

Міністерство освіти і науки України
Державний заклад
«Луганський національний університет імені Тараса Шевченка»

Факультет технологій та інформаційних систем

Кафедра інформаційних технологій та систем

Солоп Денис Анатолійович

**ВПРОВАДЖЕННЯ ВІДДАЛЕНОГО ДОСТУПУ ДО RASPBERRY PI 4
НА ЗАСАДАХ ПАКЕТУ XRDP**

кваліфікаційна робота

здобувача вищої освіти першого (бакалаврського) рівня

освітньої програми «Комп'ютерна інженерія»

за спеціальністю 123 Комп'ютерна інженерія

Особистий підпис



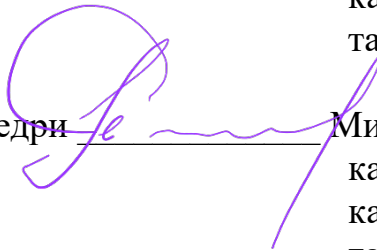
Денис СОЛОП

Науковий керівник



Геннадій МОГИЛЬНИЙ,
кандидат технічних наук, доцент
кафедри інформаційних технологій
та систем

Завідувач кафедри



Микола СЕМЕНОВ,
кандидат педагогічних наук, доцент
кафедри інформаційних технологій
та систем

Міністерство освіти і науки України
Державний заклад «Луганський національний університет
імені Тараса Шевченка»

Факультет (інститут)	Факультет технологій та інформаційних систем (повна назва)
Кафедра	Кафедра інформаційних технологій та систем (повна назва)
Освітній ступень	Бакалавр (код, назва)
Напрямок підготовки	123 Комп'ютерна інженерія (код, назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри ІТС
М.А. Семенов

(підпис)

(ініціали, прізвище)

“ ”

2025 р.

ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТУ

Солоп Денис Анатолійович

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи)	Впровадження віддаленого доступу до Raspberry PI 4 на засадах пакету XRDP
--------------------------	--

Керівник кваліфікаційної роботи	Могильний Г.А., доцент (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
---------------------------------	--

затверджена наказом по університету

2. Строк подання студентом проекту (роботи)	
---	--

3. Вихідні дані до роботи (проекту)	Налаштовано систему на Raspberry PI для віддаленого підключення та використання у навчальному процесі
-------------------------------------	--

(визначаються кількісні або (та) якісні показники, яким повинен відповідати об'єкт розробки)

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) АНАЛІТИЧНИЙ ОГЛЯД МІКРОКОМП'ЮТЕРІВ ТА ОПЕРАЦІЙНИХ СИСТЕМ МІКРОКОМП'ЮТЕРІВ, НАЛАШТУВАННЯ ПРИЄДНАННЯ ДО MICROSOFT AD, НАЛАШТУВАННЯ ВІДДАЛЕНОГО ДОСТУПУ ДО МІКРОКОМП'ЮТЕРІВ

(визначаються назви розділів або (та) перелік питань, які повинні увійти до тексту ПЗ)

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
--

6. Консультанти розділів проекту/роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання „ 24 ” жовтня 2025р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
	Вибір теми роботи, вивчення наукової літератури, затвердження теми та керівника.	До 30 жовтня	
	Аналіз літературних джерел за темою роботи. Розробка та апробація методики дослідно-експериментальної роботи. Подання структури теоретичної частини роботи та плану експериментальних досліджень.	Другий тиждень листопада (10 листопада)	
	Робота над теоретичною частиною. Подання теоретичної частини роботи для першого читання науковим керівником.	До 15 грудня	
	Усунення зауважень, урахування рекомендацій наукового керівника. Подання теоретичної частини роботи на друге читання.	До 28 січня	
	Проведення експериментальної роботи. Поетапний аналіз та обговорення її результатів. Перевірка стану виконання роботи.	Перший тиждень березня	
	Урахування рекомендацій наукового керівника, усунення недоліків, підготовка варіанта роботи до передзахисту. Розробка презентації.	До 1 травня	
	Попередній захист роботи на кафедрі	травень	
	Доопрацювання роботи з урахуванням рекомендацій після передзахисту. Подання роботи науковому керівникові та рецензентові на підготовку відгуку та рецензії	За 10 днів до державної атестації	
	Подання на кафедру остаточного варіанта роботи, переплетеного та підписаного автором, науковим керівником і рецензентом.	За 5 днів до державної атестації	

Студент

Керівник проекту (роботи)

підпис

підпис

Д.А. Солоп

(ініціали, прізвище)

Г.А. Могильний

(ініціали, прізвище)

АНОТАЦІЯ

Солоп Денис Анатолійович

Тема: Впровадження віддаленого доступу до Raspberry PI 4 на засадах пакету XRDP.

Спеціальність: 123 (F7) «Комп'ютерна інженерія»

Установа: ЛНУ імені Тараса Шевченка, 2026 р.

Бакалаврська робота містить: технічне завдання, відомість проєкту, пояснювальну записку на 79 с., 41 рис., 1 табл., 25 джерел та додатки.

Об'єкт дослідження – мікрокомп'ютери .

Предмет дослідження – впровадження віддаленого підключення до Raspberry PI 4 на засадах пакету XRDP.

Мета дослідження – аналіз та впровадження додаткового програмного забезпечення для організації віддаленого використання Raspberry PI 4 у навчальному процесі в умовах змішаного навчання.

Для вирішення поставлених задач використовувались методи аналізу та програмування; методи створення автоматизованих мікропроцесорних систем; методи математичної статистики та обробки експериментальних результатів.

Результати роботи.

Проведено огляд існуючих систем мікрокомп'ютерів, що дозволило виявити їхні переваги, недоліки та визначити актуальні тенденції в даній галузі. Здійснено детальне виявлення вимог до проєктованої системи віддаленого доступу на основі потреб навчального процесу з серверною інфраструктурою. Проведено порівняльний аналіз, основних операційних систем, які зараз використовуються. Розроблено алгоритм підключення мікрокомп'ютерів до домену Ms AD. Описана послідовність налаштування програмних модулів системи віддаленого доступу на базі платформи Raspberry PI за допомогою пакету XRDP. Зроблено всі додаткові налаштування системи, що необхідні для використання у навчальному процесі: корегування кількості підключень, корегування SSH підключень, автозавантаження додатків.

Ключові слова. МІКРОКОМП'ЮТЕР, ОПЕРАЦІЙНА СИСТЕМА, ДОМЕН MICROSOFT, RASPBERRY PI, RDP, DNS, IOT (INTERNET OF THINGS), МІКРОПРОЦЕСОРНА СИСТЕМА.

ABSTRACT

Denis Solop

Theme: Implementing remote access to Raspberry PI 4 based on the XRDP package.

Speciality: 123(F7) "Computer Engineering"

Institution: Luhansk Taras Shevchenko National University (LTSNU), 2026.

Diploma work contains: technical specifications, project information, explanatory note on 79 pages, 41 figures, 1 table, 25 sources and appendices.

A research object is microcomputers.

The article of research implementing remote connection to Raspberry PI 4 based on the XRDP package.

The purpose of the study is to analyze and implement additional software for organizing the remote use of Raspberry PI in the educational process in distance learning environments.

To solve the tasks, methods of analysis and programming were used; methods of creating automated microprocessor systems; methods of mathematical statistics and processing of experimental results.

Job performances. A review of existing microcomputer systems was conducted, which allowed us to identify their advantages, disadvantages and identify current trends in this industry. A detailed identification of the requirements for the designed remote access system based on the needs of the educational process with server infrastructure was carried out. A comparative analysis of the main operating systems currently in use was carried out. An algorithm for connecting microcomputers to the Ms AD domain was developed. The sequence of configuring the software modules of the remote access system based on the Raspberry PI platform using the XRDP package was described. All additional system settings necessary for use in the educational process were made: adjusting the number of connections, adjusting SSH connections, autoloading applications.

Keywords. MICROCOMPUTER, OPERATING SYSTEM, MICROSOFT DOMAIN, RASPBERRY PI, RDP, DNS, IOT (INTERNET OF THINGS), MICROPROCESSOR SYSTEM.

ІТС.4КІ.0226.01-ВП

ВПРОВАДЖЕННЯ ВІДДАЛЕНОГО ДОСТУПУ ДО RASPBERRY PI 4 НА ЗАСАДАХ ПАКЕТУ XRDP.

[illegible]

					ІТС.4КІ.0226.01-ВП								
Змн.	Арк.	№ докум.	Підпис	Дата	ВІДОМІСТЬ ПРОЄКТУ				Літ.	Арк.	Акрушів		
Розроб.	Слоп Д											1	1
Керівник	Могильний Г.А..								ЛНУ				
Реценз.	Козуб Ю.Г.								Кафедра ІТС, Гр.4КІ				
Н. Контр.													
Зав. каф.	Семенов М.А..												

Міністерство освіти і науки України
Державний заклад «Луганський національний університет
імені Тараса Шевченка»
Факультет (інститут) Факультет технологій та інформаційних систе
(повна назва)
Кафедра інформаційних технологій та систем
(повна назва)
XRDP
(код, назва)

ТЕХНІЧНЕ ЗАВДАННЯ
на виконання програмної розробки (ПР):
" Впровадження віддаленого доступу до Raspberry PI 4 на засадах пакету
XRDP "

ІТС.4КІ.0226.02-ТЗ

ПОГОДЖЕНО
Керівник кваліфікаційної роботи

Могильний Г.А. 

“ ” 2026р

ВИКОНАВЕЦЬ
Студент групи 4КІ

Солоп Д.А.

“ ” 2026р

Лубни – 2026

ЗМІСТ

ВСТУП.....	3
1. ПРИЗНАЧЕННЯ ПРОДУКЦІЇ	3
2. ТЕХНІЧНІ ХАРАКТЕРИСТИКИ	4
3. АЛГОРИТМ РОБОТИ ПРИСТРОЮ	4
4. ТЕХНІКО-ЕКОНОМІЧНІ ВИМОГИ ДО КІНЦЕВОГО ПРОДУКТУ	5
5. ВИМОГИ ДО МАТЕРІАЛІВ ТА КОМПЛЕКТУЮЧИХ.....	5
6. ЕТАПИ ВИКОНАННЯ ПР.....	5
7. ПРИЙМАННЯ	6
8. ПОРЯДОК ВНЕСЕННЯ ЗМІН ДО ТЕХНІЧНОГО ЗАВДАННЯ, ЩО ЗАТВЕРДЖЕНО.....	7

					ІТС.4КІ.0226.02-ТЗ					
Змн.	Арк.	№ докум.	Підпис	Дата	ТЕХНІЧНЕ ЗАВДАННЯ			Літ.	Арк.	Акрушів
Розроб.		Солоп Д.А.								
Керівник		Могильний Г.А.							2	7
Реценз.		Козуб Ю.Г.						ЛНУ Кафедра ІТС, Гр.4КІ		
Н. Контр.										
Зав. каф.		Семенов М.А..								

ВСТУП

1.1 Найменування: Впровадження віддаленого доступу до Raspberry PI 4 на засадах пакету XRDP.

1.2 Шифр ПР: XRDP

1.3 Підстава до виконання ПР: Підставою для виконання даної розробки є завдання на бакалаврську роботу.

1.4 Терміни розробки:

1.4.1 Початок 30 жовтня 2025р.

1.4.2 Закінчення 20 травня 2026р.

1.5 Фінансується за рахунок коштів замовника.

1. ПРИЗНАЧЕННЯ ПРОДУКЦІЇ

1.1. Призначення:

Ця розробка призначений для управління та налаштування мікро комп'ютерів Raspberry PI та включає наступні складові:

- Стислий огляд мікро комп'ютерів;
- Стислий огляд аналогів для віддаленого керування Raspberry PI;
- Стислий огляд операційних систем та системних програмних засобів віддаленого підключення
- Налаштування віддаленого доступу з використанням пакету XRDP

Мікрокомплекс створюється з метою:

- Використання мікрокомп'ютерів у навчальному процесі
- Розробка та впровадження програмних засобів для віддаленого керування мікро комп'ютерів Raspberry PI.
- Створення робочого та готового до впровадження комплексу для студентів.

Завдання, що вирішуються

- Складання звіту про мікрокомп'ютери.
- Розробка технічних умов для налаштування мікрокомп'ютери

					ІТС.4КІ.0226.02-ТЗ	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		

Raspberry PI.

- Налаштування основного програмного модуля.
- Підготовку комплексу додаткових файлів –параметрів.

2. ТЕХНІЧНІ ХАРАКТЕРИСТИКИ

Особливих вимог до мови програмування та мови скриптів не надається. Мікрокомп'ютер повинен працювати на мінімальних технічних характеристиках мікрокомп'ютера, а саме:

- 1) Операційна система – RaspbianOS;
- 2) Пристрій Raspberry PI 4;
- 3) ОЗУ – не більше 4 Гбайт;
- 4) Накопичувач – не більше 64 Гбайт;
- 5) Використовувати USB ВЕБ камеру .

Також винесено додаткові параметри, які немало важливі для запуску та коректної роботи приладу.

- A. Потрібно попередньо налаштувати операційну систему RasbyanOS;
- B. Підключення до локальної мережі установи;
- C. Швидкість з'єднання не менше ніж 100Мбит/с;
- D. Під'єднати до внутрішнього домену Ms AD ;

Робота пристрою повинна бути зручною для використання всіма студентами при виконанні лабораторних робіт, мати зручний та інтуїтивно зрозумілий інтерфейс.

3. АЛГОРИТМ РОБОТИ ПРИСТРОЮ

Мікрокомплекс на базі мікрокомп'ютеру повинен складатися з наступних модулів і їх функцій:

1) Програмний модуль підтримки зовнішньої ВЕБ камери для перегляду результатів програмування приладів у віддаленому режимі

2) Програмний модуль інтеграції з Ms AD повинен дозволяти вхід певних користувачам– студентам та співробітникам

					<i>ITC.4KI.0226.02-T3</i>	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		

- 3) Програмний модуль контролю доступу до зовнішніх пристроїв
- 4) Налаштування клієнтської частини повинно бути зручним у використанні та мати рекомендації по використанню

4. ТЕХНІКО-ЕКОНОМІЧНІ ВИМОГИ ДО КІНЦЕВОГО ПРОДУКТУ

Вартість робіт по розробці даної ПР визначається згідно договору на розробку. Вартість запропонованих аналогів повинна забезпечити економічну доцільність їх застосування.

5. ВИМОГИ ДО МАТЕРІАЛІВ ТА КОМПЛЕКТУЮЧИХ

5.1. Вимоги до екологічної безпечності під час експлуатації.

Не пред'являються.

5.2. Спеціальні вимоги до кінцевого продукту.

Не пред'являються.

5.3. Вимоги до кібербезпеки та безпеки для населення під час експлуатації продукції.

Враховувати особливості використання у навчальному процесі.

6. ЕТАПИ ВИКОНАННЯ ПР.

Етапи виконання ПР можуть уточнювати згідно календарного плану робіт по узгодженню між замовником та виконавцем

№	Етапи виконання роботи	Термін виконання та обсяг робіт	звітні матеріали
1	Аналіз мікрокомп'ютерів та аналогів Raspberry PI 4. Аналіз вимог. Вибір та налаштування структури. Попереднє тестування	Листопад 2025-січень 2026	Частковий програмний комплекс на ЕОМ замовника, що виконує всі основні функції та звітна документація п.7.2
2	Коректування структури. Налаштування допоміжних	січень 2026-	Готовий програмний

					ІТС.4КІ.0226.02-ТЗ	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

№	Етапи виконання роботи	Термін виконання та обсяг робіт	звітні матеріали
	функцій. Розробка остаточної версії програмного комплексу та його опрацювання. Тестування	березень 2026	комплекс на ЕОМ замовника та звітна документація п.7.2
3	Доопрацювання окремих системних модулів та навчання користувачів. Розробка звітних матеріалів згідно п.8 цього ТЗ	квітень 2026-травень 2026	звітні матеріали згідно пункту 7

7. ПРИЙМАННЯ

7.1. Необхідні вимоги для впровадження ПР та завершення робіт.

Оцінка результатів розробки і доцільність її продовження здійснюється замовником по представленню наступних матеріалів:

- встановлений програмний комплекс на мікрокомп'ютерах замовника;
- перелік файлів на резервному носії;
- стислий опис роботи ПР та опис всіх файлів, які необхідні для роботи ПР.
- перелік документів
 - Технічне завдання
 - Пояснювальна записка

7.2. Перелік звітних документів, необхідних для прийняття етапів роботи:

- стислий опис результатів етапу у вигляді анотованого звіту(для 1 та 2 етапів);
- частковий програмний комплекс на Raspberry PI замовника згідно календарного плану робіт;
- акт приймання продукції.

7.3. Загальний перелік до приймання звітних документів, макетів, експериментальних зразків.

До приймання пред'являються: акт здачі-приймання продукції, акт впровадження ПР.

					ІТС.4КІ.0226.02-ТЗ	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

8. ПОРЯДОК ВНЕСЕННЯ ЗМІН ДО ТЕХНІЧНОГО ЗАВДАННЯ, ЩО ЗАТВЕРДЖЕНО.

Дане технічне завдання може уточнюватися в процесі розробки ПР при узгодженні сторін з оформленням доповнень до ТЗ.

					<i>ІТС.4КІ.0226.02-ТЗ</i>	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЗ «ЛУГАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА»

Факультет технологій та інформаційних систем

(назва факультету, інституту)

Інформаційних технологій та систем

(назва кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

БАКАЛАВРА

(освітньо-кваліфікаційний рівень)

на тему:

**ВПРОВАДЖЕННЯ ВІДДАЛЕНОГО ДОСТУПУ ДО RASPBERRY PI 4 НА
ЗАСАДАХ ПАКЕТУ XRDP**

Виконав: студент 4 курсу, групи 4КІ
напряму підготовки (спеціальності)

123(F7) «Комп'ютерна інженерія»

(шифр і назва напряму підготовки, спеціальності)

Солоп Д.А.

(прізвище та ініціали)

Керівник



Могильний Г.А.

(прізвище та ініціали)

Рецензент

Козуб Ю.Г.

(прізвище та ініціали)

ЗМІСТ

ВСТУП.....	4
РОЗДІЛ 1 ОПИС RASPBERRY PI	7
1.1 Огляд RASPBERRY PI ТА НАПРЯМКИ ВИКОРИСТАННЯ.....	7
1.2 Огляд ОПЕРАЦІЙНИХ СИСТЕМ ДЛЯ RASPBERRY PI.....	8
<i>Головні характеристики Raspberry Pi OS.....</i>	<i>10</i>
<i>Історія розвитку операційної системи</i>	<i>11</i>
<i>Порівняння Raspberry Pi OS проти Ubuntu</i>	<i>11</i>
1.3 РОЗГОРТАННЯ RASPBERRYPIOS	13
Висновки до розділу.....	24
РОЗДІЛ 2 ВІДДАЛЕНИЙ ДОСТУП.....	26
2.1 ВСТУП ДО ВІДДАЛЕНОГО ДОСТУПУ	26
<i>Аналіз версії ОС.....</i>	<i>26</i>
<i>Загальні додаткові налаштування.....</i>	<i>27</i>
<i>Типове програмне забезпечення віддаленого керування.....</i>	<i>27</i>
<i>Налаштування локальної IP-адреси.....</i>	<i>28</i>
2.2 ІНТЕГРАЦІЯ СПИСКУ КОРИСТУВАЧІВ З ІНФОРМАЦІЙНОЮ СИСТЕМОЮ	29
2.3 ДОСТУП ДО RASPBERRY PI ЧЕРЕЗ SSH	31
<i>Підключення до SSH-сервера</i>	<i>33</i>
<i>Налаштування обмежень на використання SSH.....</i>	<i>34</i>
2.4 ОСОБЛИВОСТІ СИСТЕМНИХ НАЛАШТУВАНЬ ПРИ ВИКОРИСТАННІ XRDP.....	36
<i>Перевірка з'єднання та загальні проблеми</i>	<i>36</i>
<i>Налаштування використання зовнішніх пристроїв/ портів.....</i>	<i>39</i>
<i>Налаштування процесу обмежень кількості та тривалості підключень ...</i>	<i>41</i>
<i>Налаштування перезавантаження за розкладом</i>	<i>46</i>
<i>Налаштування перезавантаження після виходу.....</i>	<i>49</i>
<i>Налаштування процесу входу.....</i>	<i>51</i>
<i>Виправлення помилок виходу.....</i>	<i>54</i>
<i>Налаштування автозавантаження програм</i>	<i>56</i>
Висновки до розділу.....	58
ЗАГАЛЬНІ ВИСНОВКИ.....	61
СПИСОК ЛІТЕРАТУРИ.....	64
ДОДАТКИ.....	67

Додаток А. ФАЙЛ НАЛАШТУВАНЬ ETC\POLKIT- 1\RULES.D\NEW_ALLOW_WIN_REPOSIT.RULES.....	67
Додаток Б. ФАЙЛ НАЛАШТУВАНЬ \ETC\ SECURITY\ PAM_MOUNT.CONF.XML.....	68
Додаток В. ФАЙЛ НАЛАШТУВАНЬ \ETC\ SSH\ SSHD_CONFIG	70
Додаток Д. ФАЙЛ НАЛАШТУВАНЬ \ETC\SSSD\ SSSD.CONF	72
Додаток Е. ФАЙЛ НАЛАШТУВАНЬ \ETC\ SYSTEMD\ LOGIND.CONF	73
Додаток Ж. ФАЙЛ НАЛАШТУВАНЬ \ETC\XDG\AUTOSTART\CHEESE.DESKTOP	74
Додаток З. ФАЙЛ НАЛАШТУВАНЬ \ETC\XRDP\ SESMAN.INI.....	75
Додаток К. ФАЙЛ НАЛАШТУВАНЬ \ETC\XRDP\ STARTWM.SH	79

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		3

ВСТУП

Запровадження правового режиму воєнного стану в Україні зумовило вимушену трансформацію освітнього процесу та масовий перехід закладів вищої освіти (ЗВО) на гібридний та дистанційний формати навчання. Специфіка функціонування ЗВО в умовах систематичних безпекових загроз та регулярних оголошень повітряних тривог пов'язана з необхідністю переміщення учасників освітнього процесу до захисних споруд (бомбосховищ). Подібний режим життєдіяльності створює суттєві деструктивні чинники для технічного забезпечення навчання, зокрема через потребу в віддленому використанні комп'ютерного обладнання та його подальшому повторному перезапуску після нормалізації ситуації. Відтак, виникає гостра науково-практична потреба у реорганізації функціонування кафедральних та загальноінститутських комп'ютерних лабораторій на основі впровадження концепції безперервного, віддаленого та автоматизованого моніторингу й управління обчислювальними ресурсами.

У контексті забезпечення дистанційного доступу до лабораторної бази ЗВО критично важливою науково-технічною задачею є розробка та інтеграція підсистем віддаленого використання та керування обчислювальними ресурсами. Автоматизація та інструменти дистанційного підключення дозволяють мінімізувати часові витрати на адміністрування інфраструктури, оптимізувати використання аудиторного фонду згідно із затвердженим розкладом занять, а також підвищити загальну дидактичну й ергономічну ефективність освітнього процесу [1,2].

Сучасний етап розвитку інформаційно-комунікаційних технологій (станом на 2025–2026 рр.) демонструє зміщення акцентів у бік створення комплексних систем віддаленого керування.

					ІТС.4КІ.0226.03-ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Солоп Д.А.			Вступ			
Керівник		Могильний Г.А.						
Реценз.		Козуб Ю.Г.						
Н. Контр.								
Зав. каф.		Семенов М.А.						
						Лім.	Арк.	Акрушів
							3	3
					ЛНУ			
					Кафедра ІТС, Гр.4КІ			

Доцільним та інноваційним розв'язанням окресленої проблеми є проектування chuûên sýstems (спеціалізованого програмного забезпечення) та гетерогенних мережевих додатків з використанням сучасних технологій мереж [3-6]. Такі рішення, окрім базової функції, мають інтегрувати інструменти безпечних тунелів (VPN/WireGuard, Zero Trust Network Access — ZTNA) для надання захищеного термінального доступу до фізичних обчислювальних потужностей лабораторій без безпосередньої присутності користувачів у навчальних приміщеннях

Таким чином, комплексний аналіз методів дистанційного розгортання мікрокомп'ютерних систем та інтеграції відповідного спеціалізованого програмного забезпечення є високоактуальним науково-практичним завданням. Реалізація систем віддаленого доступу в умовах сучасних безпекових викликів дозволяє не лише оптимізувати коефіцієнт корисного використання наявних обчислювальних потужностей ЗВО, а й гарантує безперервність освітнього процесу за гібридної та повністю дистанційної форм навчання.

Впровадження таких систем забезпечує критично важливу стійкість, адаптивність та автономність цифрової інфраструктури закладу освіти в умовах нестабільного зовнішнього середовища.

Об'єкт дослідження – мікрокомп'ютери .

Предмет дослідження – впровадження віддаленого підключення до Raspberry PI 4 на засадах пакету XRDP.

Мета дослідження – аналіз та впровадження додаткового програмного забезпечення для організації віддаленого використання Raspberry PI 4 у навчальному процесі в умовах змішаного навчання.

Для вирішення поставлених задач використовувались методи аналізу та програмування; методи створення автоматизованих мікропроцесорних систем; методи математичної статистики та обробки експериментальних результатів.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		5

1. Провести аналіз операційних систем, що використовуються на Raspberry PI;
2. Провести загальний аналіз системного програмного забезпечення віддаленого доступу до Raspberry PI;
3. Провести налаштування віддаленого доступу до мікрокомп'ютерів у лабораторії ЛНУ імені Т.Шевченка на засадах пакету XRDP.

У першому розділі було проведено опис Raspberry PI, Наведено загальний огляд RASPBERRY PI, його апаратне забезпечення для різноманітних серій. Виконано стисла порівняльна характеристика та огляд операційних систем для RASPBERRY PI, наведено ґрунтовний опис розгортання операційної системи RapberryPIOS

У другому розділі розглянуто вимоги до впровадження додаткових системних програмних додатків для реалізації функції віддалено використання мікрокомп'ютерів, описано процеси розгортання, інтерфейс користувача та архітектуру програмного забезпечення. Також були описані основні особливості та можливості додатків, зручність їх використання для досягнення поставлених цілей – організації віддаленого доступу за рахунок пакету XRDP.

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		6

РОЗДІЛ 1

ОПИС RASPBERRY PI

1.1 Огляд Raspberry Pi та напрямки використання

Однопалатні комп'ютери (SBC — Single Board Computers) серії Raspberry Pi, розроблені однойменним благодійним фондом (Raspberry Pi Foundation), є малогабаритними високо інтегрованими обчислювальними системами. Апаратна архітектура цих пристроїв базується на енергоефективних системах на кристалі (SoC) з процесорними ядрами ARM, інтегрованою оперативною пам'яттю (LPDDR4X/LPDDR5), універсальними інтерфейсами введення-виведення (GPIO), а також підсистемами бездротового (Wi-Fi, Bluetooth) та дротового (Gigabit Ethernet) зв'язку. Висока питома обчислювальна потужність у поєднанні з мінімальним форм-фактором обумовлює широке впровадження цих рішень у різноманітних галузях науки, техніки та виробництва.

До основних доменів прикладного застосування мікрокомп'ютерів Raspberry Pi належать:

Академічно-освітнє середовище: Платформа позиціонується як базовий інструмент для вивчення комп'ютерних наук, архітектури обчислювальних систем, кібер безпеки та низькорівневого програмування. Завдяки доступності та широкій підтримці операційних систем на базі Linux (зокрема Raspberry Pi OS), ці пристрої інтегруються в лабораторні практикуми з робототехніки, теорії автоматичного керування та проектування ембеддед-систем (embedded systems).

Системи автоматизації («Smart Home»): У межах концепції DIY (Do It Yourself) мікрокомп'ютери застосовуються як центральні хаби локальної автоматизації, медіасервери (на базі рішень Plex або Kodi) та автономні комплекси моніторингу. Низьке енергоспоживання та гнучкість налаштування роблять їх оптимальною апаратною основою для індивідуальних розробок.

					ІТС.4КІ.0226.03-ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Солоп Д.			Розділ 1		
Керівник		Могильний Г.А.					
Реценз.		Козуб Ю.Г.					
Н. Контр.							
Зав. каф.		Семенов М.А.					
					Літ.	Арк.	Акрушів
						3	3
					ЛНУ		
					Кафедра ІТС, Гр.4КІ		

Завдяки наявності інтерфейсного роз'єму GPIO, що підтримує протоколи I2C, SPI, UART, мікрокомп'ютер виступає як ефективний контролер для збору даних із гетерогенної мережі датчиків для Інтернет речей (IoT) та периферійних обчислень (Edge Computing). Сучасні моделі (зокрема Raspberry Pi 5) активно залучаються до концепції Edge AI — виконання нейромережевих моделей та первинної аналітики безпосередньо на периферійному пристрої без надсилання сирих даних у хмару.

Платформа на засадах Raspbey PI Забезпечує повноцінне середовище для розробки та компіляції програмного коду високорівневими мовами (Python, C/C++, Java, Go, Rust). Наявність інтерфейсу PCIe (у сучасних ітераціях) дозволяє розробникам підключати швидкісні NVMe-накопичувачі та спеціалізовані ШІ-прискорювачі, що суттєво розширює можливості створення прототипів складних систем.

Окремі модифікації (наприклад, Compute Module 4/5) інтегруються у промислові контролери, системи контролю та керування доступом (СККД), термінали самообслуговування (киоски), а також у системи цифрових вивісок (Digital Signage). Промислове використання обумовлене високою відмовостійкістю, прогнозованим життєвим циклом компонентів та економічною доцільністю впровадження.

1.2 Огляд операційних систем для Raspberry pi

Деякі дослідники зосереджені саме на порівнянні Ubuntu з іншими дистрибутивами, а саме дослідник Rahman у публікації «Performance Analysis of Raspberry PI OS vs. Ubuntu for IoT Edge Computing» [7] провів порівняльний аналіз споживання ресурсів (CPU, RAM). Дослідження показало, що хоча Ubuntu споживає більше пам'яті, її системна архітектура краще підходить для навчання сучасним DevOps-технологіям (Kubernetes, Docker), ніж полегшена Raspberry PI OS.

Сучасний ринок ОС для Raspberry PI можна класифікувати за кількома ключовими напрямками[13-20]:

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

- Універсальні та освітні ОС: офіційна Raspberry PI OS (колишня Raspbian) залишається стандартом де-факто завдяки своїй легкості, стабільності та інтегрованим інструментам для навчання, як-от Python та Scratch. Поруч із нею більшої популярності набуває Ubuntu, що пропонує повноцінний робочий стіл та сучасне 64-бітне середовище для професійної розробки;

- Спеціалізовані інженерні платформи: для завдань кібербезпеки та пентестингу виділяється Kali Linux, а для створення вузлів розумного будинку — Home Assistant OS (HAOS), яка працює за принципом «чорної скриньки», максимально спрощуючи автоматизацію;

- Специфічні та ретро-системи: окремий сегмент займають мультимедійні оболонки LibreELEC та для ретро-ігор (наприклад, RetroPie) та системи з унікальною архітектурою, як-от RISC OS, що наразі мають переважно історичну або вузькопрофільну цінність.

В цілому, комплексна оцінка різноманітних ОС наведена у таблиці 1.

Таблиця 1 Зведена інформація операційних систем для Raspberry PI

Назва ОС	Особливості ОС	Актуальність	Зручність для навчання	Коментар
Raspberry PI OS	Офіційна, стабільна, легка	+++++++	Висока (вбудовані Python, Scratch)	Найкращий вибір для старту
Ubuntu	Сучасна, популярна, 64-біт	+++++++	Висока (для студентів IT/CS)	Чудова для сервера або як заміна ПК
DietPi / Home Assistant OS	Для специфічного керування	+++++	Тільки профільна (Автоматизація)	Стандарт індустрії DIY-автоматизації
RetroPie / LibreELEC	Ігровий комбайн (емулятори)	+++++	Низька (розваги)	Найкраще для ретро-ігор
Kali Linux	Інструменти для хакерів/безпеки	+++++	Тільки профільна (Кібербезпека)	Звичайному користувачеві не потрібна
Arch Linux ARM	Конструктор для профі	+++	Тільки для поглибленого вивчення Linux	Складно, для досвідчених користувачів.

Назва ОС	Особливості ОС	Актуальність	Зручність для навчання	Коментар
Windows 10 IoT	Урізана Windows без робочого столу			Проект закритий, не має сенсу
RISC OS	Дуже стара ОС.	Не актуальна	Ні	Тільки як музейний експонат
Slackware / SARPİ	Найстаріший Linux, складний			Занадто складно

Raspberry Pi OS (раніше відома як Raspbian) — це офіційна операційна система для одноплатних комп'ютерів Raspberry Pi. Вона створена на базі дистрибутиву Debian Linux і є головним інструментом для розробників, мейкерів, студентів та інженерів, які працюють із цією платформою.

Нижче наведено детальний огляд характеристик цієї ОС, історії її розвитку, а також порівняння з популярною ОС Ubuntu.

Головні характеристики Raspberry Pi OS

Архітектура та ядро: Базується на монолітному ядрі Linux. Доступна як у 32-бітному (для старіших моделей), так і в 64-бітному варіантах (для Raspberry Pi 3, 4, 5, Zero 2W).

Графічне середовище (PIXEL): Використовує власне легке робоче оточення PIXEL (Pi Improved Xwindows Environment, Lightweight), яке побудоване на базі LXDE. Воно виглядає сучасно, але майже не навантажує слабкі процесори.

Оптимізація під залізо: ОС містить специфічні драйвери для інтеграції з чіпами Broadcom, апаратним прискоренням відео (VideoCore) та керуванням живленням.

Робота з периферією (GPIO): Головна фішка — підтримка роботи з 40-піновим роз'ємом GPIO «з коробки». Бібліотеки для керування датчиками, моторами та інтерфейсами (I2C, SPI, UART) вже встановлені та налаштовані.

Інструмент `raspi-config`: Утиліта (консольна або графічна), яка дозволяє в один клік розігнати процесор, увімкнути камеру, налаштувати Wi-Fi або змінити роздільну здатність екрана без ручного редагування системних файлів.

Історія розвитку операційної системи

1. Епоха «Raspbian» (2012–2020)

Липень 2012 року: Майк Томпсон і Пітер Грін створили незалежний порт Debian під архітектуру ARMv6 (для першого процесора Raspberry Pi 1). Вони назвали його Raspbian (Debian + Raspberry).

Офіційне визнання: Raspberry Pi Foundation швидко оцінила потенціал проєкту та зробила Raspbian своєю офіційною ОС, взявши розробку під своє крило.

2016 рік: З'явилася графічна оболонка PIXEL. Замість базового браузерера Eriqhanу система перейшла на Chromium з апаратним прискоренням.

2. Ребрендинг та перехід на 64 біти (2020 — дотепер)

Травень 2020 року: ОС офіційно перейменували з Raspbian на Raspberry Pi OS. Це було пов'язано з виходом 64-бітних процесорів (починаючи з Pi 3/4) та поступовим відходом від виключно кодової бази оригінального Raspbian для ARMv6.

Лютий 2022 року: Після тривалого бета-тестування вийшла перша стабільна 64-бітна версія Raspberry Pi OS. Це дозволило повноцінно використовувати об'єми оперативної пам'яті понад 4 ГБ (на Pi 4 та Pi 5).

Сучасний стан (база Debian Bookworm/Trixie): Сьогодні система повністю підтримує графічний сервер Wayland (замість старого X11), пропонує сучасний стек аудіо PipeWire і залишається основним стандартом для IoT-розробки.

Порівняння Raspberry Pi OS проти Ubuntu

Хоча обидва дистрибутиви мають спільне коріння (Ubuntu також базується на Debian), вони створені для абсолютно різних філософій використання.

Переваги та недоліки Raspberry Pi OS

					ITC.4KI.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

Переваги (Pros):

- Максимальна продуктивність на слабкому залізі: Система «витискає» все можливе навіть зі старих плат або мікрокомп'ютерів серії Zero завдяки легкій оболонці.
- Бездоганна сумісність із залізом: Жодних проблем із драйверами камери (libcamera), дисплеїв, капелюхів розширення (HATs) чи аудіокарт для Pi.
- Енергоефективність та стабільність: Консервативний цикл оновлень Debian гарантує, що система не «впаде» після чергового апдейту посеред ночі.
- Утиліти для новачків: Наявність `raspi-config` та передвстановленого софту для навчання (Thonny Python, Scratch) робить її ідеальним стартом.

Недоліки (Cons):

- Застарілий дизайн інтерфейсу: Графічна оболонка PIXEL виглядає досить просто і бідно порівняно з сучасним та красивим GNOME в Ubuntu.
- Консервативне ПЗ: Оскільки база системи — Debian, версії програм у репозиторіях часто старіші (хоча й стабільніші), ніж в Ubuntu.
- Гірша адаптація під корпоративні хмари: Якщо вам потрібно розгорнути Kubernetes-кластер (K8s) або складні enterprise-платформи, Ubuntu має кращу інтеграцію на ринку.
- Відсутність вбудованої підтримки Snap: Canonical просуває свій формат пакетів Snap, який спрощує встановлення деякого пропрієтарного софту, чого в Pi OS немає за замовчуванням.

Таким чином, так як мета — робота з «залізом», робототехніка, GPIO, створення медіацентру, та проект запускається на слабкій платі — обирайте Raspberry Pi OS.

Якщо будується потужний домашній сервер (Homelab), планується запускати складні Docker-стеки, комерційні бази даних або хочете отримати повноцінний

					ITC.4K1.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

сучасний Linux-десктоп на Raspberry Pi 5 з наявним ОЗУ 8-16ГБ — тоді краще розгорнути Ubuntu.

У навчальній лабораторії ЛНУ імені Тараса Шевченка використовують Raspberry Pi 4 с ОЗУ 4ГБ тому, як базову ОС навчальних стендів обрано Raspberry Pi OS . Однак, у випадку коли апаратна частина буде модернізована та мати більш великі обчислювальні потужності то краще обрати UBUNTU, яка має один із найбільших репозиторіїв ПЗ, підтримку контейнерів, адаптована під 64-бітну архітектуру ARM для Raspberry Pi, підтримує повноцінний Desktop, що надає сучасний інтерфейс (GNOME), який виглядає і працює як повноцінний персональний комп'ютер (ПК).

1.3 Розгортання RaspberyPIOS

Найпростіший спосіб – використовувати Raspberry Pi Imager [20], який дозволяє вибрати образ Ubuntu під час прошивки SD-карти.

Якщо користуєтеся ОС Linux, наприклад UBUNTU то, відкрийте термінал і виконайте команду:

```
sudo snap install rpi-imager
```



Рис. 1.1 Розгортання Raspberry PI

Однак врахуйте, що для встановлення нової Ос на прилад Raspbery PI спочатку вставте microSD картку в комп'ютер. Включить живлення та натисніть

клавішу «shift». З'явиться вікно завантаження. Оберіть необхідний прилад (рис. 1.2).

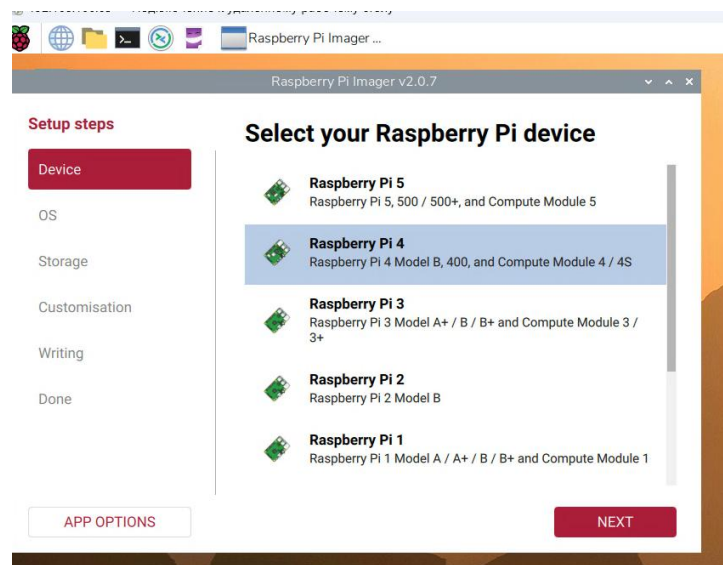


Рис.1.2 Вибір приладу

Оберіть пункт меню « CHOOSE OS».

Прокрутка вниз меню натисніть « Інше » загального призначення ОС ».

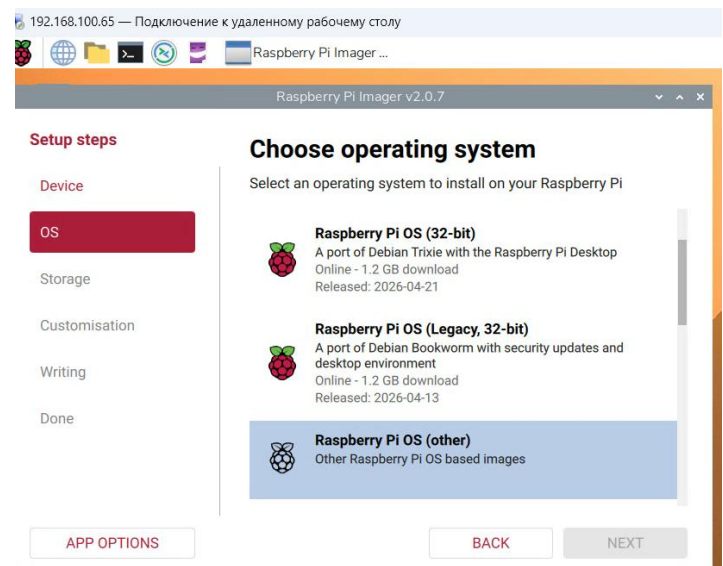


Рис. 1.3 Розгортання Ubuntu

Тоді буде можливість до переглянути список завантажень до вибрати від. Виберіть останню версію ОС

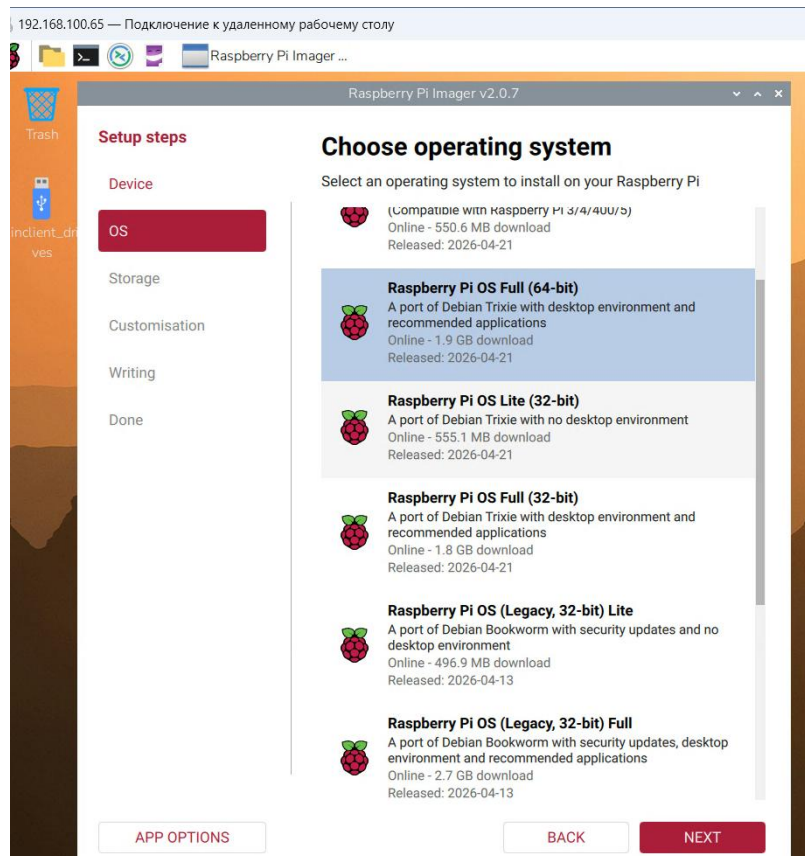


Рис. 1.4 Вибір версії

Виберіть Меню «SD Card ». Виберіть необхідний microSD картка, яку повинно бути вставленою, і натисніть кнопку « WRITE».

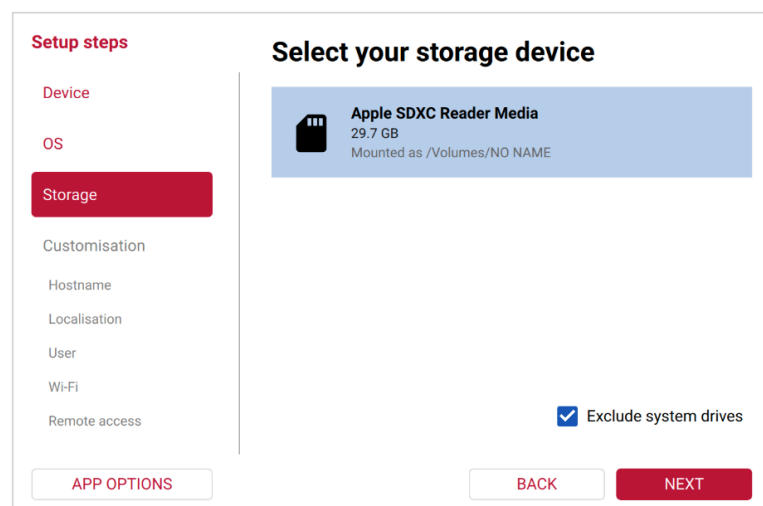


Рис. 1.5 Вибір носія

Raspberry Pi транслює це ім'я хоста в мережу за допомогою mDNS . Коли підключаєте Raspberry Pi до мережі, інші пристрої в цій мережі можуть зв'язуватися з Raspberry Pi за допомогою `<hostname>.local` або `<hostname>`.

У підвкладці «Локалізація» виберіть свою столицю. Imager автоматично завершить часовий пояс і розкладку клавіатури для цього міста. Також можете змінити ці налаштування незалежно один від одного.

Цей вибір також встановлює регуляторний домен Wi-Fi; регуляторний домен завжди відповідає вибраній столиці.

У підвкладці «Користувач» введіть ім'я користувача та пароль для адміністратора Raspberry Pi, а потім виберіть «Далі» .

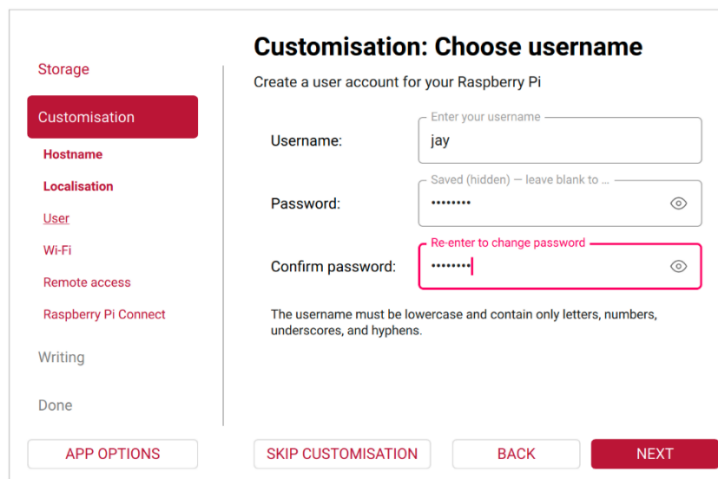


Рис. 1.6 Вибір користувача

Ім'я користувача має бути написане малими літерами та містити лише літери, цифри, символи підкреслення та дефіси.

Створення імені користувача та пароля на цьому етапі не вимагає введення цих облікових даних під час завантаження Raspberry Pi. Щоб запитувати пароль під час завантаження, вимкніть автоматичний вхід. Інструкції див. у статті Вимагати пароль під час запуску .

У підвкладці «Віддалений доступ» налаштуйте параметри SSH, а потім натисніть «Далі» .

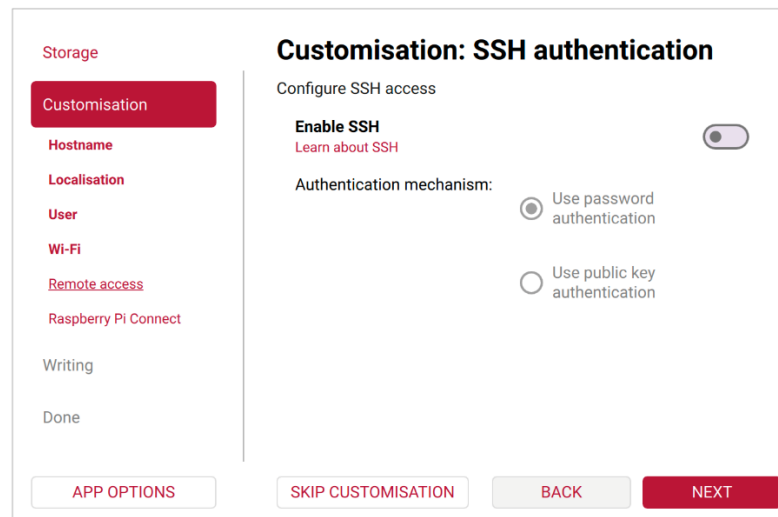


Рис. 1.7 Налаштування SSH

Щоб використовувати Raspberry Pi віддалено через мережу, увімкніть опцію «**Увімкнути SSH**» за допомогою відповідного перемикача.

Якщо плануєте використовувати Raspberry Pi без підключення до комп'ютера (а не як настільний комп'ютер), можете або увімкнути SSH, або налаштувати **Raspberry Pi Connect** для свого пристрою.

Якщо увімкнули SSH, виберіть один із наведених нижче методів автентифікації:

- **Використовувати автентифікацію за паролем** : щоб увійти до Raspberry Pi через мережу, використовуючи ім'я користувача та пароль, які вказали на вкладці **Налаштування > Користувач** .
- **Використовувати автентифікацію за допомогою відкритого ключа** : щоб налаштувати Raspberry Pi на використання автентифікації на основі SSH-ключа, Imager відкриє поле для введення відкритого ключа. Якщо у конфігурації SSH вже є відкритий ключ RSA, Imager використовуватиме цей відкритий ключ. Якщо його немає, можете скопіювати відкритий ключ у Imager або вибрати «**Огляд**» , щоб знайти файл відкритого ключа на комп'ютері.

Після перегляду вкладок у Imager можете переглянути **зведення** своїх виборів на вкладці «**Написання**» .

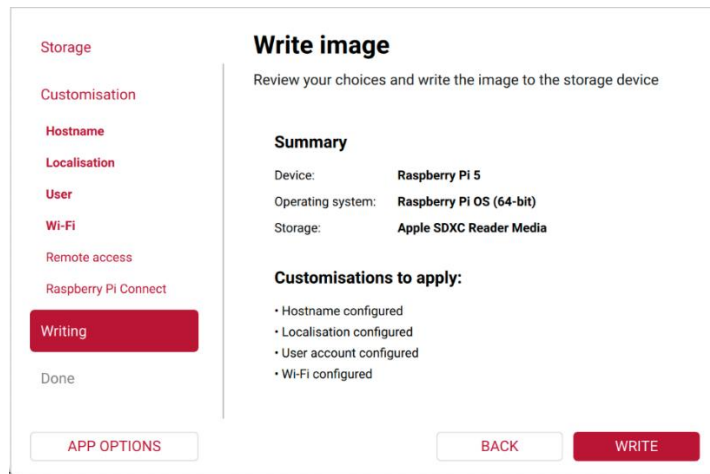


Рис. 1.8 Перевірка налаштувань

Якщо впенені своїм вибором і хочете продовжити:

1. Виберіть «Запис» у правому нижньому куті Imager.
2. Imager відображає попередження про те, що збирається перезаписати всі дані на носії інформації. Щоб продовжити, виберіть « **Я розумію, стерти та записати** » . Якщо бачите запит адміністратора із запитом на дозволи на читання та запис на носій інформації, надайте Imager дозволи для продовження.

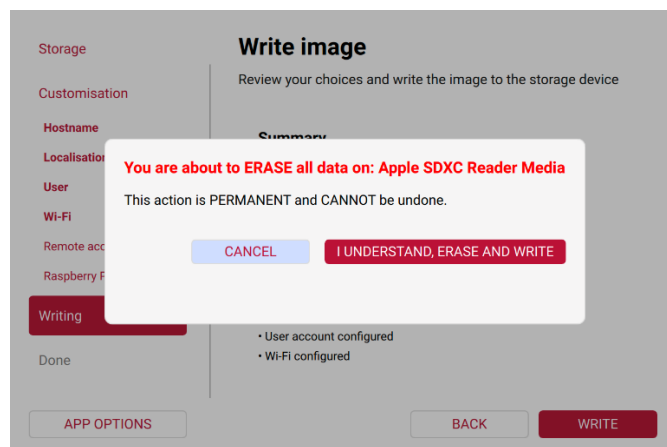


Рис. 1.9 Початок запису

Imager починає записувати зображення на пристрій зберігання даних. Це може тривати кілька хвилин.

3. Після запису зображення Imager перевіряє правильність запису зображення. Це також може тривати кілька хвилин. Ми рекомендуємо вам виконати крок перевірки. Однак, якщо хочете скасувати перевірку, виберіть « **Пропустити перевірку** » .

4. Після запису та перевірки зображення Imager відображає вкладку «Готово» . Натисніть «Готово» , щоб вийти з Imager.

Тепер можемо виконати інструкції для стандартного (настільного) або безголового (віддаленого) налаштування , щоб запустити Raspberry Pi відповідно до потреб.

Конфігурація при першому завантаженні

У цьому розділі розглядається налаштування після встановлення, незалежно від того:

- Пропущено кроки налаштування в Imager.
- Не використовував Imager для встановлення ОС.
- Використовуєте Raspberry Pi з попередньо встановленою ОС, або
- Хочете змінити налаштування після встановлення, щоб налаштувати

Raspberry Pi відповідно до потреб.

Якщо виконао кроки для попереднього налаштування Raspberry Pi під час налаштування ОС в Imager, як описано в ранніше, не потрібно налаштовувати Raspberry Pi знову, якщо не хочете змінити якісь параметри.

Якщо маєте налаштування для робочого столу та пропустили налаштування ОС в Imager, або якщо використовуєте Raspberry Pi з попередньо встановленою ОС, під час першого завантаження – буде представлено майстер налаштування.

Безголове налаштування оминає майстер налаштування; оскільки було попередньо налаштовано параметри в Imager, Raspberry Pi завантажується безпосередньо з мережі. Якщо вибрали SSH для віддаленого доступу, можете віддалено підключитися до свого безголового Raspberry Pi, ввівши команду в командний рядок на комп'ютері.

Для навігації майстром налаштування знадобляться монітор і клавіатура. Миша необов'язкова, і за потреби можете використовувати клавішу **Tab** для переміщення по майстру.

Якщо планується підключити USB-мишу або клавіатуру Bluetooth як частину цього процесу налаштування, підключіть USB-адаптер перед завантаженням Raspberry Pi.

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

Майстер налаштування проведе вас через наведені нижче параметри налаштування. Виберіть «Далі» у правому нижньому куті, щоб переглянути кожен із параметрів.

На наступному зображенні показано вікно привітання майстра налаштування, яке також відображається, коли Raspberry Pi намагається підключитися до будь-яких знайдених периферійних пристроїв Bluetooth.

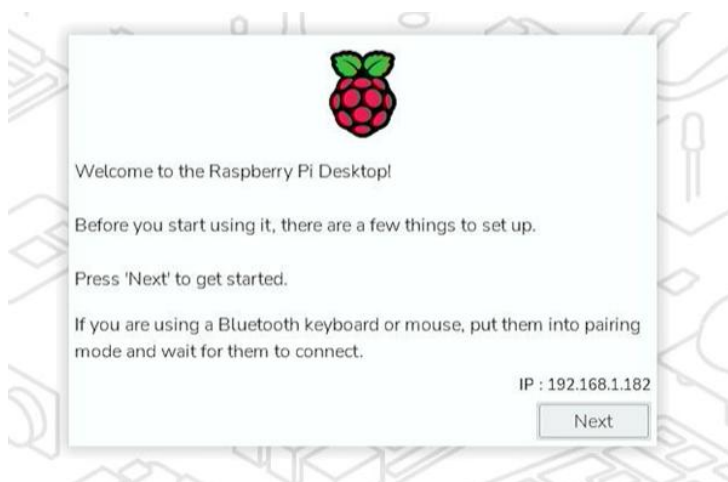


Рис. 1.10 Вітальне вікно майстра налаштування

Якщо використовуєте клавіатуру або мишу Bluetooth, Raspberry Pi шукає сумісні пристрої та підключається до першого знайденого пристрою для кожного периферійного пристрою. Цей процес працює з вбудованими або зовнішніми USB-адаптерами Bluetooth. Якщо використовуєте USB-адаптер, підключіть його перед завантаженням Raspberry Pi.

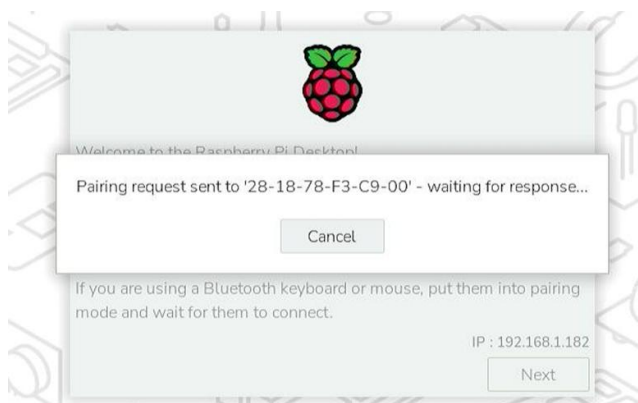


Рис. 1.11 Спроба сполучення з пристроєм Bluetooth

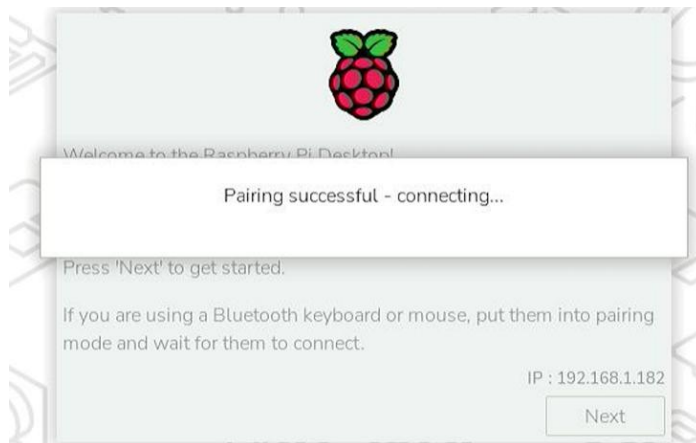


Рис. 1.12 Повідомлення про успішне сполучення Bluetooth

Скористайтесь випадаючими меню, щоб встановити країну, мову та часовий пояс. За потреби також можете:

- Виберіть «Використовувати англійську мову», щоб інтерфейс робочого столу відображався англійською мовою, замінюючи локальну мову за замовчуванням.
- Виберіть «Використовувати американську розкладку клавіатури», щоб ОС правильно інтерпретувала введення з клавіатури з американською розкладкою.

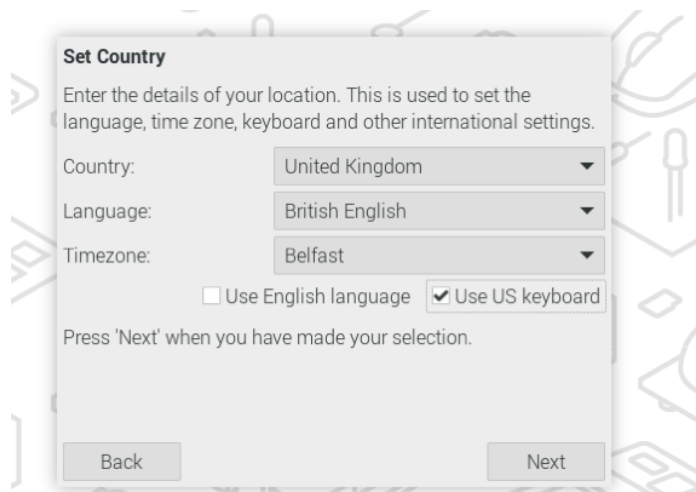


Рис. 1.13 Налаштування вікна країни в майстрі налаштування

Налаштуйте ім'я користувача та пароль для облікового запису користувача за замовчуванням. Це ім'я користувача має починатися з літери та може містити лише малі літери, цифри, символи підкреслення та дефіси.

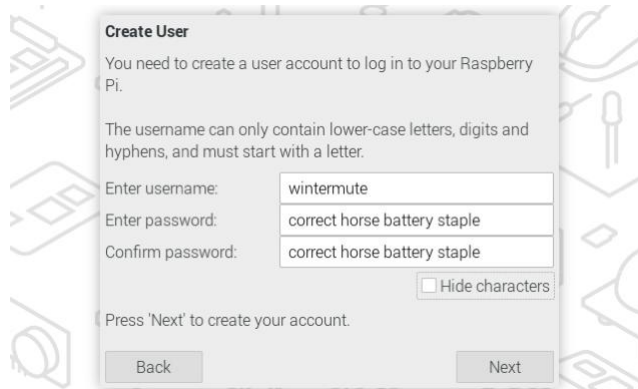


Рис. 1.14 Вікно «Створити користувача» у майстрі налаштування

Бездротова мережа

Виберіть потрібну мережу Wi-Fi зі списку. Якщо мережа вимагає пароля – буде запропоновано ввести його, перш ніж продовжити. Можете пропустити цей крок, якщо бажаєте.

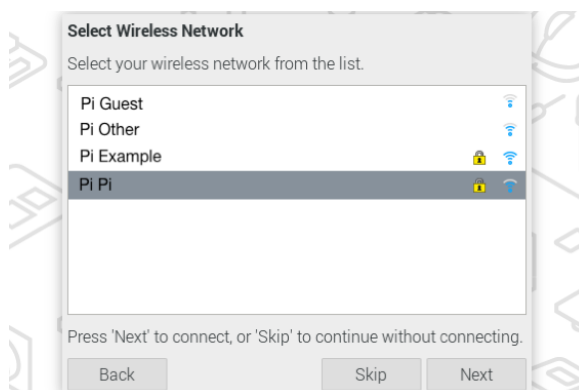


Рис. 1.15 Вибір бездротової мережі в майстрі налаштування

Виберіть веб-браузер за замовчуванням: Chromium або Firefox . За потреби, щоб заощадити місце на завантажувальному носії, видаліть браузер, який не встановили як браузер за замовчуванням.

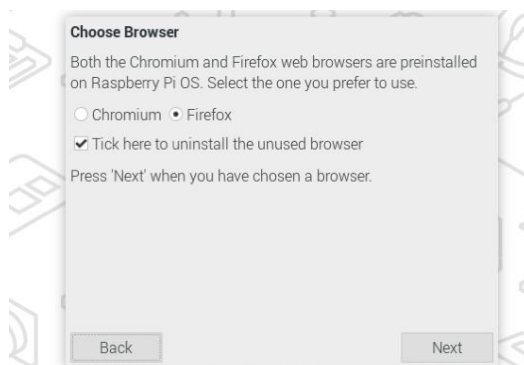


Рис. 1.16 Оновлення програмного забезпечення

Якщо налаштували доступ до Інтернету для Raspberry Pi або якщо Raspberry Pi підключено до Інтернету за допомогою кабелю Ethernet, можете оновити свою ОС та програмне забезпечення до останніх версій.

Під час оновлення програмного забезпечення майстер налаштування видаляє браузер, який не є браузером за замовчуванням, якщо вирішили видалити його на кроці «Браузер» .

Оновлення програмного забезпечення може тривати до півгодини, залежно від кількості доступних оновлень та швидкості інтернет-з'єднання.

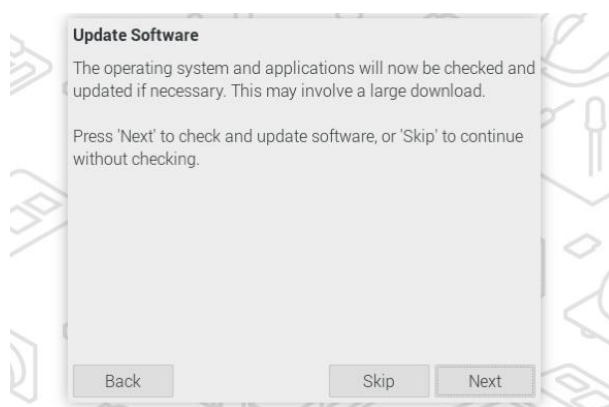


Рис. 1.17 Оновлення програмного забезпечення в майстрі налаштування

Після виконання кожного кроку майстра налаштування з'явиться вікно підтвердження про налаштування системи. Виберіть «ОК» , а потім «Запустити» (або, якщо замість цього бачите «Перезавантажити» , виберіть цю опцію, щоб перезавантажити Raspberry Pi). Вибрана конфігурація застосовується до Raspberry Pi, який потім завантажується та завантажує графічне середовище робочого столу, де можете використовувати вікна, значки та меню, подібно до інших настільних комп'ютерів.

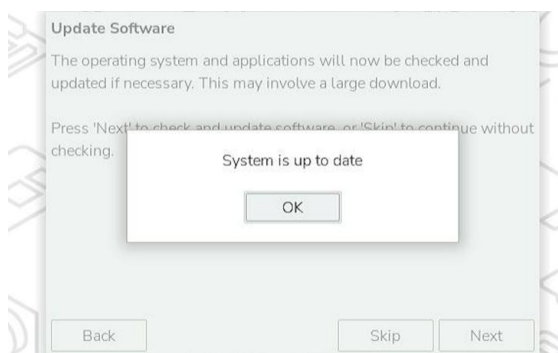


Рис. 1.18 Повідомлення про успішне оновлення системи

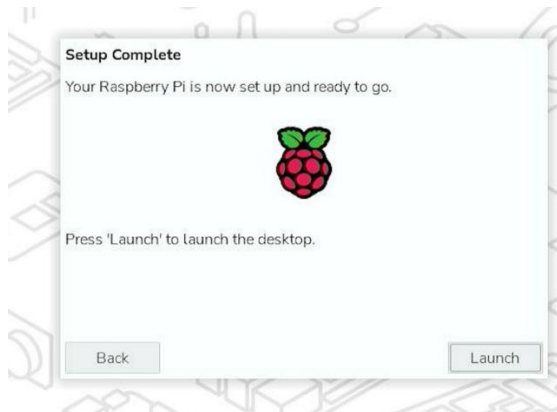


Рис. 1.19 Повідомлення про успішне завершення налаштування

Висновки до розділу

У першому розділі кваліфікаційної роботи проведено детальний теоретико-технічний аналіз апаратного та програмного забезпечення одноплатних мікрокомп'ютерів сімейства Raspberry Pi, що послужило підґрунтям для розробки системи віддаленого доступу.

За результатами проведеного дослідження сформовано такі висновки:

Обґрунтовано вибір апаратної платформи. Мікрокомп'ютер Raspberry Pi 4 є високоефективним рішенням завдяки збалансованому поєднанню обчислювальної потужності ARM-архітектури, наявності універсальних інтерфейсів введення-виведення (GPIO), низького енергоспоживання та компактного форм-фактора. Це робить його оптимальною апаратною основою для створення гнучких інженерних рішень, інтеграції в мережі Інтернету речей (IoT) та використання в ролі базису для лабораторних практикумів у вищій школі.

Визначено оптимальну операційну систему для проєкту. Шляхом компаративного аналізу сучасних дистрибутивів Linux (зокрема, порівняння Raspberry Pi OS та Ubuntu) встановлено, що для розгортання на наявних у навчальній лабораторії ЛНУ імені Тараса Шевченка стендах Raspberry Pi 4 з об'ємом ОЗУ 4 ГБ найбільш раціональним є використання офіційної Raspberry Pi OS. Дана ОС забезпечує максимальну продуктивність і стабільність на обмежених ресурсах за рахунок полегшеної графічної оболонки PIXEL (на базі

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

LXDE) та має вбудований інструментарій для взаємодії з периферійними пристроями.

Оцінено технологію підготовки та конфігурування системи. Досліджено та практично реалізовано методику розгортання операційної системи за допомогою спеціалізованого програмного забезпечення Raspberry Pi Imager. Особливу увагу приділено можливостям кастомізації образу на етапі запису, зокрема автоматизації створення облікових записів користувачів, конфігурування Wi-Fi мережі та активації демона SSH. Це дозволяє виконувати первинне налаштування у «безголовому» (headless) режимі без необхідності підключення локального монітора та клавіатури, що критично важливо для побудови віддалених лабораторій.

Таким чином, результати аналізу, проведеного в першому розділі, повністю підтверджують технічну доцільність і готовність платформи Raspberry Pi 4 до інтеграції пакету XRDP з метою організації повноцінного, графічного віддаленого доступу до робочого столу в межах навчального процесу

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

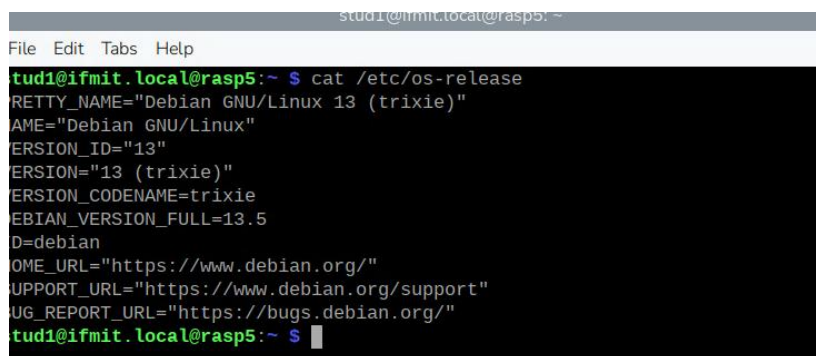
РОЗДІЛ 2 ВІДДАЛЕНИЙ ДОСТУП

2.1 Вступ до віддаленого доступу

Аналіз версії ОС

В умовах впровадження змішаної форми навчання потрібно отримати доступ до Raspberry Pi, через локальну мережу та Інтернет. Або, можливо, у вас немає запасного монітора [21]. Для цього використовують різноманітні додаткові програмні засоби. Всі подальші розробки буде виконувати на наступній версії ОС, особливості якої визначимо за допомогою наступних команд:

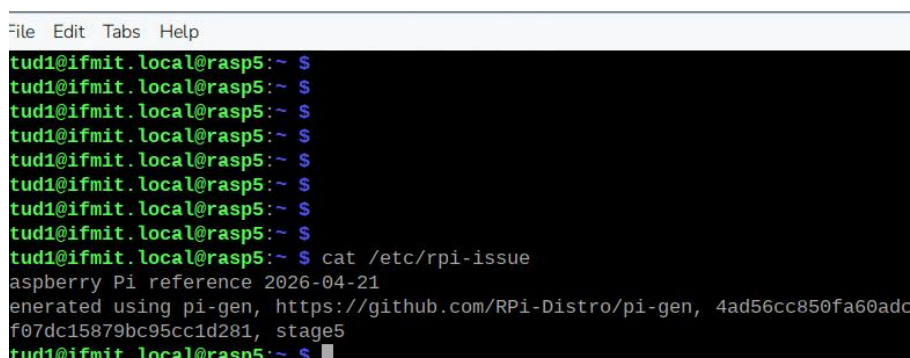
```
cat /etc/os-release
```



```
stud1@ifmit.local@rasp5: ~  
File Edit Tabs Help  
stud1@ifmit.local@rasp5:~ $ cat /etc/os-release  
PRETTY_NAME="Debian GNU/Linux 13 (trixie)"  
NAME="Debian GNU/Linux"  
VERSION_ID="13"  
VERSION="13 (trixie)"  
VERSION_CODENAME=trixie  
DEBIAN_VERSION_FULL=13.5  
ID=debian  
HOME_URL="https://www.debian.org/"  
SUPPORT_URL="https://www.debian.org/support"  
BUG_REPORT_URL="https://bugs.debian.org/"  
stud1@ifmit.local@rasp5:~ $
```

Рис.2.1 Версія ОС

```
cat /etc/rpi-issue
```

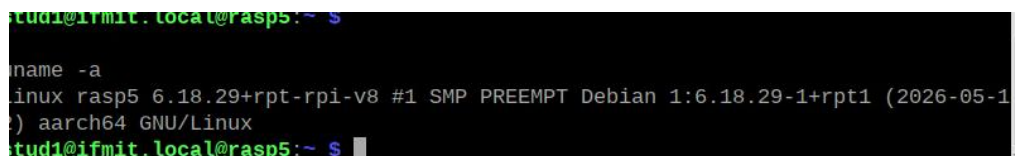


```
File Edit Tabs Help  
tud1@ifmit.local@rasp5:~ $  
tud1@ifmit.local@rasp5:~ $  
tud1@ifmit.local@rasp5:~ $  
tud1@ifmit.local@rasp5:~ $  
tud1@ifmit.local@rasp5:~ $  
tud1@ifmit.local@rasp5:~ $  
tud1@ifmit.local@rasp5:~ $  
tud1@ifmit.local@rasp5:~ $  
tud1@ifmit.local@rasp5:~ $  
tud1@ifmit.local@rasp5:~ $ cat /etc/rpi-issue  
Raspberry Pi reference 2026-04-21  
Generated using pi-gen, https://github.com/RPi-Distro/pi-gen, 4ad56cc850fa60adcf07dc15879bc95cc1d281, stage5  
tud1@ifmit.local@rasp5:~ $
```

Рис. 2.2 Версія ОС

					ИТС.4КІ.0624.03-ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата								
Розроб.		Солоп Д.			Розділ 2			Літ.	Арк.	Акрушів		
Керівник		Могильний Г.А.								3	3	
Реценз.		Козуб Ю.Г.						ЛНУ				
Н. Контр.								Кафедра ИТС, Гр.4КІ				
Зав. каф.		Семенов М.А.										

uname -a



```
stud1@ifmit.local@rasp5:~$  
uname -a  
Linux rasp5 6.18.29+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.18.29-1+rpt1 (2026-05-16) aarch64 GNU/Linux  
stud1@ifmit.local@rasp5:~$
```

Рис.2.3 Версія ОС

Загальні додаткові налаштування

У даному дослідженні до переліку програмного забезпечення, яке встановлено автоматично при розгортанні ОС додатково встановили наступні пакети за допомогою команд:

```
sudo apt-get install mc  
sudo apt-get install net-tools  
sudo apt install traceroute  
sudo apt install libreoffice --- не треба вже інстальовано  
sudo apt-get install remmina remmina-plugin-rdp  
sudo apt install python3 --- не треба вже інстальовано  
sudo apt install python3-pip --- не треба вже інстальовано  
sudo apt-get install cheese  
sudo apt-get install ssh --- не треба вже інстальовано  
sudo apt install chromium-browser --- не треба вже інстальовано  
sudo apt-get install sssd realmd adcli  
sudo apt-get install sssd-tools samba-common libnss-sss libpam-sss  
sudo apt-get install xrdp  
sudo apt-get install tasksel  
sudo apt-get install arduino  
sudo apt-get install libpam-mount cifs-utils  
sudo apt install gxkb - індикатор раскладки
```

Типове програмне забезпечення віддаленого керування

Щоб дистанційно керувати Raspberry Pi з іншого пристрою у локальній мережі, скористайтесь однією з наступних служб, що входять у дистрибутив та можуть бути активовані при інсталуванні[21-24]:

- SSH
- VNC
- Raspberry Pi Connect

SSH (**Secure Shell**) забезпечує безпечний доступ до термінального сеансу на Raspberry Pi.

VNC (Virtual Network Computing) забезпечує безпечний доступ до спільного доступу до екрана робочого столу на Raspberry Pi. Все, що потрібно, це інший комп'ютер, локальна мережа та локальна IP-адреса Raspberry Pi.

Raspberry Pi Connect безпечно надає доступ до екрана Raspberry Pi без необхідності визначення локальної IP-адреси.

Більш ґрунтовний аналіз наукової літератури та джерел Інтернет дозволив знайти інші додаткові програмні засоби:

- **Xrdp**
- NoMachine
- Gnome-remote
- X2go

Ці додаткові програмні засоби мають певні переваги, але потребують окремого дослідження.

В межах цієї роботи будемо створювати та налаштовувати віддалене підключення за допомогою додаткового пакету **XRDP**, а можливості SSH підключення адаптуємо до вимог навчальних комп'ютерних лабораторій.

Щоб знайти локальну IP-адресу Raspberry Pi, скористайтеся одним із наведених нижче методів.

Налаштування локальної IP-адреси

Робочий стіл – знайти локальну IP-адресу

Наведіть курсор на значок мережі в системному треї, і з'явиться підказка. Ця підказка відображає назву мережі, до якої зараз підключені, та IP-адресу.

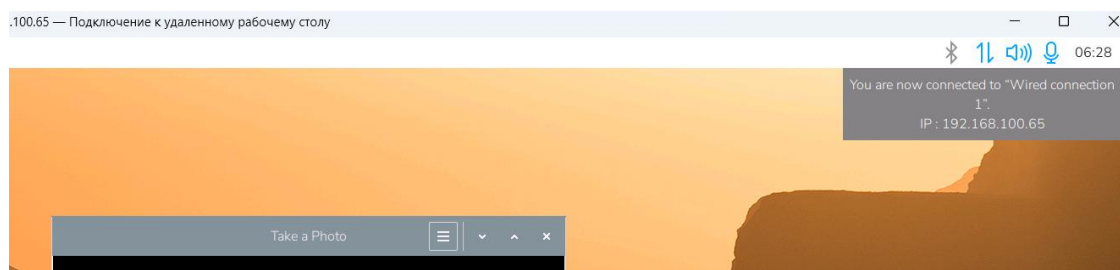


Рис. 2.4 Знайти локальну IP-адресу

Командний рядок – знайти локальну IP-адресу

Виконайте таку команду, щоб вивести локальну IP-адресу в командний рядок:

```
$ hostname
```



```
stud1@ifmit.local@rasp5:~$ hostname
rasp5
stud1@ifmit.local@rasp5:~$
```

Рис. 2.5 Ім'я машини

Якщо використовуєте дисплей з Raspberry Pi та завантажуєтесь з командного рядка замість робочого столу, послідовність завантаження містить IP-адресу як одне з кількох останніх повідомлень перед запитом на вхід.

2.2 Інтеграція списку користувачів з інформаційною системою

Одна з багатьох можливостей отримати доступ до терміналу Raspberry Pi віддалено з іншого комп'ютера з виростанням єдиного, встановленого у інституті математики та інформаційних технологій імені користувача та відповідного паролю – це включення даного мікрокомп'ютеру до домену Microsoft AD

Встановлено, що для цього слід виконати декілька додаткових налаштувань. Всі налаштування необхідно виконати від імені користувача SUDO.

Перевести комп'ютер на статичну адресу

Необхідно налаштувати ім'я комп'ютеру у файлі (рис.2.6)



```
/etc/hostname
ubuntu02
```

Рис. 2.6 Налаштування імені комп'ютеру

Додати DNS основних інформаційних ресурси до файлу hosts (рис. 2.7)

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		29

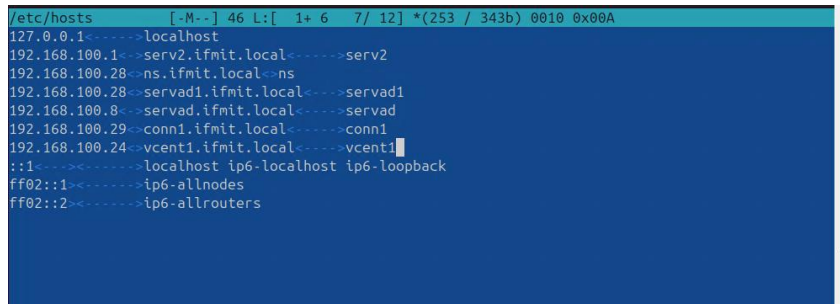


Рис. 2.7 Налаштування DNS для основних ресурсів

Встановити додаткові системні пакети до операційної системи за допомогою наступної команди

```

sudo apt-get update
sudo apt-get upgrade
sudo apt-get install realmd sssd oddjob oddjob-mkhomedir adcli
                        samba-common

```

Перевірити можливість інтеграції з доменом за допомогою команди

```

sudo realm -v discover < ім'я домену>

```

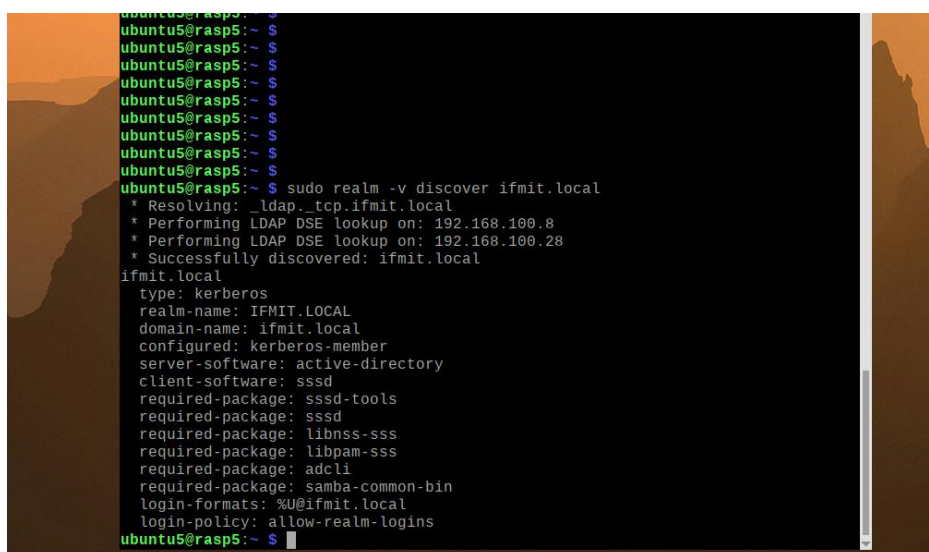


Рис. 2.8 Перевірка налаштувань для підключення до домену

Провести інтеграцію до домену IFMIT

```

sudo realm join -v -U <ім'я_адмініу_домену> ifmit.local

```

Скоригувати процедуру аутентифікації до комп'ютеру

```

sudo pam-auth-update

```

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

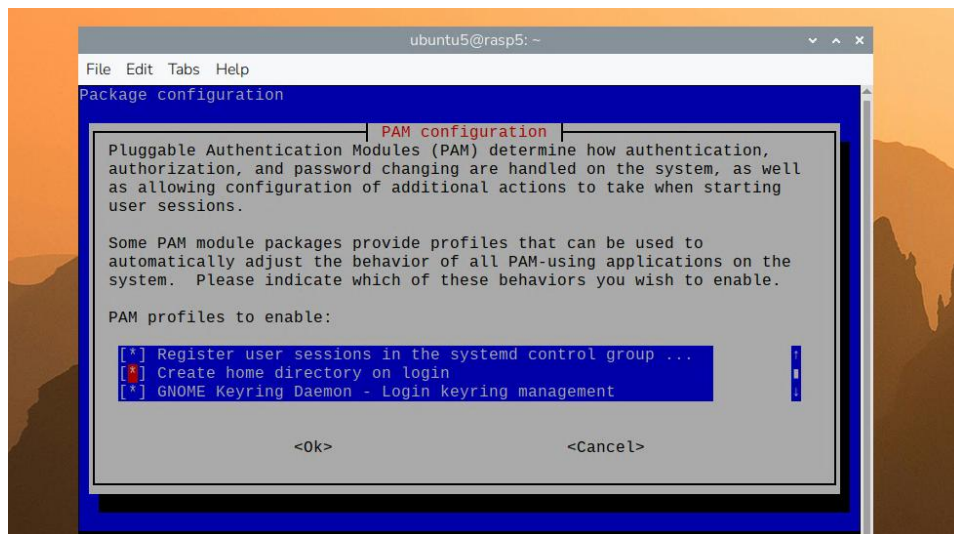


Рис. 2.9 PAM configurator

Перевірити, що операційна система «бачить» користувачів домену

```
getent passwd stud1@ifmit.local
```

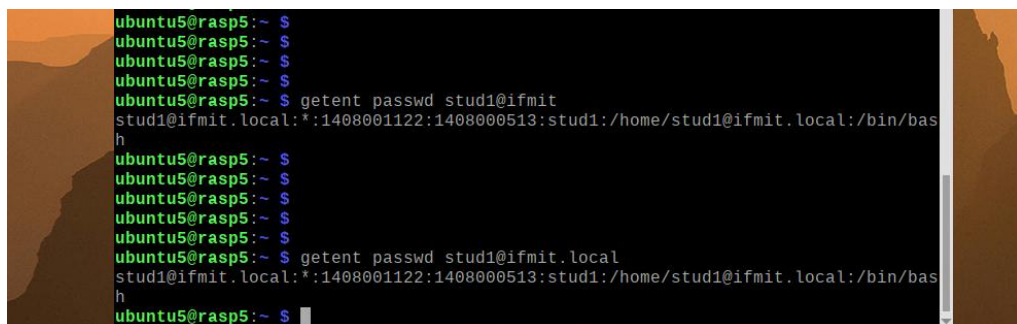


Рис. 2.10 Перевірка результатів підключення до домену Ms AD

Перезавантажити комп'ютер за допомогою команди

```
sudo reboot
```

2.3 Доступ до Raspberry PI через SSH

Одна з багатьох можливостей отримати доступ до терміналу Raspberry Pi віддалено з іншого комп'ютера в тій самій мережі за допомогою протоколу SecureShell (SSH).

Увімкнути SSH-сервер

За замовчуванням, Raspberry Pi OS вимикає SSH-сервер. Увімкніть SSH одним із наведених нижче способів:

1. На робочому столі – From the Preferences menu, launch Raspberry Pi Configuration.

Navigate to the Interfaces tab.

Select Enabled next to SSH.

Click OK.

2. З терміналу (рис.2.11) – Enter `sudo raspi-config` in a terminal window.

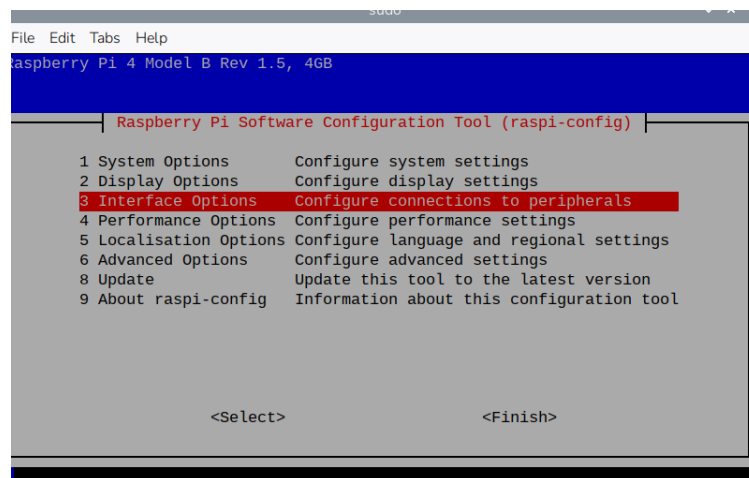
Select Interfacing Options.

Navigate to and select SSH.

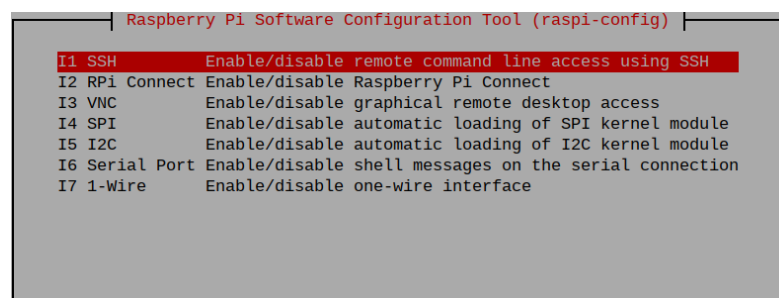
Choose Yes.

Select Ok.

Choose Finish.



a)



б)

Рис. 2.11 Налаштування SSH

3. Вручну Створіть порожній файл з назвою `ssh` в розділі завантаження:

```
$ sudo touch /boot/firmware/ssh
```

4. Перезавантажте машину:

```
$ sudo reboot
```

Підключення до SSH-сервера

Відкрийте вікно терміналу на комп'ютері та введіть таку команду, замінивши `<ip address>` тимчасове поле IP-адресою Raspberry Pi, до якого намагаєтеся підключитися, та `<username>` своїм ім'ям користувача:

```
$ ssh <username>@<ip address>
```

Коли з'єднання запрацює – побачите попередження безпеки. Введіть код `yes`, щоб продовжити. Побачите це попередження лише під час першого підключення.

Введіть пароль свого облікового запису, коли буде запропоновано.

Тепер необхідно побачити командний рядок Raspberry Pi:

```
<username>@<hostname> ~ $
```

Тепер створено підключення до Raspberry Pi віддалено та можете виконувати команди.

У випадку, коли підключаєтеся доменним користувачем слід використовувати інший синтаксис

```
$ ssh -l <username>@<domen> <ip address>
```

Перевіряємо SSH з'єднання від `root` користувача

```
C:\Users\new20>ssh -l ubuntu5 192.168.100.65
ubuntu5@192.168.100.65's password:
Linux rasp5 6.18.29+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.18.29-1+rpt1 (2026-05-12) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu May 28 17:21:12 2026 from 192.168.99.127
ubuntu5@rasp5:~$
```

Рис. 2.12 Перевірка з'єднання rootкористувача

Перевіряємо ssh для інших користувачів

```
C:\Users\new20>ssh -l stud6@ifmit 192.168.100.65
stud6@ifmit@192.168.100.65's password:
Linux rasp5 6.18.29+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.18.29-1+rpt1 (2026-05-12) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
stud6@ifmit.local@rasp5:~$
```

Рис. 2.13 Перевірка з'єднання користувачів домену

Налаштування обмежень на використання SSH

У навчальному середовищі бажано надати можливість використання SSH для локальних адміністраторів (root) та адміністраторів домену AD.

Щоб дозволити підключення по SSH тільки одному конкретному користувачеві (і повністю заборонити всім іншим, включаючи root), необхідно відредагувати основний конфігураційний файл SSH-сервера — /etc/ssh/sshd_config.

Для цього використовується директива AllowUsers. Як тільки додаєте цю директиву, SSH-сервер автоматично переходить у режим "білого списку": доступ дозволяється лише тим, хто вказаний у цьому рядку, а всі інші користувачі системи блокуються за замовчуванням.

Нижче наведено оптимальні варіанти налаштування залежно від впроваджених вимог до безпеки.

Варіант 1. Суворе обмеження за іменем. Якщо вам потрібно дозволити доступ користувачеві (наприклад, з іменем developer) з будь-якої IP-адреси:

1. Відкрийте конфігураційний файл для редагування для цього у Bash введіть команду:

```
sudo nano /etc/ssh/sshd_config
```

2. Знайдіть кінець файлу або секцію з авторизацією та додайте рядок:

```
AllowUsers <ім'я користувача>
```

3. Задля безпеки переконайтеся, що вхід для користувача root також закритий, знайшовши або додавши параметр (для навчальних лабораторій цей параметр можна не застосовувати):

```
PermitRootLogin no
# для навчальних лабораторій
PermitRootLogin yes
```

Варіант 2. Максимальна безпека (Обмеження за іменем + IP-адресою) Для підвищеної системи безпеки можливо щоб цей єдиний користувач міг підключитися тільки з конкретної робочої машини (наприклад, з IP-адреси 192.168.100.50), директиві AllowUsers можна передати синтаксис користувач@IP:

```
AllowUsers developer@192.168.100.50
```

При такій конфігурації, навіть якщо хтось дізнається пароль або викраде SSH-ключ користувача developer, він не зможе увійти, якщо перебуває в іншій підмережі.

Варіант 3. Обмеження за іменем для конкретної підмережі Якщо у локальній лабораторії чи офісі IP-адреси динамічні, але знаходяться в одному діапазоні (наприклад, 192.168.100.X), можна використовувати маску:

```
AllowUsers developer@192.168.100.*
```

При редагуванні файлу /etc/ssh/sshd_config важливий покроковий алгоритм застосування змін щоб випадково не заблокувати самому собі доступ до приладу (особливо якщо налаштовуєте його віддалено), виконуйте перевірку за цим алгоритмом:

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35

1. Внесіть зміни у файл `/etc/ssh/sshd_config` (наприклад, додайте `AllowUsers adminuser`).

2. Обов'язково перевірте синтаксис конфігурації перед перезапуском служби, виконавши команду у Bash:

```
sudo sshd -t
```

Якщо команда нічого не вивела — все налаштовано без помилок. Якщо вивела помилку — виправте її у файлі.

3. Перезапустіть службу SSH, щоб застосувати конфігурацію виконавши команду у Bash:

```
sudo systemctl restart ssh
# або на деяких дистрибутивах (CentOS/RHEL):
sudo systemctl restart sshd
```

4. НЕ ЗАКРИВАЙТЕ поточну сесію терміналу! Відкрийте нове окреме вікно терміналу (або нову вкладку RDP/SSH) і спробуйте підключитися під дозволеним користувачем. Якщо підключення пройшло успішно — роботу завершено. Якщо виникла помилка, зможете виправити її у першому (все ще відкритому) вікні.

2.4 Особливості системних налаштувань при використанні

XRDP

Перевірка з'єднання та загальні проблеми

Для перевірки з'єднання скористуємось стандартним програмним додатком «Підключення до віддаленого робочого столу» - клієнт RDP (рис. 2.14)

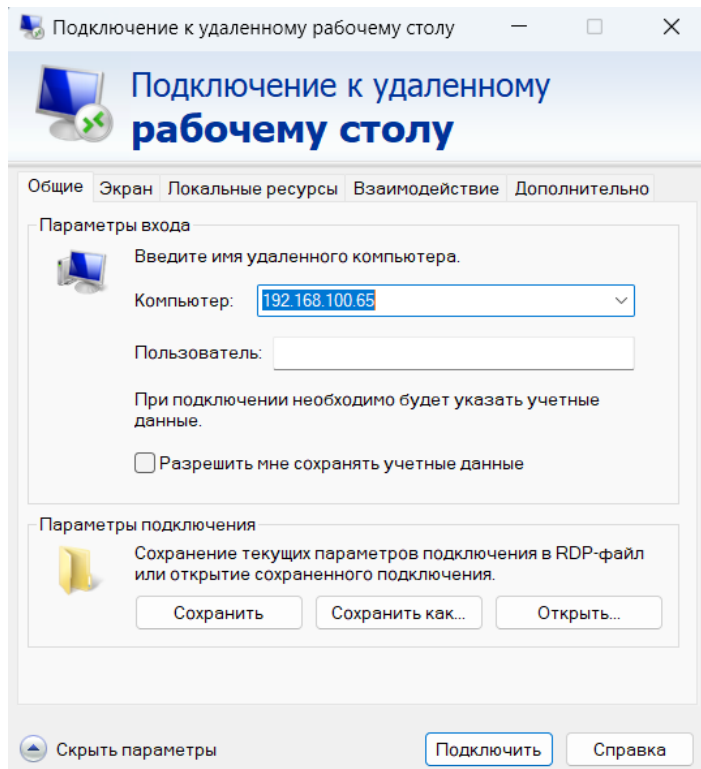


Рис. 2.14 Загальний вигляд клієнту RDP

Налаштування клієнту зображені на рис. 2.15-2.16

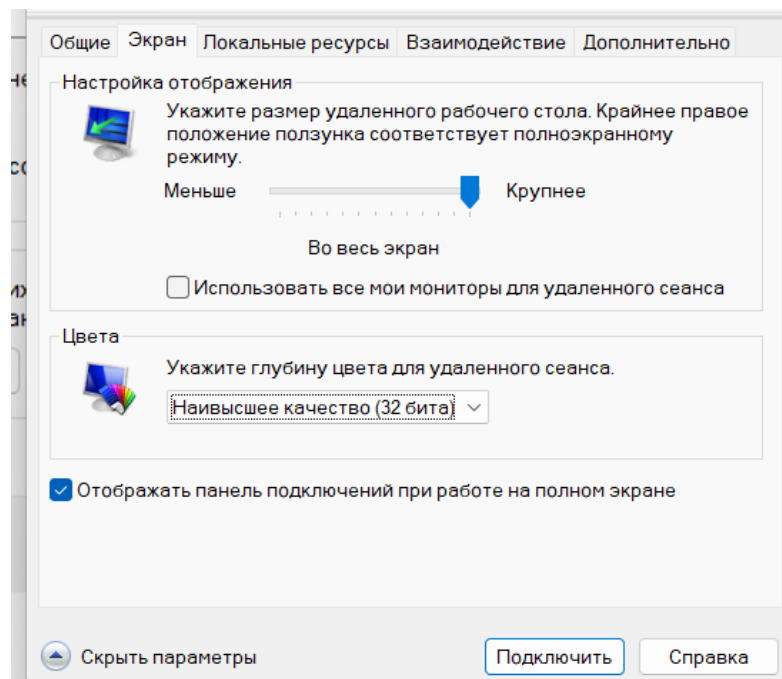


Рис. 2.15 Параметры экрана

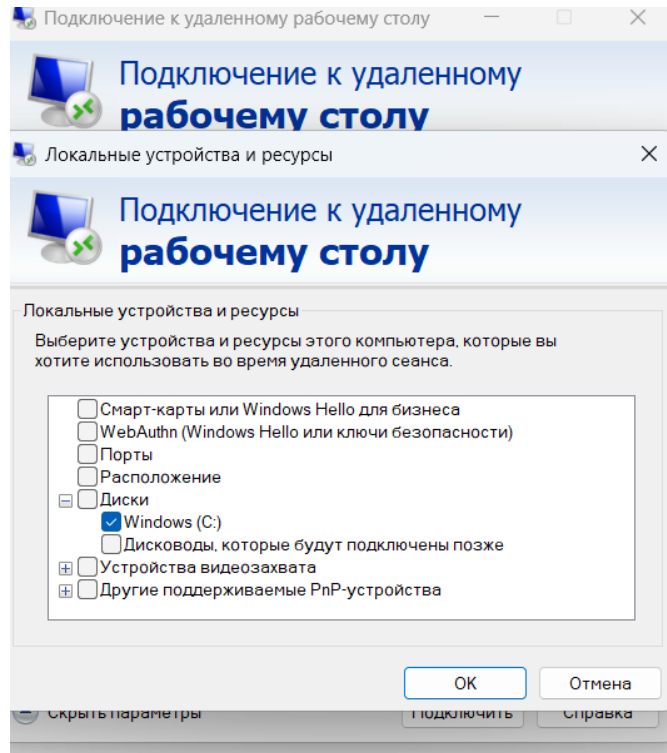


Рис. 2.16 Параметры локальных ресурсов

В процесі використання віддалено доступу з'ясувалося:

- Доменні користувачі не можуть підключитися
- Проблеми використання репозиторію – виникає помилка для усіх користувачів
- Не працює команда завершення сеансу
- Користувачі не мають права використовувати зовнішні пристрої/порти та камеру
- В деяких випадках після відключення користувача його сесія не завершується та чекає завершення фонових процесів, що запустив користувач – треба примусово завершувати всі процеси користувача

Крім цього, в процесі використання та налаштування з'ясувалися додаткові особливості, що пов'язані з організацією навчального процесу.

- Необхідно обмежити кількість одночасних підключень до Raspberry PI – одне віддалене підключення
- Відсутність автозавантаження додатку для використання камери, для перегляду результатів роботи зовнішніх приладів.

- Відсутність автопідключення до зовнішніх мережевих дисків з домену Ms AD для передавання результатів роботи здобувачем освіти та використання додаткового методичного забезпечення
- Періодичне переповнення навантаження на процесор – необхідно періодично перезавантажувати систему за певним розкладом, наприклад, у перервах між навчальними заняттями
- При виході користувача з системи (відключенні) необхідно також перезавантажити Raspberry PI для скидання налаштувань на зовнішніх портах та інших програмних додатках

Для вирішення цього питання було проведено додатковий аналіз літературних джерел.

Налаштування використання зовнішніх пристроїв/ портів

В процесі використання різноманітних варіантів віддалено доступу з'ясувалося, що користувачі не мають права використовувати зовнішні порти.

Для вирішення цього питання було проведено додатковий аналіз літературних джерел.

Встановлено, що є можливість скористатися спеціальним керуванням за допомогою компоненту UDEV [25]

За допомогою спеціальних файлів є можливість призначити певний доступ до зовнішніх приладів/портів.

Необхідно створити додаткові правила.

Правила udev зберігаються в папці /etc/udev/rules.d . Файл правил обов'язково повинен мати розширення .rules . Зазвичай у цій папці вже є кілька файлів udev rules, але їх чіпати не рекомендується, для своїх правил краще створити окремий файл, наприклад:

```
touch /etc/udev/rules.d/dop.rules
```

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		39

Правило udev складається з декількох пар ключ - значення , розділених комою. Одні ключі використовуються для перевірки відповідності пристрою певному правилу. У таких ключах використовується знак == для поділу пари, наприклад: SUBSYSTEM == "block" . Це означає, що правило буде застосовано тільки якщо значення ключа SUBSYSTEM для цього пристрою дорівнює block . Інші ключі використовуються для вказівки, якщо всі умови відповідності виконуються. Для розділення пар у таких ключах використовується знак дорівнює "=", наприклад, NAME = "mydisk" . Ну і цілком правило:

```
SUBSYSTEM=="block", ATTR(size)=="1343153213", NAME="mydisk"
```

- SUBSYSTEM – підсистема пристрою;
- KERNEL - ім'я, яке видається пристрою ядром;
- DRIVER - драйвер, який обслуговує пристрій;
- ATTR – sysfs-атрибут пристрою;
- SUBSYSTEMS – підсистема батьківського пристрою.

Пристрій може мати батьківські пристрої, наприклад, жорсткий диск має батьківський пристрій SCSI , який, у свою чергу, має батьківський пристрій - шину BUS . Іноді потрібно отримати інформацію від батьківського пристрою. Для цього використовуються ключі SUBSYSTEMS, KERNELS, DRIVERS, ATTRS відповідно.

Для дій використовуються ключі:

- NAME – встановити ім'я файлу пристрою;
- SYMLINK – альтернативне ім'я пристрою;
- RUN – виконати скрипт при підключенні пристрою;
- GROUP - група, яка має доступ до файлу;
- OWNER – власник файлу пристрою;
- MODE – маска прав доступу.

В результаті проведення налаштування для використання ВЕБ камери та зовнішнього мікроконтролера – файл Arduino.rules

- `UserWindowManager=startwm.sh` — задає ім'я скрипта запуску графічної оболонки в домашній директорії користувача (шлях вказується відносно домашнього каталогу).

- `DefaultWindowManager=startwm.sh` — якщо у користувача немає індивідуального скрипта, буде запущено цей глобальний сценарій за замовчуванням (відносний або повний шлях до `/etc/xrdp`). Саме він відповідає за те, яке графічне середовище (XFCE, MATE, GNOME) завантажиться у користувача.

- `ReconnectScript=reconnectwm.sh` — скрипт, який виконується, коли користувач перепідключається до вже існуючої (раніше від'єднаної) сесії.

2. Безпека та авторизація ([Security])

Параметри, що контролюють права доступу та обмеження при авторизації:

- `AllowRootLogin=true` — дозволяє користувачу `root` підключатися через RDP. У продакшен-середовищах цей параметр часто рекомендується вимикати (`false`) з міркувань безпеки.

- `MaxLoginRetry=4` — максимальна кількість спроб введення пального. Якщо користувач помилиться 4 рази поспіль, сесія підключення перерветься.

- `TerminalServerUsers=tsusers` та `TerminalServerAdmins=tsadmins` — визначають групи локальних користувачів у Linux, яким дозволено звичайний та адміністративний доступ до термінального Raspberry PI.

- `AlwaysGroupCheck=false` — якщо встановлено `false`, перевірка членства в групі `tsusers` ігноруватиметься, якщо сама група не створена в системі (тобто доступ буде дозволено всім легітимним користувачам).

3. Політика виділення та життєвого циклу сесій ([Sessions])

`X11DisplayOffset=10`

- Тип даних: Ціле число (Integer).
- Значення за замовчуванням: 10.

- **Опис:** Визначає зміщення (початковий номер) для графічних дисплеїв X11, які виділяються користувачам. Перша створена XRDP-сесія отримає номер дисплея :10 (що відповідає мережевому TCP-порту 6010), наступна — :11 і так далі. Це потрібно для того, щоб віртуальні сесії не конфліктували з локальними дисплеями Raspberry PI (наприклад, фізичний монітор зазвичай займає дисплей :0 або :1).

MaxSessions=1

- **Тип даних:** Ціле число (Integer).
- **Значення за замовчуванням:** 0 (без обмежень).
- **Опис:** Встановлює максимальну кількість одночасних RDP-підключень (активних сесій), яку може обслуговувати цей XRDP-сервер. У цьому випадку встановлено ліміт у 1 сесію. Як тільки цей ліміт буде вичерпано, нові користувачі не зможуть підключитися, поки хтось не завершить роботу.

#MaxDisplayNumber=63

- **Тип даних:** Ціле число (Integer).
- **Значення за замовчуванням:** 63.

Параметр закоментований символом #, тому діє значення за замовчуванням). Він визначає максимальний номер дисплея X11, який XRDP може виділити сесії. Обмеження 63 пов'язане з тим, що організація IANA офіційно виділяє TCP-порти для X-серверів лише до порту 6063. Якщо вимкнути мережеві TCP-підключення до X-серверів, це число можна безпечно збільшити.

KillDisconnected=false

Встановлюємо KillDisconnected=true

- **Тип даних:** Логічний (Boolean: true/false, 1/0, yes/no).
- **Значення за замовчуванням:** false.
- **Опис:** Визначає долю сесії після того, як користувач відключився (наприклад, просто закрив хрестиком вікно RDP-клієнта, не виходячи з системи через "Log Out").

○ При `false` (як у вас): сесія залишається активною ("замороженою") на Raspberry PI, програми продовжують виконуватися. Користувач може підключитися пізніше і продовжити роботу.

○ При `true`: сесія та всі запуснені в ній програми будуть повністю примусово закриті (вбиті) через час, указаний у параметрі `DisconnectedTimeLimit`.

`DisconnectedTimeLimit=0`

Встановлюємо `DisconnectedTimeLimit=120`

- Тип даних: Ціле число (секунди).
- Значення за замовчуванням: 0.
- Опис: Час очікування перед повним знищенням відключеної сесії. Якщо параметр `KillDisconnected` встановлено у значення `false`, цей параметр повністю ігнорується. Коли `KillDisconnected` був активований, то значення 0 означало б миттєве знищення сесії відразу після втрати зв'язку.

`IdleTimeLimit=0`

Встановлюємо `IdleTimeLimit=1200`

- Тип даних: Ціле число (секунди).
- Значення за замовчуванням: 0.
- Опис: Тайм-аут бездіяльності користувача. Визначає, скільки часу сесія може простоювати без будь-яких дій (рухів миші, натискань клавіш), перш ніж Raspberry PI примусово від'єднає користувача. Значення 0 повністю вимикає авто-відключення за простій.

`Policy=Default`

- Тип даних: Перерахування (Enum).
- Доступні варіанти: `Default`, `Separate` або комбінація літер із набору `{UBDI}`.
- Опис: Політика виділення та розподілу сесій. Визначає логіку: створювати користувачу новий робочий стіл чи підключати до старого.
 - `Default` (застосовано у вас): наразі працює так само як режим UB.

- **Separate:** усі сесії суворо розділені. Користувач ніколи не зможе повернутися до своєї старої відключеної сесії (завжди створюється нова, а старі треба чистити вручну або іншими налаштуваннями).
- **Комбінаційні опції (літери):**
 - **U (User):** Сесії розділяються за іменем користувача.
 - **B (Bits-per-pixel):** Сесії розділяються за глибиною кольору RDP-клієнта.
 - **D (Display size):** Сесії розділяються за початковим роздільною здатністю (розміром екрана).
 - **I (IP address):** Сесії розділяються за IP-адресою, з якої підключається клієнт.

При цьому треба враховувати, що опції **U** та **B** завжди активні за замовчуванням і їх не можна вимкнути. Режим **Default (UB)** гарантує: якщо користувач **alex** підключився з глибиною кольору 24 біти, закрив вікно, а потім знову зайшов як **alex** із тими ж 24 бітами — він гарантовано повернеться у свій старий робочий стіл.

4. Перенаправлення ресурсів та канали ([Chansrv] / [SessionVariables])

Ці налаштування відповідають за інтеграцію локального оточення користувача з видаленим сервером:

- **RestrictOutboundClipboard=none** та **RestrictInboundClipboard=none** (у секції **[Security]**) — повністю дозволяють двосторонній обмін даними через буфер обміну (текст, файли, зображення) між клієнтським комп'ютером і сервером XRDP.

- **FuseMountName=thinclient_drives** — активує перенаправлення локальних дисків (клієнтських папок) всередину видаленої сесії Linux за допомогою підсистеми FUSE. Спільні папки будуть змонтовані в домашній директорії користувача під вказаним ім'ям.

- `FileUmask=077` — маска прав доступу для змонтованих дисків. Значення 077 гарантує, що доступ до перенаправлених файлів матиме виключно сам власник сесії.

- `PULSE_SCRIPT=/etc/xrdp/pulse/default.pa` — змінна оточення, яка вказує аудіосерверу PulseAudio використовувати специфічний скрипт XRDP для перенаправлення звуку з видаленої Linux-системи на динаміки клієнтського комп'ютера.

Налаштування перезавантаження за розкладом

Налаштувати перезавантаження Linux-системи за розкладом найпростіше і надійніше за все за допомогою вбудованого планувальника завдань Cron. Він працює на рівні системи та дозволяє гнучко задавати час (хвилини, години, дні тижня тощо).

Щоб перевірити, чи встановлена та чи працює служба `cron` (планувальник завдань) у Linux-системі, зазвичай використовують кілька стандартних термінальних команд.

Ось детальний алгоритм перевірки:

Крок 1. Перевірка статусу служби (чи працює вона зараз)

У більшості сучасних дистрибутивів (Ubuntu, Debian, Linux Mint, CentOS, RHEL) керування службами відбувається через `systemd`. Виконайте команду:

```
sudo systemctl status cron
```

Оскільки перезавантаження вимагає прав суперкористувача, налаштування потрібно робити в конфігураційній таблиці (`crontab`) користувача `root`.

Покрокова інструкція з налаштування

1. Відкрийте планувальник `crontab` для користувача `root` за допомогою команди:

```
sudo crontab -e
```

Якщо запускаєте цю команду вперше, система може запропонувати вибрати текстовий редактор. Виберіть nano (зазвичай під номером 1), він найпростіший.

2. Перейдіть у самий кінець файлу та додайте рядок із розкладом.

Приклади готових шаблонів розкладу:

- Кожного дня о 03:00 ночі:

```
0 3 * * * /sbin/shutdown -r now
```

- Кожного понеділка о 04:30 ранку:

```
30 4 * * 1 /sbin/shutdown -r now
```

- Першого числа кожного місяця о 02:00 ночі:

```
0 2 1 * * /sbin/shutdown -r now
```

- Кожні 3 дні о Midnight (00:00):

```
0 0 */3 * * /sbin/shutdown -r now
```

3. Збережіть файл та вийдіть з редактора:

Якщо це nano: натисніть Ctrl + O, потім Enter (для збереження) і Ctrl + X (для виходу).

4. Після виходу побачите повідомлення: crontab: installing new crontab. Це означає, що розклад успішно активовано.

Всі зміни, які вносите в конфігураційну таблицю crontab, записуються в спеціальний текстовий файл на диску Raspberry PI (для користувача root в Ubuntu/Debian цей файл фізично знаходиться на шляху /var/spool/cron/crontabs/root).

Щоб була можливість налаштувати власний час, подивіться на структуру п'яти зі-рочок у Cron:

```
.----- хвилини (0 - 59)
| .----- години (0 - 23)
| | .----- день місяця (1 - 31)
| | | .----- місяць (1 - 12)
| | | | .---- день тижня (0 - 6) (0 або 7 = не-діля, 1 =
| | | | |                             понеділок...)
| | | | |
* * * * * [КОМАНДА]
```

При цьому враховуйте, що у Cron краще завжди вказувати повний шлях до бінарних файлів (наприклад, /sbin/shutdown або /sbin/reboot), оскільки змінні оточення PATH у планувальника мінімальні, і проста команда reboot може не спрацювати.

Існує альтернативний спосіб перезавантаження системи – безпечне перезавантаження через systemctl. На Raspberry PI працює сучасна система ініціалізації systemd (Ubuntu 16.04 і новіші, Debian, CentOS), замість старого shutdown можна викликати команду керування системними службами. Рядок у crontab виглядатиме так:

```
0 3 * * * /bin/systemctl reboot
0 6-22/2 * * * /bin/systemctl reboot
```

Він діє ідентично, але вважається більш нативним для сучасних дистрибутивів.

Щоб подивитися поточний розклад перезавантаження, не відкриваючи редактор:

```
sudo crontab -l
```

- Щоб видалити або змінити розклад, знову виконайте `sudo crontab -e` і просто видаліть або закоментуйте (поставте # на початку) відповідний рядок.

Однак даній ОС існує **альтернативний спосіб** – є можливість відредагувати файл /etc/crontab. Файл /etc/crontab — це системна таблиця планувальника задач Cron (системний crontab). Він використовується операційною системою для автоматичного запуску фонових скриптів, утиліт обслуговування дисків, оновлень та інших адміністративних завдань від імені різних користувачів.

На відміну від таблиць користувача (які редагуються через crontab -e), цей файл має кілька важливих особливостей у використанні та синтаксисі.— головна особливість – наявність поля USER

Якщо у звичайному користувальницькому crontab -е рядок складається з 5 полів часу і самої команди, то /etc/crontab додається шосте поле — ім'я користувача, від імені якого потрібно запустити завдання. Синтаксис рядка в /etc/crontab: хвилина година день_місяця місяць день_тижня КОРИСТУВАЧ коман-да

Приклад реального запису:

```
0 1 * * * root /usr/sbin/logrotate /etc/logrotate.conf
0 6-22/2 * * * root /bin/systemctl reboot
```

У останньому випадку завдання на перевантаження запускається кожного дня, кожного місяця, кожного дня тижня, кожного другого часу в період з 6 годин до 22 години.

Налаштування перезавантаження після виходу

Тепер потрібно змінити сценарій /etc/xrdp/startwm.sh, який керує запуском графічної оболонки користувача. Обернемо запуск сесії в таймер, а відразу після завершення цього процесу (за будь-якої причини) додамо команду перезавантаження комп'ютера.

Відкрийте скрипт запуску

```
sudo nano /etc/xrdp/startwm.sh
```

Знайдіть в самому кінці файлу рядок, який запускає сесію. Зазвичай це:

```
. /etc/X11/Xsession
```

Замініть цю фінальну частину на наступний блок коду (наприклад, обмежимо час роботи користувача 3 годинами – 3h):

Замість старих рядків:

```
test -x /etc/X11/Xsession && exec /etc/X11/Xsession
exec /bin/sh /etc/X11/Xsession
```

Вставте наступний код:

```
#
# 1. Запуск сесій с обмеженням 1 час 57 минут (117m)
```

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49

```
# убрали 'exec', щоб скрипт не преривался, а йшов далі після
                                закриття стола
if [ -x /etc/X11/Xsession ]; then
    timeout 117m /etc/X11/Xsession
else
    timeout 117m /bin/sh /etc/X11/Xsession
fi

# 2. Сюди керування переходЕ АВТОМАТИЧНО за будь-якого сценаріі:
# - Минули 1 годину 57 хвилин
# - Користувач сам натиснув "Log Out" (достроковий вихід)
# - Пройшло 120 секунд після натискання на хрестик (sesman убив
                                сесію)
# додаємо КОМАНДУ ПЕРЕЗАВАНТАЖЕННЯ:
sudo /bin/systemctl reboot
```

Додатково необхідно надати дозвіл користувачам викликати перезавантаження. Скрипт startwm.sh виконується з правами поточного зареєстрованого користувача, а не root. За промовчаням звичайному користувачеві Linux заборонено виконувати команду systemctl reboot без введення пароля адміністратора через sudo. Щоб перезавантаження спрацювало автоматично, потрібно надати користувачам таке право.

Відкрийте файл конфігурації sudoers через спеціальний безпечний редактор:

```
sudo visudo
```

Додайте в самий кінець файлу рядок, який дозволяє всім користувачам (або конкретній групі) виконувати перезавантаження без пароля:

```
ALL ALL=(ALL) NOPASSWD: /bin/systemctl reboot
```

Якщо хочете дозволити це лише конкретній групі користувачів (наприклад, students), напишіть

```
%students ALL=(ALL) NOPASSWD: /bin/systemctl reboot.
```

Збережіть файл та закрийте файл (в visudo для Nano: Ctrl+O, Enter, Ctrl+X).

Налаштування процесу входу

При вході користувача виникає помилка (рис. 2.19)

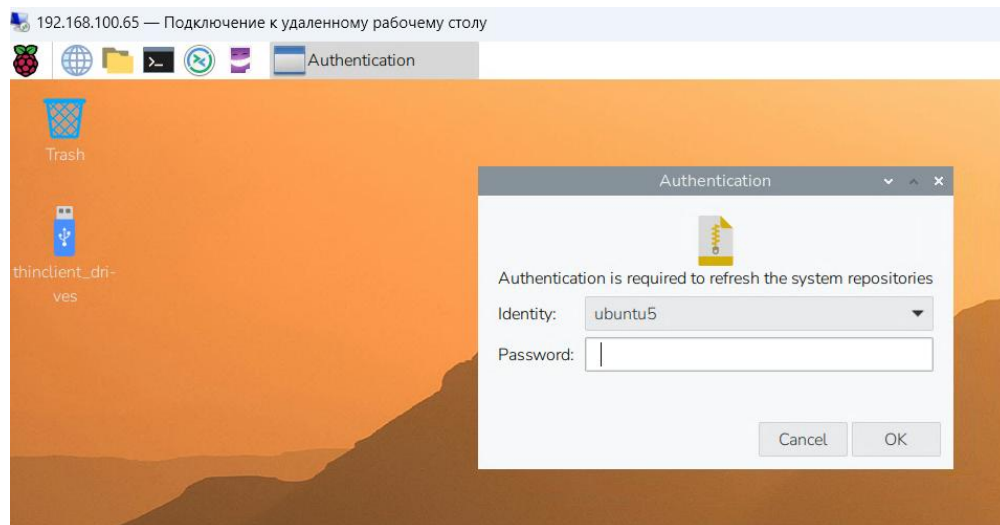


Рис.2.19 Помилка при вході

Це класична проблема Polkit (PolicyKit) при використанні RDP-сесій.

Коли підключаєтеся через xrdp, система вважає цю сесію «віддаленою» (inactive), а не локальною. За замовчуванням політики безпеки Linux дозволяють оновлювати репозиторії або керувати живленням без пароля лише користувачеві, який сидить за фізичним монітором та клавіатурою. Для RDP-користувача (навіть із правами sudo/root) система запитує підтвердження.

У сучасних графічних оболонках (як-от на Raspberry Pi) за лаштунками працює служба PackageKit. Вона відповідає за перевірку оновлень, щоб система завжди була актуальною. Логіка системи така:

Коли сидите за комп'ютером фізично (монітор + клавіатура), Polkit вважає сесію «локальною» та довіреною. Він дозволяє перевірку оновлень без пароля. Коли студент заходить через XRDP, Polkit бачить це як «віддалене підключення». З погляду безпеки це ризик: раптом хтось сторонній через мережу намагається щось змінити? Тому система каже: «Гей, я хочу перевірити репозиторії, але ти зайшов віддалено. Доведи, що ти адмін — введи пароль». Оскільки студенти — не адміни, вони застрягають із цим вікном, яке не можна просто так прибрати. Це створює відчуття «зламаної» системи.

Для вирішення цієї проблеми пропонується створити правило винятку.

Ми явно скажемо системі: «Якщо хтось просить оновити список пакетів (не

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

встановити, а саме оновити список) — дозволяй це будь-якому авторизованому користувачеві, навіть якщо він зайшов через RDP».

Конкретні дії для Debian: Оскільки ми використовуємо саме 13-ту версію, правила пишуться мовою JavaScript. Створюємо файл правила:

```
sudo nano /etc/polkit-1/rules.d/45-allow-xrdp-fixes.rules
```

Вставляємо туди цей код:

```
polkit.addRule(function(action, subject) {  
    if ((action.id == "org.freedesktop.packagekit.system-sources-  
        refresh" ||  
        action.id == "org.freedesktop.packagekit.system-network-  
            proxy-configure" ||  
        action.id.indexOf("org.freedesktop.color-manager.") == 0)  
        &&  
        subject.active) {  
        return polkit.Result.YES;  
    }  
});
```

Цей код виконує наступне:

org.freedesktop.packagekit.system-sources-refresh — прибирає те саме вікно з репозиторіями.

org.freedesktop.color-manager. — прибирає друге за популярністю вікно «Authentication is required to create a color profile» (воно часто вилітає наступним).

subject.active — гарантує, що правило спрацює тільки для того, хто реально залогінився і чия сесія зараз активна.

Застосовуємо: Після збереження файлу перезавантажте службу:

```
sudo systemctl restart polkit
```

Це правило не дає студентам прав встановлювати програми чи видаляти щось системне. Воно дозволяє лише «refresh» (оновити список доступного софту). Для реальних змін (install/remove) система все одно вимагатиме пароль адміністратора. Результат: студент заходить в RDP і відразу бачить чистий робочий стіл без зайвих запитань.

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

Крім того з’ясувалось, що користувачі домену не можуть підключитися (рис.2.20)

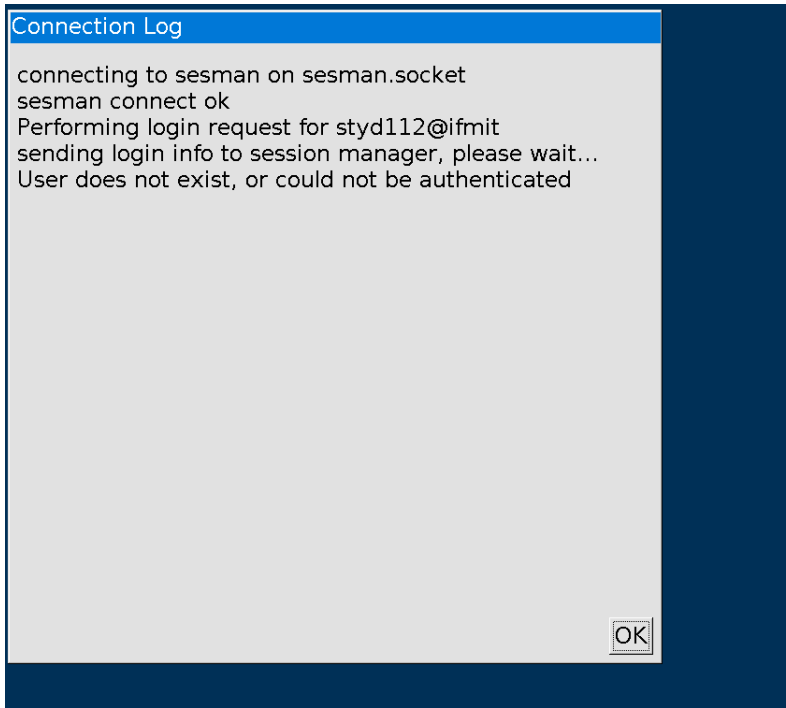


Рис.2 20 Помилка входу доменних користувачів

В процесі аналізу з’ясувалось наступне.

Raspberry Pi завантажує групові політики (GPO) з контролера домену Windows. Але в цих політиках немає правила конкретно для служби xrdp-sesman. Оскільки SSSD не знає, чи дозволено цій службі пускати користувачів, він обирає найбезпечніший варіант: заборонити все. Для виправлення є можливість явно сказати системі, що xrdp-sesman — це тип віддаленого входу, і до нього потрібно застосовувати ті самі правила, що й до звичайного RDP у Windows.

У файлі `/etc/sss/sss.conf` у секції домену додайте:

```
ad_gpo_map_remote_interactive = +xrdp-sesman
```

Це додасть xrdp до списку сервісів, яким дозволено вхід, якщо у Windows GPO дозволено "Remote Desktop" для цього користувача.

Після редагування файлу SSSD потрібно обов'язково зробити дві речі, інакше служба не запуститься або не побачить змін:

1. Встановіть жорсткі права на файл (вимога безпеки SSSD):

```
sudo chmod 600 /etc/sss/sss.conf
```

2. Очистіть кеш і перезапустіть службу:

```
sudo systemctl stop sssd  
sudo rm -f /var/lib/sss/db/*  
sudo systemctl start sssd
```

Виправлення помилок виходу

При виході користувача та завершенні процесу роботи виникає помилка – процедура виходу не працює крім того, у деяких випадках після завершення роботи поточна сесія не закривається.

Для вирішення цього завдання необхідно переключити роботу дисплею на режим X11.

За допомогою `raspi-config` переключитися на X11 (рис.2. 21, 22)

```
Sudo raspi-config
```

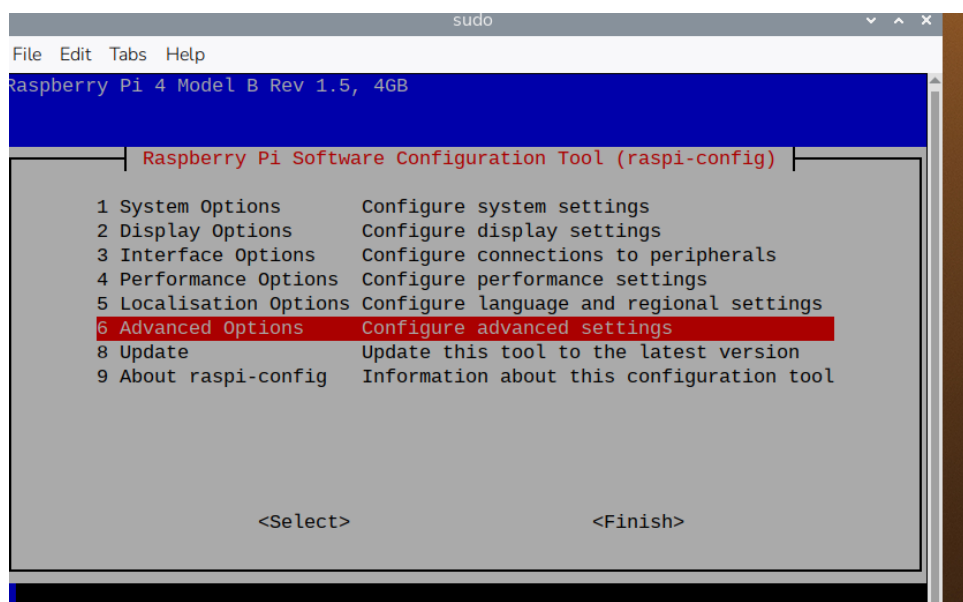


Рис 2. 21 Початкове вікно `raspi-config`

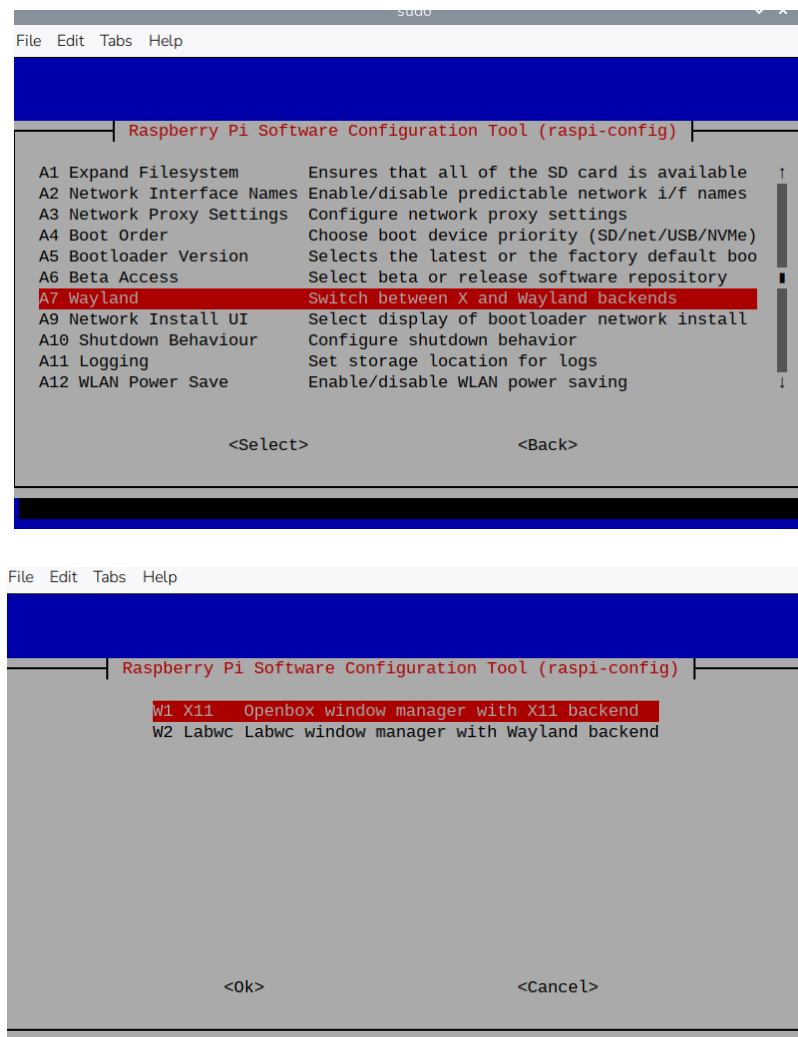


Рис. 2. 22 Налаштування переключення на X11

Для автоматичного завершення всіх сесій після завершення пропонується налаштувати ОС на примусове завершення всіх процесів, що запущені користувачем. У такому випадку всі користувачі та адміністратор не будуть в змозі зробити фоновий процес, який буде працювати після виходу користувача з ОС. Для цього необхідно відредагувати файл

etc/systemd/logind.conf

#змінити параметр KillUserProcesses=no
KillUserProcesses=yes

Налаштування автозавантаження програм

Налаштування автозагрузки програм для всіх користувачів у Linux (особливо в середовищах віддаленого робочого столу, таких як XRDP, де найчастіше використовуються легковагові оболонки на кшталт XFCE, MATE або стандартний GNOME) залежить від того, чи хочете запустити графічний додаток (GUI), чи консольний скрипт у фоні.

Розглянемо 3 найнадійніші способи реалізації глобальної автозагрузки.

Спосіб 1. Через XDG Autostart – це найкраще для графічних програм: браузері, месенджери, панелі. Цей спосіб відповідає стандартам freedesktop.org і працює для будь-якого сучасного графічного середовища (GNOME, XFCE, MATE, KDE), коли користувач заходить у свій робочий стіл.

Глобальні файли автозапуску зберігаються в каталозі: /etc/xdg/autostart/

Для реалізації цього способу необхідно:

Створіть новий .desktop файл у цій директорії- /etc/xdg/autostart/ (наприклад, для автоматичного запуску браузера або скрипта):

```
sudo nano /etc/xdg/autostart/my_global_app.desktop
```

Додайте всередину такий вміст, но у рядку Exec вкажіть повний шлях до програми або автоматизаційного скрипта:

```
[Desktop Entry]
Type=Application
Name=My Global Autostart App
Comment=Запуск програми для всіх користувачів
Exec=/usr/bin/python3 /path/to/your/script.py
Icon=utilities-terminal
Terminal=false
X-GNOME-Autostart-enabled=true
```

Збережіть файл. Тепер під час кожного входу через XRDP або локально у будь-якого користувача автоматично стартуватиме цей процес з його правами.

Для більш швидкого створення цього файлу є можливість взяти за основу існуючі файли desktop. Скористайтесь текою, де розташовані Глобальні ярлики (для всіх користувачів системи). Врахуйте наступне. Якщо програму встановлено через стандартний пакетний менеджер (apt, dpkg та інших.), її ярлик

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

створюється у теці `/usr/share/applications/`. Щоб відредагувати або додати файл, потрібні права `sudo`. Але існує додатковий шлях (для програм, встановлених вручну до каталогу `/usr/local`): `/usr/local/share/applications/`

Спосіб 2. Через глобальний скрипт запуску XRDP (`startwm.sh`) Оскільки налаштовується XRDP, є можливість вбудувати запуск необхідних програм безпосередньо в процес ініціалізації сесії. Як вказано у файлі `sesman.ini`, за замовчуванням для всіх використовується скрипт `DefaultWindowManager=startwm.sh` (який зазвичай знаходиться за шляхом `/etc/xrdp/startwm.sh`). Для цього необхідно виконати наступне

Відкрийте глобальний скрипт запуску сесії, наприклад, за допомогою редактору `nano`. Для цього скористайтесь командою :

```
sudo nano /etc/xrdp/startwm.sh
```

Знайдіть рядки, де викликається запуск самого віконного менеджера (наприклад, `test -x /etc/X11/Xsession && exec /etc/X11/Xsession` або `startxfce4`).

Перед цими фінальними рядками (які передають керування оболонці через `exec`) додайте запуск програми у фоновому режимі (обов'язково зі знаком `&` в кінці):

```
# Глобальний автозапуск для XRDP-сесій
/usr/bin/my_admin_script.sh &
# Далі йде стандартний запуск віконного менеджера, який вже є у файлі:
if [ -r /etc/X11/Xsession ]; then
    /etc/X11/Xsession
fi
```

Знак `&` критично важливий! Якщо його не поставити, скрипт `startwm.sh` зупиниться на цій програмі й сам робочий стіл у користувача ніколи не завантажиться.

Спосіб 3. Через `Systemd` (Якщо програма повинна працювати у фоні як служба, незалежно від входу користувача) Якщо програма — це сервіс автоматизації (наприклад, моніторинг живлення, керування MikroTik чи серверами vSphere), їй взагалі не потрібен графічний стіл користувача. Вона має стартувати разом із Raspberry PI. Для цього необхідно виконати наступні кроки

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

Створіть файл конфігурації служби:

```
sudo nano /etc/systemd/system/my_automation.service
```

Припишіть структуру сервісу, наприклад:

```
[Unit]
Description=My Global Automation Service
After=network.target
[Service]
Type=simple
ExecStart=/usr/bin/python3 /opt/my_project/main.py
Restart=always
User=root
[Install]
WantedBy=multi-user.target
```

Перезавантажте конфігурацію systemd, увімкніть службу в автозавантаження та запустіть її за допомогою наступних команд:

```
sudo systemctl daemon-reload
sudo systemctl enable my_automation.service
sudo systemctl start my_automation.service
```

В результаті порівняльного аналізу встановлено:

У нашому випадку, потрібен графічний додаток або скрипт, що взаємодіє з екраном користувача треба використовувати Спосіб 1 (універсальний) або Спосіб 2 (суто для RDP-сесій). Пропонується для автозавантаження використати Спосіб 1

Якщо потрібен фоновий скрипт/демон, який працює завжди й керує ресурсами Raspberry Pi – Спосіб 3.

Висновки до розділу

У даному розділі розглянуто вимоги до впровадження додаткових системних програмних додатків для реалізації функції віддалено використання мікрокомп'ютерів. До основних результатів відноситься:

1. Проведено аналіз наявних засобів віддаленого доступу для платформи Raspberry Pi (зокрема SSH, VNC, Raspberry Pi Connect, NoMachine, X2go тощо). З урахуванням умов змішаного навчання та вимог комп'ютерних

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58

лабораторій, як базове технологічне рішення було обрано кросплатформний пакет **XRDP**, що дозволяє інтегрувати мікрокомп'ютер у наявну інфраструктуру за допомогою стандартного протоколу RDP.

2. Дослідження та практичне впровадження виконано на базі сучасної архітектури операційної системи Debian GNU/Linux 13 (trixie) (версія ядра 6.18.29+rpt-rpi-v8, архітектура aarch64). У процесі підготовки платформи успішно реалізовано алгоритми базового та додаткового налаштування мережових параметрів, розгортання пакетного менеджера та системних утиліт автоматизації.

3. Важливою науково-практичною складовою розділу є успішна інтеграція Raspberry Pi до домену Microsoft AD (**на прикладі локального домену інституту ifmit.local**) за допомогою компонентів realmd, sssd, adcli та samba-common. Це дозволило забезпечити централізовану автентифікацію викладачів та здобувачів освіти за допомогою єдиного корпоративного облікового запису (через механізми PAM/sss).

4. Практично реалізовано технологію обмеження віддаленого доступу через термінальний сеанс SSH. Через конфігурування директиви AllowUsers у головному конфігураційному файлі sshd_config розроблено та перевірено сценарії створення «білих списків» доступу (обмеження за конкретними іменами користувачів, IP-адресами та масками підмереж), що суттєво підвищує стійкість лабораторного середовища до несанкціонованого доступу.

5. У ході тестування RDP-клієнтів виявлено низку критичних проблем та специфічних особливостей (зокрема блокування доступу доменних користувачів, відсутність автоматичного підключення мережових дисків, проблеми з автозавантаженням графічних додатків та обмеження прав доступу до периферії). Для усунення проблеми блокування доступу користувачів до зовнішніх пристроїв (веб-камер, мікроконтролерів Arduino тощо) було розроблено та впроваджено персоналізовані правила диспетчера пристроїв UDEV (шляхом створення файлу ddp.rules у каталозі /etc/udev/rules.d/), які

базуються на перевірці атрибутів підсистем та автоматичному наданні прав доступу.

6. Досліджено та систематизовано три основні методи керування автозавантаженням у Linux-системах (через користувацький каталог Autostart, конфігураційні файли X11/Xrdp startwm.sh та фонові служби **Systemd**). Доведено доцільність застосування Systemd-сервісів (створено конфігурацію my_automation.service) для утиліт глобальної автоматизації та моніторингу живлення лабораторних стендів, що забезпечує стабільну роботу фонових процесів незалежно від графічного сеансу користувача.

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		60

ЗАГАЛЬНІ ВИСНОВКИ

Метою даного дослідження було проаналізувати методи дистанційного використання мікрокомп'ютерів Raspberry PI 4 і розробити послідовність дій, яка дозволяє в умовах віддаленого доступу реалізувати можливості дистанційного використання у навчальному процесі. Проблема дистанційного керування комп'ютерними системами стає дедалі актуальнішою в сучасному світі, де віддалена робота і доступ до ресурсів є нормою.

Було проаналізовано існуючі апаратні аналоги Raspberry PI 4,. В процесі аналізу особлива увага приділена їх архітектурі та технічним характеристикам, що дозволило виявити сильні та слабкі сторони та визначити основні напрямки для покращення начального процесу.

Шляхом компаративного аналізу дистрибутивів обґрунтовано використання офіційної операційної системи на базі Debian (зокрема, Debian 13 Trixie) з полегшеною графічною оболонкою. Це забезпечує максимальну продуктивність і стабільність функціонування за умов обмежених апаратних ресурсів мікрокомп'ютера. Первинне конфігурування успішно реалізовано за допомогою утиліти Raspberry Pi Imager у «безголовому» (headless) режим

У ході роботи детально розглянуто користувацький інтерфейс, описано основні елементи та їх функції, що забезпечують попередні налаштування мікрокомп'ютерів. Крім того, проведено комплексне налаштування, яке включало підключення до домену Microsoft AD, налаштування системи DNS та використання зовнішніх портів, зовнішньої ВЕБ камери та попереднього тестування продуктивності.

Задля захисту лабораторної мережі адаптовано параметри SSH-сервера (sshd_config), де через механізм «білих списків» (AllowUsers) обмежено термінальний доступ сторонніх осіб. Одночасно, за допомогою конфігурування

					ІТС.4КІ.0624.03-ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Солоп Д			Загальні висновки		
Керівник		Могильний Г.А.					
Реценз.		Козуб Ю.Г.					
Н. Контр.							
Зав. каф.		Семенов М.А.					
					Лім.	Арк.	Акрушів
						3	3
					ЛНУ		
					Кафедра ІТС, Гр.4КІ		

менеджера сесій XRDP (sesman.ini) та налаштування сценарію startwm.sh, впроваджено жорсткі ліміти на тривалість сеансів (до 1 години 57 хвилин) та обмежено кількість одночасних підключень до одного стенда (не більше одного користувача).

В якості основного додаткового пакету для організації віддаленого доступу було запроваджено пакет XRDP. На основі прийнятих рішень було підтверджено, що за рахунок використання додаткових системних пакетів існує можливість налаштування всіх Raspberry PI, які використовуються у лабораторії ЛНУ імені Тараса Шевченка.

Таким чином, загальна мета роботи було досягнуто – проаналізовано попередні умови використання системних додатків з можливістю дистанційного керування мікрокомп'ютерів. Однак, аналіз використання цих додатків показав, що необхідне провести додаткові дослідження, які включають порівняльну характеристику надійності програмних додатків, ефективності використання CPU, використання оперативної пам'яті та інших ресурсів мікрокомп'ютерів .

В процесі безпосереднього впровадження та тестування встановлено, що важливою перевагою запропонованого рішення є впровадження додаткового налаштування для планування конкретного часу використання та контролю кількості одночасних підключень користувачів, контроль за використанням зовнішніх портів – USB та інших.

Крім того, в процесі безпосереднього впровадження описаних засобів віддалено підключення з'ясувалось, що рекомендуємі фірмою Raspberry PI засоби мало ефективні в умовах багатокористувацького доступу при впровадженні у навчальний процес.

Більш ґрунтовний аналіз наукової літератури та джерел Інтернет дозволив знайти інші додаткові програмні засоби:

- NoMachine
- X2go

Ці додаткові програмні засоби мають певні переваги, але потребують окремого дослідження. Тому прийнято рішення, розглянути їх можливості в межах наступного магістерського дослідження.

Результати дослідження є готовим до розгортання інженерним комплексом, який дозволяє перетворити фізичну аудиторію одноплатних комп'ютерів ЛНУ імені Тараса Шевченка на повноцінну, безпечну та автоматизовану хмару «Лабораторії як послуги» (LaaS) для дистанційного виконання практичних робіт.

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63

СПИСОК ЛІТЕРАТУРИ

1. Могильний Г.А., Аналіз програмно-апаратних засобів створення системи з віддаленим доступом до навчальних комп'ютерних лабораторій закладів середньої освіти. № 1 (277) (2023): Вісник Східноукраїнського національного університету імені Володимира Даля URL: <https://doi.org/10.33216/1998-7927-2019-256-8-5-19> (дата звернення: 10.04.2025).

2. Могильний Г.А., Семенов М.А., Кіреєв В.Ю. Впровадження системи віддаленого доступу до інформаційних ресурсів комп'ютерних лабораторій. № 2 (272) (2022): Вісник Східноукраїнського національного університету імені Володимира Даля URL: <https://doi.org/10.33216/1998-7927-2022-272-2-7-14> (дата звернення: 10.04.2025).

3. Computer Networks: A Systems Approach / Брюс С. Деві, Ларрі Л. Петерсон. Elsevier, 2021. 848 с.

4. Комп'ютерні мережі / Коробейнікова Т.І., Захарченко С.М. Львівська політехніка, 2022. 228 с.

5. Комп'ютерні мережі. Книга 1. Технології комп'ютерних мереж / Євсєєв С.П., Дженюк Н.В. Новий світ-2000, 2024. 471 с.

6. Systems Programming: Designing and Developing Distributed Applications / Річард Ентоні. Elsevier, 2015. 548 с.

7. Rahman M. A., Mansoor A. M., Alam F., Bashir A. K. Performance analysis of Raspberry Pi OS vs. Ubuntu for IoT edge computing. Proceedings of the 2021 International Conference on Cloud Computing and IoT (CCIOT). IEEE, 2021. P. 45–52

					ІТС.4КІ.0624.03-ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Солоп Д			Список літератури			
Керівник		Могильний Г.А.						
Реценз.		Козуб Ю.Г.						
Н. Контр.								
Зав. каф.		Семенов М.А.						
					Лім.	Арк.	Акрушів	
							3	3
					ЛНУ			
					Кафедра ІТС, Гр.4КІ			

8. Одноплатні ПК сячня 2026. [Електронний ресурс] - URL: <https://habr.com//companies/selectel/articles/872734/> (дата звернення: 05.05.2025).
9. Одноплатний комп'ютер Radxa Cubie A5E. [Електронний ресурс] - URL: <http://liliputing.com/> (дата звернення: 05.05.2026).
10. Одноплатний комп'ютер LattePanda Mu SoM. [Електронний ресурс] - URL <https://www.cnx-software.com/> (дата звернення: 05.05.2025).
11. Raspberry Pi hardware. [Електронний ресурс] - <https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#schematics-and-mechanical-drawings/> (дата звернення: 05.05.2026).
12. Raspbian. [Електронний ресурс] - <https://www.raspbian.org/> (дата звернення: 05.05.2026).
13. Kali Linux. [Електронний ресурс] - <https://www.kali.org/get-kali/#kali-arm> (дата звернення: 05.05.2026).
14. Ubuntu for desktops. [Електронний ресурс] - <https://ubuntu.com/download/desktop> (дата звернення: 05.05.2026).
15. RISC OS. [Електронний ресурс] - <https://www.riscosopen.org/content/downloads/raspberry-pi> (дата звернення: 05.05.2026).
16. SARPi. [Електронний ресурс] - <https://www.slackbuilds.org/repository/14.2/> (дата звернення: 05.05.2026).
17. Arch Linux. [Електронний ресурс] <https://archlinux.org/download/> (дата звернення: 05.05.2026).
18. Installing FreeBSD for Raspberry Pi. [Електронний ресурс] - <https://freebsdoundation.org/freebsd-project/resourcesold/installing-freebsd-for-raspberry-pi/> (дата звернення: 05.05.2026).
19. RetroPie. [Електронний ресурс] <https://retropie.org.uk/download/> (дата звернення: 05.05.2026).
20. Raspberry Pi OS. [Електронний ресурс] - <https://www.raspberrypi.com/software//> (дата звернення: 05.05.2026).

21. Introduction to remote access. [Електронний ресурс] - <https://www.raspberrypi.com/documentation/computers/remote-access.html#introduction-to-remote-access/> (дата звернення: 05.05.2026).

22. How to Install and Configure VNC on Ubuntu 20.04. [Електронний ресурс] - <https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-vnc-on-ubuntu-20-04> (дата звернення: 05.05.2025).

23. VNC Server for Windows. [Електронний ресурс] - https://www.realvnc.com/en/connect/download/vnc/?lai_vid=pX1W4m1GdFEp1&lai_sr=10-14&lai_sl=1 (дата звернення: 05.05.2026).

24. Raspberry Pi Connect. [Електронний ресурс] - <https://www.raspberrypi.com/documentation/services/connect.html> (дата звернення: 05.05.2026).

25. Настройка udev rules в Linux. [Електронний ресурс] - <https://losst.pro/nastrojka-udev-rules-v-linux> (дата звернення: 05.05.2026).

					<i>ІТС.4КІ.0226.03-ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		66

ДОДАТКИ

Додаток А. Файл налаштувань etc\polkit- 1\rules.d\new_allow_win_reposit.rules

```
polkit.addRule(function(action, subject) {  
    if ((action.id == "org.freedesktop.packagekit.system-sources-refresh" ||  
        action.id == "org.freedesktop.packagekit.system-network-proxy-  
            configure" ||  
        action.id.indexOf("org.freedesktop.color-manager.") == 0) &&  
        subject.active) {  
        return polkit.Result.YES;  
    }  
});
```

					ІТС.4КІ.0624.03-ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата								
Розроб.		Солоп Д			Додатки				Лім.	Арк.	Акрушіє	
Керівник		Могильний Г.А.									3	3
Реценз.		Козуб Ю.Г.							ЛНУ Кафедра ІТС, Гр.4КІ			
Н. Контр.												
Зав. каф.		Семенов М.А.										

**Додаток Б. Файл налаштувань \etc\ security\
pam_mount.conf.xml**

```
?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE pam_mount SYSTEM "pam_mount.conf.xml.dtd">
<!--
See pam_mount.conf(5) for a description.
-->

<pam_mount>

<!-- debug should come before everything else,
since this file is still processed in a single pass
from top-to-bottom -->

<debug enable="0" />

<!-- Volume definitions -->

<!-- pam_mount parameters: General tunables -->

<!--
<luserconf name=".pam_mount.conf.xml" />
-->

<!-- Note that commenting out mntoptions will give you the defaults.
      You will need to explicitly initialize it with the empty string
      to reset the defaults to nothing. -->
<mntoptions allow="nosuid,nodev,loop,encryption,fsck,nonempty,
                                allow_root,allow_other" />

<!--
<mntoptions deny="suid,dev" />
<mntoptions allow="*" />
<mntoptions deny="*" />
-->
<mntoptions require="nosuid,nodev" />

<!-- requires ofl from hxttools to be present -->

<logout wait="0" hup="no" term="no" kill="no" />

<!-- pam_mount parameters: Volume-related -->

<mkmountpoint enable="1" remove="true" />

<volume
  user="*"
  fstype="cifs"
  server="192.168.100.11"
  path="inst"
  mountpoint="/home/%(USER)/Documents/inst"
  options="vers=3.0,sec=ntlmssp,domain=IFMIT,cuid=%(USERUID),
                                file_mode=0700,dir_mode=0700"
/>
<volume
  user="*"
  user="*"
```

					ITC.4KI.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		68

```

fstype="cifs"
server="192.168.100.11"
path="buffer"
mountpoint="/home/%(USER)/Documents/buffer"
options="vers=3.0,sec=ntlmssp,domain=IFMIT,cuid=%(USERUID),
file_mode=0777,dir_mode=0777"
/>
</pam_mount>

```

					ITC.4KI.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		69

Додаток В. Файл налаштувань `\etc\ssh\sshd_config`

```
# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile             .ssh/authorized_keys
#                              .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to "no" here!
#PasswordAuthentication yes
#PermitEmptyPasswords no

# Change to "yes" to enable keyboard-interactive authentication.  Depending on
```

					ITC.4KI.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		70

```

# the system's configuration, this may involve passwords, challenge-response,
# one-time passwords or some combination of these and other methods.
# Beware issues with some PAM modules and threads.
KbdInteractiveAuthentication no

# Kerberos options
#KerberosAuthentication no          #KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes         #KerberosGetAFSToken no

# GSSAPI options
#GSSAPIAuthentication no          #GSSAPICleanupCredentials yes
#GSSAPIStrictAcceptorCheck yes    #GSSAPIKeyExchange no

# Set this to 'yes' to enable PAM authentication, account processing,
# and session processing. If this is enabled, PAM authentication will
# be allowed through the KbdInteractiveAuthentication and
# PasswordAuthentication. Depending on your PAM configuration,
# PAM authentication via KbdInteractiveAuthentication may bypass
# the setting of "PermitRootLogin prohibit-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and KbdInteractiveAuthentication to 'no'.
UsePAM yes

#AllowAgentForwarding yes          #AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10              #X11UseLocalhost yes
#PermitTTY yes
PrintMotd no
#PrintLastLog yes                 #TCPKeepAlive yes
#PermitUserEnvironment no         #Compression delayed
#ClientAliveInterval 0           #ClientAliveCountMax 3
#UseDNS no                        #PidFile /run/sshd.pid
#MaxStartups 10:30:100           #PermitTunnel no
#ChrootDirectory none            #VersionAddendum none

# no default banner path          #Banner none

# Allow client to pass locale and color environment variables
AcceptEnv LANG LC_* COLORTERM NO_COLOR

# override default of no subsystems
Subsystem sftp /usr/lib/openssh/sftp-server

# Example of overriding settings on a per-user basis
#Match User anoncvs
#
#                               X11Forwarding no
#                               AllowTcpForwarding no
#                               PermitTTY no
#                               ForceCommand cvs server
allowUsers ubuntu5 admin*

```

Додаток Д. Файл налаштувань \etc\sssd\sssd.conf

```
[sssd]
domains = ifmit.local
config_file_version = 2
services = nss, pam

[domain/ifmit.local]
default_shell = /bin/bash
krb5_store_password_if_offline = True
cache_credentials = True
krb5_realm = IFMIT.LOCAL
realmd_tags = manages-system joined-with-adcli
id_provider = ad
fallback_homedir = /home/%u@%d
ad_domain = ifmit.local
use_fully_qualified_names = True
ldap_id_mapping = True
access_provider = ad
ad_gpo_map_remote_interactive = +xrdp-sesman
```

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		72

Додаток Е. Файл налаштувань `\etc\ systemd\ logind.conf`

```
# This file is part of systemd.
#
# systemd is free software; you can redistribute it and/or modify it under the
# terms of the GNU Lesser General Public License as published by the Free
# Software Foundation; either version 2.1 of the License, or (at your option)
# any later version.
#
# Entries in this file show the compile time defaults. Local configuration
# should be created by either modifying this file (or a copy of it placed in
# /etc/ if the original file is shipped in /usr/), or by creating "drop-ins" in
# the /etc/systemd/logind.conf.d/ directory. The latter is generally
# recommended. Defaults can be restored by simply deleting the main
# configuration file and all drop-ins located in /etc/.
#
# Use 'systemd-analyze cat-config systemd/logind.conf' to display the full
# config.
#
# See logind.conf(5) for details.

[Login]
#NAutoVTs=6
#ReserveVT=6
##### no
KillUserProcesses=yes
#KillOnlyUsers=
#KillExcludeUsers=root
#InhibitDelayMaxSec=5
#UserStopDelaySec=10
#SleepOperation=suspend-then-hibernate suspend
#HandlePowerKey=poweroff
#HandlePowerKeyLongPress=ignore
#HandleRebootKey=reboot
#HandleRebootKeyLongPress=poweroff
#HandleSuspendKey=suspend
#HandleSuspendKeyLongPress=hibernate
#HandleHibernateKey=hibernate
#HandleHibernateKeyLongPress=ignore
#HandleLidSwitch=suspend
#HandleLidSwitchExternalPower=suspend
#HandleLidSwitchDocked=ignore
#HandleSecureAttentionKey=secure-attention-key
#PowerKeyIgnoreInhibited=no
#SuspendKeyIgnoreInhibited=no
#HibernateKeyIgnoreInhibited=no
#LidSwitchIgnoreInhibited=yes
#RebootKeyIgnoreInhibited=no
#HoldoffTimeoutSec=30s
#IdleAction=ignore
#IdleActionSec=30min
#RuntimeDirectorySize=10%
#RuntimeDirectoryInodesMax=
#RemoveIPC=yes
#InhibitorsMax=8192
#SessionsMax=8192
#StopIdleSessionSec=infinity
#DesignatedMaintenanceTime=
```

					ITC.4KI.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		73

Додаток Ж. Файл налаштувань

\etc\xdg\autostart\Cheese.desktop

```
[Desktop Entry]
Name[ab]=Cheese
Name[ro]=Cheese
Name[ru]=Cheese
Name[uk]=Сир
Name=Cheese
GenericName[ab]=Афото җышыә
GenericName[ru]=Фотовидеобудка
GenericName[uk]=Кабінка
GenericName=Webcam Booth
Comment[ku]=Bi webcamera xwe wêne û vîdyoyan bikişînin, bi efektên seyr.
Comment[ru]=Создание фотографий и видео с помощью веб-камеры, с применением
занятных эффектов
Comment[uk]=Створення фотографій та відео за допомогою веб-камери, із
використанням цікавих ефектів
Comment=Take photos and videos with your webcam, with fun graphical effects
# Translators: Search terms to find this application. Do NOT translate or
localize the semicolons! The list MUST
also end with a semicolon!
Keywords[ab]=афото;авидео;ваеб-камера;акамера; селфи;
Keywords=photo;video;webcam;camera;selfie;
Exec=cheese
Terminal=false
Type=Application
StartupNotify=true
# Translators: Do NOT translate or transliterate this text (this is an icon
file name)!
Icon=org.gnome.Cheese
Categories=GNOME;AudioVideo;Video;Recorder;
DBusActivatable=true
```

					ІТС.4КІ.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		74

Додаток 3. Файл налаштувань `\etc\xrdp\sesman.ini`

;; See ``man 5 sesman.ini`` for details

```
[Globals]
; listening port
#ListenPort=sesman.socket
EnableUserWindowManager=true
; Give in relative path to user's home directory
UserWindowManager=startwm.sh
; Give in full path or relative path to /etc/xrdp
DefaultWindowManager=startwm.sh
; Give in full path or relative path to /etc/xrdp
ReconnectScript=reconnectwm.sh

[Security]
AllowRootLogin=true
MaxLoginRetry=4
TerminalServerUsers=tsusers
TerminalServerAdmins=tsadmins
; When AlwaysGroupCheck=false access will be permitted
; if the group TerminalServerUsers is not defined.
AlwaysGroupCheck=false
; When RestrictOutboundClipboard=all clipboard from the
; server is not pushed to the client.
; In addition, you can control text/file/image transfer restrictions
; respectively. It also accepts comma separated list such as text,file,image.
; To keep compatibility, some aliases are also available:
;   true: an alias of all
;   false: an alias of none
;   yes: an alias of all
RestrictOutboundClipboard=none
; When RestrictInboundClipboard=all clipboard from the
; client is not pushed to the server.
; In addition, you can control text/file/image transfer restrictions
; respectively. It also accepts comma separated list such as text,file,image.
; To keep compatibility, some aliases are also available:
;   true: an alias of all
;   false: an alias of none
;   yes: an alias of all
RestrictInboundClipboard=none
; Set to 'no' to prevent users from logging in with alternate shells
#AllowAlternateShell=true
; On Linux systems, the Xorg X11 server is normally invoked using
; no_new_privs to avoid problems if the executable is suid. This may,
; however, interfere with the use of security modules such as AppArmor.
; Leave this unset unless you need to disable it.
#XorgNoNewPrivileges=true
; Specify the group which is to have read access to the directory where
; local sockets for the session are created. This is normally the GID
; which the xrdp process runs as.
; Default is 'root'
#SessionSockdirGroup=root

[Sessions]
;; X11DisplayOffset - x11 display number offset
; Type: integer
```

					ITC.4KI.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		75

```

; Default: 10
X11DisplayOffset=10

;; MaxSessions - maximum number of connections to an xrdp server
; Type: integer
; Default: 0
; 50
MaxSessions=1

;; MaxDisplayNumber - maximum number considered for an X display
; Type: integer
; Default: 63
;
; IANA only allocates TCP ports up to 6063 for X servers. If you are not
; allowing TCP connections to your X servers you may safely increase this
; number.
#MaxDisplayNumber=63

;; KillDisconnected - kill disconnected sessions
; Type: boolean
; Default: false
; if 1, true, or yes, every session will be killed within DisconnectedTimeLimit
; seconds after the user disconnects
; false
KillDisconnected=true

;; DisconnectedTimeLimit (seconds) - wait before kill disconnected sessions
; Type: integer
; Default: 0
; if KillDisconnected is set to false, this value is ignored
; 0
DisconnectedTimeLimit=120

;; IdleTimeLimit (seconds) - wait before disconnect idle sessions
; Type: integer
; Default: 0
; Set to 0 to disable idle disconnection.
; 0
IdleTimeLimit=1800

;; Policy - session allocation policy
;
; Type: enum [ "Default" | "Separate" | Combination from {UBDI} ]
; "Default"      Currently same as "UB"
; "Separate"     All sessions are separate. Sessions can never be rejoined,
;                and will need to be cleaned up manually, or automatically
;                by setting other sesman options.
;
; Combination options:-
;   U      Sessions are separated per user
;   B      Sessions are separated by bits-per-pixel
;   D      Sessions are separated by initial display size
;   I      Sessions are separated by IP address
;
; The options U and B are always active, and cannot be de-selected.
Policy=Default

[Logging]
; Note: Log levels can be any of: core, error, warning, info, debug, or trace
LogFile=xrdp-sesman.log
LogLevel=INFO
EnableSyslog=true
#SyslogLevel=INFO
#EnableConsole=false

```

					ITC.4KI.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		76

```

#ConsoleLevel=INFO
#EnableProcessId=false

[LoggingPerLogger]
; Note: per logger configuration is only used if xrdp is built with
; --enable-devel-logging
#sesman.c=INFO
#main()=INFO

;
; Session definitions - startup command-line parameters for each session type
;

[Xorg]
; Specify the path of non-suid Xorg executable. It might differ depending
; on your distribution and version. Find out the appropriate path for your
; environment. The typical path is known as follows:
;
; Fedora 26 or later      : param=/usr/libexec/Xorg
; Debian 9 or later      : param=/usr/lib/xorg/Xorg
; Ubuntu 16.04 or later  : param=/usr/lib/xorg/Xorg
; Arch Linux             : param=/usr/lib/Xorg
; CentOS 7               : param=/usr/bin/Xorg or param=Xorg
; CentOS 8               : param=/usr/libexec/Xorg
; FreeBSD (from 2022Q4)  : param=/usr/local/libexec/Xorg
;
param=/usr/lib/xorg/Xorg
; Leave the rest parameters as-is unless you understand what will happen.
param=-config
param=xrdp/xorg.conf
param=-noreset
param=-nolisten
param=tcp
param=-logfile
param=.xorgxrdp.%s.log

[Xvnc]
param=Xvnc
param=-bs
param=-nolisten
param=tcp
param=-localhost
param=-dpi
param=96

[Chansrv]
; drive redirection
; See sesman.ini(5) for the format of this parameter
#FuseMountName=/run/user/%u/thinclient_drives
#FuseMountName=/media/thinclient_drives/%U/thinclient_drives
FuseMountName=thinclient_drives
; this value allows only the user to access their own mapped drives.
; Make this more permissive (e.g. 022) if required.
FileUmask=077
; Can be used to disable FUSE functionality - see sesman.ini(5)
#EnableFuseMount=false
; Uncomment this line only if you are using GNOME 3 versions 3.29.92
; and up, and you wish to cut-paste files between Nautilus and Windows. Do
; not use this setting for GNOME 4, or other file managers
#UseNautilus3FlistFormat=true
; sound redirection
; workaround for Microsoft mstsc.exe to suppress noise.
; SoundNumSilentFramesAAC | SoundNumSilentFramesMP3 silent frames are sent
; before SNDC_CLOSE is sent.

```

					ITC.4KI.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		77

```

; during SoundMsecDoNotSend mS after SNDC_CLOSE is sent, sound data is not
; send.
; depending on the environment, it might be necessary to increase values.
; Defaults: SoundNumSilentFramesAAC=4, SoundNumSilentFramesMP3=2,
;           SoundMsecDoNotSend=1000
; If set to 0, this workaround is not applied.
#SoundNumSilentFramesAAC=4
#SoundNumSilentFramesMP3=2
#SoundMsecDoNotSend=1000

[ChansrvLogging]
; Note: one log file is created per display and the LogFile config value
; is ignored. The channel server log file names follow the naming convention:
; xrdp-chansrv.${DISPLAY}.log
;
; Note: Log levels can be any of: core, error, warning, info, debug, or trace
LogLevel=INFO
EnableSyslog=true
#SyslogLevel=INFO
#EnableConsole=false
#ConsoleLevel=INFO
#EnableProcessId=false

[ChansrvLoggingPerLogger]
; Note: per logger configuration is only used if xrdp is built with
; --enable-devel-logging
#chansrv.c=INFO
#main()=INFO

[SessionVariables]
PULSE_SCRIPT=/etc/xrdp/pulse/default.pa

```

Додаток К. Файл налаштувань \etc\xrdp\startwm.sh

```
#!/bin/sh
# xrdp X session start script (c) 2015, 2017, 2021 mirabilos
# published under The MirOS Licence

# Rely on /etc/pam.d/xrdp-sesman using pam_env to load both
# /etc/environment and /etc/default/locale to initialise the
# locale and the user environment properly.

if test -r /etc/profile; then
    . /etc/profile
fi

if test -r ~/.profile; then
    . ~/.profile
fi
#
#test -x /etc/X11/Xsession && exec /etc/X11/Xsession
#exec /bin/sh /etc/X11/Xsession
if [ -x /etc/X11/Xsession ]; then
    timeout 117m /etc/X11/Xsession
else
    timeout 117m /bin/sh /etc/X11/Xsession
fi
sudo /bin/systemctl reboot
```

					ITC.4KI.0226.03-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		79