner indicates the battery level or charging status. If iPhone is syncing or in use while charging, it may take longer to charge the battery.

If your iPhone's battery is very low, a low-battery image may be displayed, indicating that your iPhone needs to be charged for up to 10 minutes before it can be used. If your iPhone's battery is extremely low when you start charging it, the display may be blank for up to 2 minutes before the low-battery image appears. The iPhone has a setting that helps extend battery life by shortening the time it takes to fully charge. This setting uses machine learning to learn how you charge your device every day, and then expects to finish charging at 80% until you need to charge your device again.

Go to Settings > Battery, then tap Battery health.

Turn on "Optimized battery charging".Low Battery Mode reduces iPhone power consumption when the battery is low. It optimizes performance for basic tasks like making and receiving calls; sending and receiving emails and messages; Internet access, etc. On models with ProMotion display technology, the low charge mode limits the display refresh rate to 60 frames per second. In low battery mode, iPhone may perform some tasks more slowly.

**Відповіді на контрольні запитання**

***Готував матеріал студент Зінченко І.***

***1. Give examples of Linux server applications for database servers, messaging servers, and***

***file sharers.***

A server operating system is an operating system specifically designed to work on servers (I'm stating the obvious today). Is about a software layer on top of which other software programs or applications can run on server hardware. In other words, it works like a desktop operating system, a mobile device operating system, or a device's embedded operating system. Its role is to allow software programs and applications to run on their respective devices.

Among the Linux offerings, the most widely used is Red Hat, which is sold on a subscription model. Red Hat Enterprise Linux Server

As I mentioned above, Red Hat is available on a subscription model that provides access to software, technical support, and updates. However, if you want to explore the Linux operating system most commonly used on servers and its tools, you can do so for free by subscribing to your developer portal.

Debian

Debian is one of the best Linux distributions. It has very comprehensive repositories, an excellent package management system, and a development process that ensures that every stable version they release is truly stable.

I haven't found any reliable statistics to back up what I'm about to say, so maybe I'm terribly wrong. I get the impression that Debian on servers is not as popular as it should be. I think you should see that unlike Ubuntu they don't release a proper version for servers. During installation, the user decides whether to install desktop or server applications.

Ubuntu server

La version for Ubuntu servers or It offers the reliability of Debian along with the integration of technologies for the cloud and virtualization. In addition, the use of the Snap package format and the LivePatch service, which allows you to update the kernel without the need for a reboot, reduce equipment downtime. Although the distribution is free, you can contract for Canonical technical support. Describe the process of installing Ubuntu Server 10.10. I won't, since it is not more complicated, and maybe even easier, than installing Windows. The only thing I will ask you to do during the installation process is to select the installation of "OpenSSH Server" when choosing the software to be installed. This is so that in the future you can connect to this computer remotely using an ssh client (for example, PuTTY). And at the same time, you will have full access to the operating system from any computer in your network, since a monitor is usually attached to the server and the server itself is located in a specially designated server room (server room).

In the "User account settings and passwords" window, enter your login and set a password for it.

(adsbygoogle = window.adsbygoogle || []).push({});

After installation, you boot up for the first time, and the OS control console appears before your eyes. To begin, enter the login, press Enter, and then the password in order to enter the OS itself. You will be logged in as the newly created user who does not have root superuser privileges

***2. Compare the Bourne shell, C, Bourne Again (Bash), the tcsh, Korn shell (Ksh), and zsh.***

***Bash***

The original UNIX shell was called the Bourne shell. It was distributed under an AT&T license, and was one of the first in the GNU project to allow users to create their own versions. This led to the appearance of BASH, the Bourne-Again Shell, which has now become the de facto standard in the Linux world and is installed in most distributions. Therefore, if you have ever used a shell in Linux, it was most likely BASH.

Bash will usually cover all your needs, unless of course you have a bunch of scripts written for another shell.

***KSH (Korn Shell)***

David Korn of Bell Labs created his shell in the distant eighties. It started as closed software, but later releases began to be released under the Eclipse Public license.

Some users prefer ksh, highlighting its advantages such as cleaner exit codes, input pipes (from pipes), cool syntax for loops, easier repetition of commands and work with associative arrays, the ability to emulate the behavior of emacs and vi. Otherwise, it is very similar to bash, although it has its own subtleties.

***ZSH***

Zsh was written in the early 1990s and was similar to bash and ksh, but had unique features such as spell checking, design themes, command history synchronization between multiple terminals, and other nice features.

It is distributed under an MIT-like license, but some parts are distributed under the GPL.

***TCSH***

TCSH was written based on csh, the Berkeley Unix C shell. It is one of the oldest shells with an interesting history dating back to the early days of UNIX.

One of the most interesting features is the scripting language, whose syntax is similar to C. It should also be noted that this is not its only advantage, there is also adding arguments to aliases, cool autocompletion and convenient settings out of the box.

***C***

Most system programs in UNIX are written in C, and all calls are C functions. Since the details of the calls are known only to the compiler, this makes portability a major feature of UNIX programs. System calls for manipulating files are, for example: creat, open, read, write, close, link, unlink and trunc. A new process is created by calling fork. A process is a copy of the process space of the parent process that calls the fork. Signals are used for exception handling. There are 20 different signals in UNIX. For example, an interrupt signal is used to interrupt a command until execution is complete, usually by the user. Disadvantages of the command line: Some tasks are difficult or impossible to do from the command line. Tasks that require a lot of typing, such as long file names, can be annoying and tiring without minimal support for clipping.

***Bourne shell***

The Bourne shell is the default shell for UNIX version 7. Originally developed by Stephen Bourne at Bell Labs, it was a replacement for the Thompson sheath. The Bourne shell began its journey in 1979. Read more about the history of the Bourne shell on Wikipedia.

Even today, the Bourne shell is available on almost all Linux systems. The location of the binary is as follows:

$ some

UNIX-like systems continue to have the Bourne shell /usr/bin/sh. This may not be a Bourne shell, but a symbolic/hard reference to a more compatible shell. You can check this by running the following command:

$ file-h/trash/sh

The shell is also the name of a command-line programming language. This language is described by the POSIX standard, which all UNIX and UNIX-like systems must adhere to. The Born shell, which we have discussed so far, is one of the realizations of

***3. What is a package manager for? What package managers do you know in Linux?***

A package manager is a tool that is commonly used by Linux developers and users. With its help, users can simplify downloading applications — they do not need to search for the installation file on the developer's site, download it first, and only then install the application.

***DPKG (Debian Package)*** is a package management system in Debian and ego-based distributions, such as Ubuntu.

The DPKG utility appeared in the Debian distribution in 1995. The low-level package manager is designed only to work with local DEB packages and cannot independently resolve dependencies or download packages from repositories.

Features

Supports adding architectures from other Linux distributions.

DPKG works only with local packages.

More than 55,000 packages have been released under the DEB architecture.

DEB packages are archives with a set of installation files. To install the necessary program from the repositories into the system, a high-level APT package manager is created, which works in parallel with DPKG. PT (Advanced Packaging Tool) is a console utility that performs the role of a "search engine" and package loader from repositories. Installation of downloaded packages is performed by the DPKG utility. Thanks to efficient dependency resolution, the APT package manager is used by default in distributions with Debian architecture and keeps the system up to date.

The list of repositories is stored in the file "/etc/apt/sources.list" and can be changed by the user at any time to install or update a program that is not part of the distribution base. Installation of downloaded packages is performed by the DPKG utility.

Initially, APT was developed only to work with DEB packages used in Debian and related OSes (Ubuntu, Linux Mint). Later, support for rpm files was added to it. Thanks to this, it is possible to install software in the usual way even in RED HAT distributions and its derivatives (Fedora, CentOS, etc.).

APT shells

To simplify work with APT, you can use console shells APTITUDE or Synaptic.

***APTITUDE***

APTITUDE is a utility that acts as an "add-on" for APT. The developers of the program added useful functions that optimize the package search system, as well as fixed errors related to the resolution of dependencies.

***Synaptic*** is a graphical package manager based on APT. The program is useful for beginners who are not familiar with the command line. Despite the simplicity of the interface, the utility provides all the necessary functionality of the APT package manager (installation, deletion, update and search of packages).

***RPM (Red Hat Package Manager)*** is a package format and a low-level package manager for RED HAT systems (RHEL, CentOS, Fedora, etc.) Like DPKG, it can only work with local files.

The package manager was released in 1997. It works with RPM packages. Unlike DEB, RPM packages are archived by the cpio utility, which compresses the package using the gzip algorithm.

Features

The program is updated in an accelerated mode, thanks to the replacement of only the elements of the package edited by the developer.

For downloading, updating packages, as well as resolving dependencies, you will have to use higher-level package managers (YUM, DNF).

Since 2010, packets are signed with an MD5 hash. This eliminates the possibility of an attacker modifying the RPM file to inject virus code.

***YUM (Yellowdog Updater, Modified)*** is a high-level package manager written in Python for RED HAT systems (RHEL, CentOS, Fedora). The program is a peculiar shell for the RPM utility.

The task of YUM includes downloading and updating packages from repositories, as well as satisfying dependencies during program installation.

***4. What security tools are used in Linux?***

There are distributions loaded with hundreds or more than a thousand tools for security and for conducting audits, such as the well-known Kali Linux, DEFT, Parrot Security, etc., we have already dedicated different articles to them and you should know them well, but now we will talk about other tools , which, far from being specific to security experts, they can help us improve our own daily work, are all available for the desired distribution ...

That is, we are going to expose a list of tools that can help us perform tasks that will improve the security of our network or system:

***CIRClean:*** a tool for USB drives that can help us clean up documents on these types of drives.

Buttercup: A cross-platform password manager to help us protect our passwords and remember them.

***KeePaasXC:*** This is another tool we've already dealt with at LxA, for password management like the previous one, so it's a good alternative...

***LMD:*** stands for Linux Malware Detect and as the name suggests, it is a scanner that helps us detect malware that may be running on Linux-based systems.

Loki is a file scanner for checking so-called IPs.

***ClamAV:*** This is a well-known cross-platform antivirus that allows you to scan for malware, as well as add your own signatures, which the most well-known antivirus that it competes with does not allow. Don't forget to add its anti-rootkits.

***BleachBit:*** Another one of the tools we've talked about in this blog, a system cleaning and privacy tool because it can delete cookies, history, and more.

As you can see, these are simple tools used by security experts, but which we can use every day to protect our system and data.

***5. Why is the use of virtualization now so relevant?***

Virtualization is a broader concept that includes the creation of virtual versions of networks, applications, servers, and other components of the IT infrastructure. Why are such virtual solutions needed?

"Modern computers are very powerful. The problem is that modern processors load most programs very slowly — on average by 1-2%. If each server is used strictly for one task, it turns out to be inefficient. Virtualization allows you to place several dozen or even hundreds of virtual machines on one iron server. At a time when the processor could be idle, it performs the work of a neighboring server. Usually there are enough resources even when everyone works at the same time," explains Oleksandr Karimov, infrastructure solutions engineer at Wise IT.

Thus, the special software, the "hypervisor", simulates the physical hardware the required number of times on the same server. It also monitors the allocation of roles, resources, and priorities for copy work.

Digital hardware options that look like separate servers or computers with their own characteristics are called virtual machines. They are strictly isolated and independent of each other, so they can be managed separately.

The server on which the virtual machines are reproduced is the virtualization host.

With the use of virtualization, a huge room with rows of dozens of servers and wires between them is a thing of the past. The equipment is used much more efficiently, since tens or even hundreds of copies can be run on several physical machines. Resource optimization and cost reduction

AThe company saves on the purchase of physical equipment, spends less on its conditioning, management and administration. Virtual machine maintenance requires less staff.

"Operating costs always require huge budgets, and any large organization seeks to reduce these costs. It is better to direct them to advanced technologies, the introduction of innovations, things that increase productivity and efficiency. Everything old only slows down development," explains Denys Bazilskyi, Head of Hardware & Software development at Wise IT.

Thanks to monitoring, specialists clearly understand what resources and in what amounts are involved on certain virtual machines. This allows you to monitor the current load, redistribute capacity to optimize individual components, forecast growth rates and correctly draw up the budget.

Simplification of work with the infrastructure If previously the system administrator spent a lot of time on maintenance of computing power, connecting and redistributing cables between machines, with the advent of virtualization, network management has become easier. All networks are now configured with the click of a mouse. To create an additional virtual server, only its parameters are required. Hardware independence Virtual machines can be deployed on any hardware and an unlimited number of times. They are universal. With the help of virtualization, you can combine different services without buying separate equipment for each of them.

A virtual machine can be run in parallel with the user's operating system. For example, you work in the Windows operating system, and Linux opens in a separate window on the same computer.

Disaster Resilience Software is not always the same for different devices and requires special drivers to install. If there is a problem with the equipment, the software must be installed on a different one, or identical, or a huge effort must be spent on its adaptation. If there is a problem with the resources of the virtual machine, the hypervisor will reallocate the resources and the applications will continue to run without interruption.

If there is an accident in one data center, the application is automatically deployed at another site, while the user hardly notices the changes. All virtual machines remain functional and do not depend on the "iron". Thanks to them, the company will avoid downtime in operational activities, which can turn into colossal losses.

***6. How do you understand the concept of containerization?***

Containerization technology is another form of OS virtualization that offers isolation of applications in user spaces (containers). All containers use the same operating system. Thanks to the containerization technology, it is possible to run an application with the necessary libraries in a typical container that connects to the host or another external component using a simple interface.

All components necessary for the operation of the application (code, runtime environment, system tools, libraries and settings) are packed into one image and can be reused within the current task or for any other. The container is independent of the host's resources and architecture. It creates an isolated environment for the application without using the CPU, RAM or storage of the host OS. All processes take place internally.

Virtual machine and container: what's the difference

Virtual machines (VMs) and containers differ from each other. These are two different approaches to creating independent isolated computing environments on a physical server. What is special about each option?

Virtual machine. A hypervisor is required, each VM uses its own guest OS. Allows creating heterogeneous computing environments on one computer. Due to its own OS, the VM can take up several GB, and it takes some time to start the OS and all applications.

Container. Even several containers use the core of the same host OS. It allows creating only homogeneous computing environments on one computer. VM is much easier, the size is measured in MB. Able to start almost instantly.

***Advantages of containerization***

Creation speed. A container can be created faster than a VM. At the same time, the containerization environment provides more opportunities for some tasks.

Economy. The container takes up less space in the warehouse, which reduces overhead costs.

High productivity. The absence of network dependencies and conflicts increases development productivity. Each container is actually a microservice that can be updated independently without worrying about synchronization.

Version management. You can monitor the versioning of containers, monitor the differences between them.

The possibility of environment migration is calculated. All application and OS dependencies necessary for the application to work are encapsulated. This allows you to easily transfer the image of the container from one medium to another. Yes, one image can be run in Windows and Linux or dev/test/stage.

Standardization. As a rule, containers are created based on open standards. Therefore, they can be used in most Linux, Microsoft, and MacOS distributions.

Safety. Containers are isolated from each other and from the basic infrastructure. Changing/updating/deleting one container does not affect the other.

***Disadvantages of technology***

High complexity. The growth of the number of containers working with the application affects the complexity of their management. In a production environment, orchestrators should be used to work with multiple containers. For example, Kubernetes and Mesos.

Growth Often much more resources are packed into containers than are actually required. Because of this, the image grows, taking up more disk space.

Native Linux support. Docker and many other container technologies are based on Linux containers (LXC). Because of this, running containers in a Windows environment is not always convenient, and daily use is more difficult than when working in Linux.

Insufficient maturity. Application containerization technologies appeared on the market relatively recently. It is not always possible to solve the problem immediately. Sometimes it takes time to find a solution.

***7. What are the advantages/disadvantages of using open source software?***

Open source software refers to software that is produced and distributed with the source code open for modification by others. The copyright holder releases the source code under a license that grants the right to study, use, distribute, or modify the software.

It is usually developed collaboratively, as many developers can manipulate, change, and add to the source code to suit their needs. Open source software allows commercial companies to run, share, and modify software code.

Although OOS are usually available for free, some open source licenses have limitations. However, open source software provides an understanding of how the software works, allowing code and design to be reused if needed.

***Advantages of open source software***

Open source provides some great benefits to its users. Below are some pros to consider:

Cheaper compared to commercial products

It is estimated that open source software saves business owners $60 billion each year. This is not surprising, given that these programs are available to everyone. They are usually offered free of charge. Since this software is designed to be compatible with any computer, it allows users to extend the life of their equipment.

High reliability

A common misconception is that open source software is prone to failure. On the contrary, it is very reliable as it is tested by many developers, testers and users.

The reason open source software is so reliable is because it is developed by skilled experts. It can be from dozens to hundreds of people. This makes it possible to find errors and correct them in the shortest possible time. This results in software with excellent quality and useful features that work well.

Flexibility

Because open source software doesn't tie you to a proprietary product, you're not limited by a particular IT architecture. Proprietary products usually require both hardware and software updates. Thus, open source software allows users to mix and match their software to create a unique IT infrastructure that meets their needs.

With the wide range available in the market, the user only needs to find the one that suits his needs. Additionally, users can modify existing software accordingly. Access to the source code allows you to make it do what you want.

***Cons of open source software***

But not everything is so rosy. There are risks with open source software. Including:

Vulnerability to malicious attacks

Some people who have access to the source code of open source software do not have good intentions. While most people use this access to find defects and improve software, others look for ways to exploit vulnerabilities.

Additionally, some create bugs to infect hardware, annoy other users, or steal personal data. Such cases are rare in commercial software because companies apply strict quality control measures to ensure the safety of the software when it enters the market.

Unlike commercial software, OSS may not be user-friendly

Not all open source software is user-friendly. It was created to meet the needs of developers in realizing their ideas. Thus, they do not pay much attention to the user interface, which makes it difficult for those without special knowledge to use it.

Using OSS is not a matter of choosing the system you want and expecting it to meet your needs. You will usually need professionals to customize the system according to your expectations, which usually takes time and resources.

Lack of extensive customer support

Commercially produced software provides peace of mind to its users. After all, they know who designed, built and distributed the software. That way, they know who is responsible if the software doesn't work or causes hardware damage.

This is not the case with open source software because it is developed by different people. It is difficult for OSS users to identify a specific company or individual that they can hold accountable.

However, before you get discouraged by this, keep in mind that most software companies do not take any responsibility. Their end user license agreements disclaim all obligations, leaving you solely responsible for your use of the software.

This means that you will not receive any support if the software has problems and disrupts performance. While the open source community tends to be quick to respond to issues, no one is legally obligated to help. However, some OSS programs provide support services for a fee.

***8. How many active virtual consoles (terminals) can be in the process of Linux operation***

***silencing How to call them and switch between them? Give examples?***

Switching to Virtual Consoles

There are typically seven virtual consoles you can use. If you're running a desktop system with X11, it usually starts in the seventh virtual console. To switch to the first virtual console, you'll have to press Alt + F1. If it doesn't work for you, try Ctrl + Alt + F1 instead. In a Linux system, a console is a keyboard and monitor connected directly to the system. This will replace the serial terminal on minicomputers and mainframe Unix systems. This will allow the system administrator to log in as root, where this may be restricted on remote terminals connected to the system. Instead, Linux uses something called a framebuffer to display text on the screen.

The virtual console is similar to a full-screen Linux terminal emulator. On desktop systems, you'll often see boot messages before launching the manager display.

***9.What virtual console (terminal) performs the function of a graphical shell?***

***graphical (GUI)***

The graphical user interface (GUI) is perhaps more adapted for an untrained user who, not knowing the principles of the operating system, can get used to it intuitively one way or another. To work in command line mode (CLI), a person must first have knowledge, at least be able to use reference information.

The Linux kernel supports virtual consoles—consoles that are logically separated but have access to the same physical keyboard and display. The Linux console (and Linux virtual consoles) are implemented by the VT subsystem of the Linux kernel and do not depend on any user space software. This is different from a terminal emulator, which is a user space process that emulates a terminal and is usually used in a graphics display environment. On the Linux desktop, you can mostly use a GUI, but what if you could use a mode that was just text? With Linux virtual consoles, you can use a computer without graphics to restore a broken X system, or even use it completely without a graphical interface. Method 6: Favorites bar

Windows users prefer to pin important programs to the taskbar in order to quickly launch them when needed. In Linux graphical shells, this function is also implemented, but the line itself is called "Selected". if "Terminal" is not there at first, The ability to apply this method depends on the installed environment. It functions correctly in almost all common graphic shells, so you should definitely try it. The principle is to call the "Execute" utility, which is produced by pressing the Alt + F2 combination.

***10. Is it possible to log into a Linux system multiple times under the same system name? What benefits can it provide?***

Security includes many concepts, one of the most common is the general concept

access control. Before deciding on the issue of file access control, how is that right

ownership and permissions, you need to understand the basic concepts of user accounts

Linux, which are divided into several types.

Each user on a Linux system has an account associated with it, which apart from

login information (such as username and password) also determines how and where

the user can interact with the system. Privileges and access controls are defined

"boundaries" within which each user can work.

User Identifiers and Group Identifiers (UID/GID, User Identifiers/Group Identifiers) are basic

generated links to accounts. Early implementations were limited to 16-

bitwise (values ​​from 0 to 65535) integers, but 21st century systems

support 64-bit UIDs and GIDs. Users and groups are defined independently, therefore

the same ID can represent both a user and a group. Each user has not only a UID, but also a primary GID. The primary GID for a user can

be unique only to that user and ultimately may not

be used by other users. However, this group can also be a group to which

many users are included. In addition to these main groups, any user can be a member

other groups. By default on Linux systems, each user is assigned to a group with the same

with the same name as his username and the same GID as his UID. For example, create a new one

a user named newuser and, by default, his default group is as well

newuser. On Linux, the superuser account is root, which always has a UID of 0.

The superuser is sometimes called a system administrator and has unlimited

access and control over the system, including other users.

The default group for superuser has GID 0 and is also called root.

The home directory for superuser is a dedicated top-level directory, /root,

accessible only to the root user himself. Most system accounts in Linux are never logged in and don't need to be a certain shell in its attributes. Many processes belonging to these system accounts

records and are executed, placed in their own environment under system control,

subordinate to the specified system account. These accounts usually

have limited privileges or, more often than not, no privileges at all.

NOTE From the point of view of LPI Linux Essentials, system accounts have UID <1000, with

2 or 3-digit UID (and GID).

In general, system accounts should not have a valid login shell. Otherwise it would be

breach of security in most cases

**Висновки**