



ІТ-підтримка

Основи Linux



Q&A-сесія

менторка:

Католіченко Оксана



План сесії



Відповіді на питання у формі



Відповіді на питання у чаті



Основи Linux

Огляд з практичними порадами

1. Що таке Linux?
2. Серверні дистрибутиви
3. Робота з Linux
4. Установка програм пакетним менеджером
5. Структура файлової системи і робота з файлами
6. Процеси та споживання ресурсів сервера
7. Програмний комплекс system
8. Мережа



Що таке Linux

Історія та визначення



UNIX

У 1969 році в дочірньому підрозділі компанії AT&T – Bell Laboratories – було випущено операційну систему Unix, яка стала основною для великої кількості операційних систем того часу.

GNU – GNU is Not Unix

Метою проекту стало створення вільно розповсюджуваної операційної системи. Річардом Столлманом був написаний відомий маніфест GNU, який став основою ліцензії GPL (GNU General Public License), актуальною і донині. На початку 1990-х років у рамках проекту GNU було написано більшість компонентів ОС – оболонка bash, компілятори, відладчик, текстовий редактор та ін.

ОС Linux

У 1991 році Лінус Торвальдс, будучи студентом фінського університету, захопився ідеєю написання ядра операційної системи для свого персонального комп'ютера із процесором Intel.

Linux (або GNU/Linux)

- родина Unix-подібних операційних систем на базі ядра Linux, що включають той чи інший набір утиліт та програм проекту GNU. Linux-системи поширюються у вигляді різних дистрибутивів, що мають свій набір системних та прикладних компонентів (як вільних, так і пропрієтарних).



Серверні дистрибутиви

Що таке Linux?



Дистрибутив Linux — це операційна система, створена на основі ядра Linux, яка включає набір бібліотек і утиліт (пакетів), розроблених в рамках проекту GNU, а також систему управління пакетами (менеджер пакетів).

В даний час існує більше 500 різних дистрибутивів .

Дистрибутиви поділяються на кілька типів, залежно від базового дистрибутива та системи керування пакетом.

[RPM-based](#) (використовують формат пакетів .rpm)
RedHat Enterprise Linux, CentOS, Fedora

[DEB-based](#) (використовують формат пакетів .deb)
Debian, Ubuntu, Astra Linux



Завантаження

Робота з Linux



- BIOS / UEFI → MBR / GPT
- MBR / GPT → GRUB2
- GRUB2 → Kernel
- Kernel → Init



Командна оболонка

- command-line interface, CLI
- graphical user interface, GUI

*При вході користувача на сервер CLI запускається командна оболонка (у GUI командну оболонку можна запустити через емулятор, наприклад Terminal). **Командна оболонка (shell)** - це програма, яка приймає команди з клавіатури та передає їх операційній системі для виконання. Найбільш поширеною командною оболонкою в Linux є GNU bash (Bourne Again SHell). bash ґрунтується на іншій легковажній оболонці-попереднику – sh (Bourne sh), створеній Стефеном Борном.*



SSH

Робота з Linux



SSH (Secure SHell) – це протокол, що дозволяє здійснювати віддалене керування операційною системою та тунелювання TCP-з'єднань (наприклад, для копіювання файлів).

Серверна частина встановлюється на віддаленому сервері, а клієнтська – на комп'ютері, з якого здійснюється підключення.



SSH

Робота з Linux



Кілька прикладів для різних операційних систем:

Windows

PuTTY – вибір новичка, PowerShell (команда ssh), Xshell – вибір автора, MobaXterm , mRemoteNG

Linux

Terminal (команда ssh) – вибір новачка та автора, Asbru Connection Manager

MacOS

Terminal (команда ssh) – вибір новачка та автора, Core Shell

⦿ **Установка програм(утиліт) пакетним менеджером**



Пакетні менеджери

- Встановлення додатків у них виконується з пакетів – архівів із файлами скомпільованої програми.
- Пакетні менеджери мають власний перелік репозиторіїв – серверів з базою пакетів.
- Під час встановлення алгоритм менеджера знаходить необхідний пакет у базі та здійснює автоматичне скачування, встановлення та налаштування.



Менеджери пакетів



Найбільшого поширення набули формати пакетів .deb і .rpm

- **DEB (.deb)**

ОС – DEB-based, наприклад Debian, Ubuntu, AstraLinux

Система управління пакетами – DPKG (працює тільки з локальними пакетами)

Пакетний менеджер – apt

- **RPM (.rpm)**

ОС – RPM-based, наприклад RedHat Enterprise Linux, Fedora, CentOS

Система управління пакетами – RPM (працює тільки з локальними пакетами)

Пакетний менеджер – yum (в останніх дистрибутивах замінений на dnf)

⦿ Команди для взаємодії з пакетними менеджерами



посібник (довідкова інформація)

man <utility> # перегляд довідкової інформації по утиліті

<utility> --help # перегляд довідкової інформації по утиліті

менеджер пакетів **apt (.deb)**

sudo apt install nano # установка пакету

sudo apt remove nano # видалення пакету

sudo apt autoclean # видалення кешу пакету

менеджер пакетів **yum (.rpm)** - для dnf команди повністю подібні

sudo yum install nano # установка пакету

sudo yum remove nano # видалення пакету

sudo yum clean all # видалення кешу пакету



Структура файлів системи і робота з файлами



Типи файлів

- Звичайні файли –
Символьні та двійкові дані (текст, картинки, програми та ін.)
- Каталог (директорія) **d**
Список посилань на файли чи інші каталоги
- Символьні посилання **l**
Посилання на інші файли на ім'я
- Блокові пристрої **b**, символьні пристрої **c**
Інтерфейси для взаємодії з апаратним забезпеченням (диски, термінали, клавіатури, принтери та ін.). Коли відбувається звернення до файлу пристрою, ядро операційної системи надсилає запит драйверу цього пристрою.
- Сокети **s** та канали **p**
Інтерфейси для взаємодії процесів



Структура файлової системи



/bin (двійкові файли) – використовувані файли найбільш необхідних утиліт.

Може бути символьним посиланням в /usr/bin/boot – файли, необхідні для самого першого етапу завантаження – завантаження ядра (і зазвичай саме ядро)

/dev (devices) – блочні та символьні файли пристроїв (диски, термінали, клавіатури, принтери та ін.)

/etc (і так далі) – конфігураційні файли системи та різних програм

/home – домашні каталоги користувачів для зберігання «особистих» файлів

/lib (libraries) – файли бібліотек (стандартних функцій, необхідних багатьом програмам), необхідних для роботи утиліт. Може бути символьним посиланням в /usr/bin

/mnt (mount) – каталог для підключення файлових систем (з'ємних носіїв та ін.)

/opt (необов'язково) – каталог для додаткових програм (пропрієтарних драйверів, моніторингу агентів тощо)



Структура файлової системи



/proc (process) – файли оперативної пам'яті, в яких міститься інформація про виконані в системних процесах

/root – домашній каталог користувача root

/sbin (системні двійкові файли) – файли системних утиліт, необхідні для завантаження, резервного копіювання та відновлення системи. Може бути символьним посиланням в /usr/sbin

/sys (system) – віртуальна файлова система sysfs, яка містить інформацію про апаратне забезпечення (ЦПУ, ОЗУ, диски, мережеві пристрої), драйверах, ядрі системи та ін.

/tmp – каталог для тимчасових файлів, зазвичай зачищається при кожній завантажці системи

/usr – каталог користувача, який містить каталоги виконуваних файлів і конфігураційних файлів

/var (змінна) – файли, створені або використовувані різними програмами (логи, чергові, ідентифікатори процесів, БД та ін.)



Права доступу



- права для власника (u, user) – read, write, execution
- права для групи власника (g, group) – read, write, execution
- права для інших користувачів (o, other) – read, write, execution

Приклад :

`rwX r-- ---` у власника є права на все, у групи власника доступ тільки на читання, в інших доступ відсутній



Приклади команд



довідник (довідкова інформація)

man <utility> # перегляд довідкової інформації утилити

<utility> --help # перегляд довідкової інформації утилити

навігація

ls -la <dir> # перегляд змісту каталогу (також і прав доступу)

cd <dir> # перехід до каталогу

каталоги

mkdir <dir> # створення каталогу

rmdir <dir> # видалення порожнього каталогу

rm -rf <dir> # видалення каталогу з файлами



Приклади команд



файли

cat <file> # виведення файлу в консоль

less <file> # виведення файлу по сторінкам

tail <file> # виведення останніх рядків («хвоста») файлу

vi <file> # редагування файлу текстовим редактором

vi nano <file> # редагування файлу текстовим редактором

nano cp <file> <dir> # копіювання файлу в каталог

mv <file> <dir> # перемещення файлу в каталог (для зміни назви перемістіть файл в то й же каталог)

rm <file> # видалення файлу

rm -rf <dir>/* # видалення всіх файлів з каталогу



Приклади команд



права доступу

chown <user> <dir> # встановити користувача власником каталогу

chown <user>:<group> <dir> # встановити користувача і групу власниками каталогу

chmod 644 <file> # встановити права доступу rw- r-- r-- (читання і запис для власника, читання для групи власника, чтення для інших користувачів)

chmod 740 <file> # встановити права доступу rwx r-- --- (читання, запис і виконання для власника, читання для групи власників)

chmod u+w <file> # додати власнику право на запис файлу



Процеси та споживання ресурсів сервера



Статуси процесу

- Виконання (**R**, Running)
Виконання чи очікування ЦПУ для виконання
- Сон (**S**, Sleep)
Перерване програмне очікування
- Безперервний сон (**D**, Direct)
Очікування «прямого» сигналу від апаратної частини для переривання
- Припинено (**T**, Tracing)
Налагодження
- Зомбі (**Z**, Zombie)
Виконання завершено, проте ресурси не звільнено



Процеси та споживання ресурсів сервера



Основні команди

довідник (довідкова інформація)

man <utility> # перегляд довідкової інформації по утиліті

<utility> --help # перегляд довідкової інформації по утиліті

процеси

top

sudo ps aux

sudo kill -9 <pid> # вбивство процесу з PID

sudo killall -s 9 <name> # вбивство всіх процесів на ім'я

Утиліти для моніторингу використання ресурсів

htop # використання ресурсів за процесами (може знадобитися встановлення пакету htop)

nmon # використання процесора по ядрах, пам'яті, дисків та ін. (може знадобитися встановлення пакету nmon)

iostat # використання процесора в середньому по ядрах та читання/запис по дисках



Процеси та споживання ресурсів сервера



Основні команди

ЦПУ

lscpu # загальна інформація

cat /proc/cpuinfo # докладна інформація

ОЗУ

cat /proc/meminfo # загальна інформація та споживання

free -h # зручне представлення на основі даних із файлу meminfo

ps aux --sort -rss # використання пам'яті за процесами

диски

lsblk # загальна інформація

df -h # перегляд зайнятого місця по розділам

du -ch <dir> # перегляд зайнятого місця в каталозі

мережа

/sys/class/net/<interface>/speed # перегляд максимальної швидкості інтерфейсу



Програмний комплекс `systemd`



SystemD – це програмний комплекс, що складається із системних компонентів Linux. Основним компонентом є система ініціалізації системи SystemD, яка прийшла на зміну SystemV у більшості сучасних дистрибутивів. Ядро Linux запускає процес `systemd`, який, своєю чергою, запускає й інші процеси системи.



Програмний комплекс systemd

SystemD надає наступний функціонал:



- Запуск служб під час старту системи (по можливості паралельно) за різними націленнями (target, аналог рівнів завантаження в SystemV). Забезпечується зворотна сумісність із системами ініціалізації SystemV та LSB
- Контроль стану запущених служб
- Управління пристроями, входом до системи, мережевими підключеннями, ведення журналу подій
- Інтерфейси для реалізації функціональних можливостей ядра – cgroups, autofs, kdbus
- Набір утиліт для керування системою – systemctl, journalctl та ін.



SystemD Uni



SystemD Unit – це звичайний текстовий файл у стилі ini, який декларативно описує інформацію про служби .service, пристрої .device, цілі запуску .target та інші типи модулів systemd.

Опис юніту складається з кількох секцій:

Unit

Description (короткий опис), After (очікування запуску), Requires (обов'язкова залежність) та ін.

Service

Type (тип) – за замовчуванням Simple, ExecStart (команда для запуску), ExecStop (команда для зупинки) та ін.

Install

WantedBy (таргет або рівень запуску, на якому юніт повинен запуснитися)



SystemD Uni



Файли systemd unit розміщуються в наступних каталогах:

/etc/systemd/system – юніти, створювані адміністраторами

/usr/lib/systemd/system – юніти із встановлених пакетів

/run/systemd/system – юніти, які створюються під час роботи системи (в runtime)

Редагувати та створювати юніти можна за допомогою текстового редактора (наприклад, **vi** або **nano**)



SystemD Uni

Команди



довідник (довідкова інформація)

man <utility> # перегляд довідкової інформації по утиліті

<utility> --help # перегляд довідкової інформації по утиліті

перегляд юнітів

systemctl # список юнітів (це псевдонім команди systemctl list-units)

systemctl --type=service # юніти типу сервіси

systemctl | grep <keyword> # юніти, що містять ключове слово (корисно для швидкого пошуку)

systemctl cat <unit> # опис юніту

управління юнітами

systemctl status <unit> # статус (стан) юніту

sudo systemctl daemon-reload # перезавантаження конфігураційних файлів у systemd

sudo systemctl enable <unit> # включення сервісу в автозавантаження

sudo systemctl disable <unit> # відключення сервісу з автозавантаження

sudo systemctl start <unit> # запуск

sudo systemctl stop <unit> # зупинка

журнали логів

journalctl -u <unit> # читання логів з юніту

journalctl -u <unit> --since today # читання логів з юніту зі зміни дня



Мережа

Команди



довідник (довідкова інформація)

man <utility> # перегляд довідкової інформації по утиліті

<utility> --help # перегляд довідкової інформації по утиліті

перегляд мережних налаштувань сервера

ip a # IP адреса

ip n # ARP таблиця

cat /etc/resolv.conf # конфігурація DNS

sudo netstat -tulpn # відкриті порти

sudo ss -tulpn # відкриті порти (молодіжний варіант)

діагностика мережевих проблем

nslookup <hostname> # перевірка дозволу DNS імені

ping <host> # відправка ICMP пакетів до хоста

traceroute <host> # трасування до хоста UDP пакетами

telnet <host> <port> # перевірка доступності TCP порту на хості

nmap <host> # мережне сканування хоста

nmap -p T:<port> <host> # перевірка доступності TCP порту на хості (молодіжний варіант)

nmap -p U:<port> <host> # перевірка доступності UDP порту на хості (молодіжний варіант)



Як встановити Linux

Покрокова інструкція



Посилання 1

Посилання 2

Посилання 3