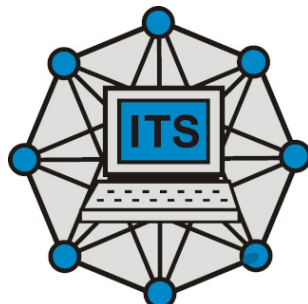


Міністерство освіти і науки України
Український державний університет науки і технологій
ННІ "Інститут промислових та бізнес технологій"
Кафедра інформаційних технологій і систем



МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт
з дисципліни

**“Хмарні обчислення в наукових дослідженнях та
сучасних педагогічних технологіях”**
за ОП “Комп’ютерні науки і технології”
спеціальності 122 — Комп’ютерні науки

Дніпро — 2022

Методичні вказівки до виконання лабораторних робіт з навчальної дисципліни “Хмарні обчислення в наукових дослідженнях та сучасних педагогічних технологіях” для здобувачів вищої освіти 3-го (освітньо-наукового) рівня за освітньою програмою “Комп’ютерні науки і технології” спеціальності 122 — Комп’ютерні науки денної та заочної форм навчання [Електронне видання] / Гуда А.І., Зимогляд А.Ю., — Дніпро: УДУНТ, ННІ “ІПБТ”, 2022. — 5 с.

Укладачі: проф., д.т.н. Гуда А.І., доц., к.т.н. Зимогляд А.Ю.

Методичні вказівки ухвалено Групою забезпечення якості освітньої програми “Інженерія програмного забезпечення у промисловості і бізнесі” третього (освітньо-наукового) рівня вищої освіти 15 червня 2022 р. (протокол № 4).

Гарант ОП доц., д.т.н. Гнатушенко В.В.

Методичні вказівки ухвалено Групою забезпечення якості освітньої програми “Комп’ютерні науки і технології” погоджено кафедрою інформаційних технологій і систем 15 червня 2022 р. (протокол №13).

Завідувач кафедри, проф., д.т.н. Гнатушенко В.В.

Лабораторна робота № 1

Створення та запуск віртуальної машини у OS Linux з використанням KVM

Ціль роботи: Навчитися створювати віртуальні машини у OS Linux з використанням технології KVM.

Завдання.

Написати програму мовою C (або C++) для OS Linux, яка повинна створювати віртуальну машину x86 за допомогою KVM. При цьому у машину повинні бути добавлені усі віртуальні пристрої, необхідні для запуску класичного x86 BIOS, але без подальшої завантаження операційної системи. Симулювання додаткових пристроїв, у тому числі класу “storage device” — за бажанням.

У якості образів BIOS рекомендовано використовувати набір opensource образів з проекту SeaBIOS.

Результат роботи кожного системного виклику, який може привести до помилки виконання, слід перевіряти. У разі помилки слід видавати повідомлення з відповідною інформацією.

Програма не повинна приводити до її завершення за допомогою сигналів SIGSEGV, SIGILL . . . Виконання програми не повинно приводити до “undefined behaviour”. Слід передбачити існування образів BIOS різного розміру, але відповідного до x86 системи без UEFI.

Рекомендовано видавати довідку у разі існування некоректних аргументів або використання ключу “-h”.

Коректним результатом роботи програми вважається старт віртуальної машини з відображенням інформації роботи образу BIOS та повідомленням про неможливість подальшого завантаження операційної системи.

Рекомендований порядок виконання лабораторної роботи

1. Створити пусту програму — перевірити працездатність компілятора та середовища сборки.
2. Визначити потрібні аргументи командної строки. Обробити ці аргументи. Рекомендовано застосовувати рівень налагодження та відповідний налагоджувальний друк.
3. Знайти і прочитати файл з BIOS. При цьому коректно визначити його розмір.
4. Зарезервувати об'єм пам'яті, достатній для створення віртуальної машини. Рекомендовано використовувати анонімне відображення у пам'ять (mmap). Якщо потрібно, скопіювати образ BIOS у потрібне місце (або об'єднати ці дії).

5. Відкрити файл KVM та отримати дескриптор.
6. Використовуючи цей дескриптор, створити віртуальну машину.
7. У віртуальній машині створити підсистему обробки IRQ (KVM_CREATE_IRQCHIP).
8. Створити PIT2 (KVM_CREATE_PIT2).
9. У віртуальній машині створити один віртуальний процесор (KVM_CREATE_IRQCHIP).
10. Задати пам'ять, що була виділена, як фізичну пам'ять віртуальної машини (KVM_SET_USER_MEMORY_REGION)
11. Задати початкові значення сегментних (селекторних) реєстрів, що відповідають старту машини у реальному режимі (KVM_SET_SREGS).
12. Для взаємодії з віртуальною машиною створити структуру kvm_run та зареєструвати її.
13. Запустити цикл роботи віртуальної машини (KVM_RUN).
14. Реалізувати взаємодію з виводом віртуальної машини з використанням порту 0x402.

Базові питання до лабораторної роботи

1. Яка технологія використовується для швидкої візуалізації у OS Linux?
2. Які необхідні апаратні та програмні передумови для можливості використання цієї технології?
3. Який спеціальний файл використовується для створення віртуальних машин?
4. Який системний виклик використовується для додавання віртуальних пристроїв до машини?
5. Який є мінімальний набір віртуальних пристроїв для старту образу BIOS?
6. Як можна отримати початкову налагоджувальну інформацію при старті образу SeaBIOS?
7. Як створюється фізичний адрес при роботі x86 у реальному режимі?

Рекомендовані інформаційні матеріали

1. The Definitive KVM (Kernel-based Virtual Machine) API Documentation. <https://www.kernel.org/doc/html/next/virt/kvm/index.html>
2. Linux KVM documentation. <https://www.linux-kvm.org/page/Documents>
3. Oracle Linux. KVM User's Guide. <https://docs.oracle.com/en/operating-systems/oracle-linux/kvm-user>
4. QEMU GettingStartedDevelopers <https://wiki.qemu.org/Documentation/GettingStartedDevelopers>
5. Kernel-based Virtual Machine https://en.wikipedia.org/wiki/Kernel-based_Virtual_Machine