КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ

Кваліфікаційна робота бакалавра

Захищена система Інтернету речей на міні-комп'ютері Raspberry Pi

Серверна частина

Виконав: Задорожний Іван Михайлович, КСМ-22

Керівник: к.т.н., доцент Вишняков В. М.

Київ - 2026

АКТУАЛЬНІСТЬ

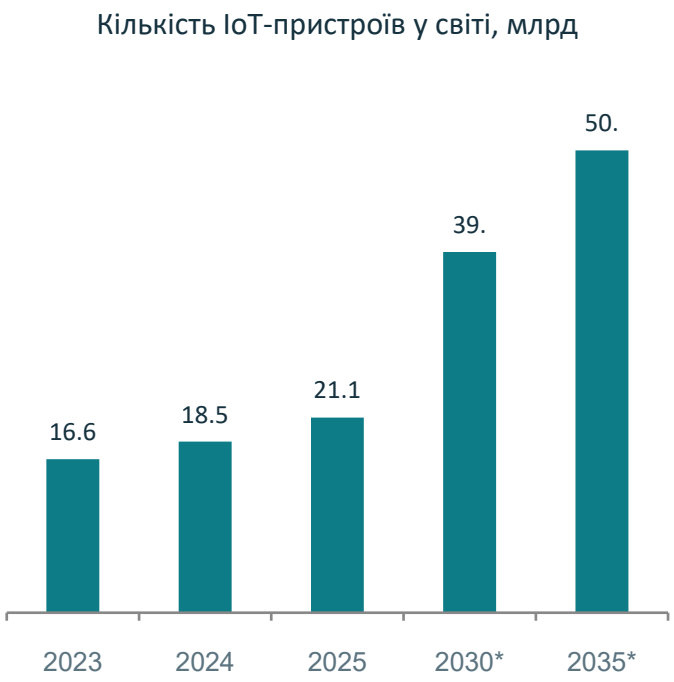
Інтернет речей зростає - разом із загрозами

Дедалі більше фізичних об'єктів керуються дистанційно через відкриті канали мережі Інтернет. Масштаб технології робить її захист критично важливим завданням.

18,5 млрд пристроїв у світі (2024)

~39 млрд прогноз на 2030 рік

90 % користувачів не впевнені у захисті*



** за опитуванням Gemalto*

ПРОБЛЕМА

Чому пристрої IoT уразливі

Відкриті канали

Пряме підключення вузла до Інтернету відкриває його для несанкціонованого доступу ззовні.

Слабкий захист

Заради низької вартості пристрої мають слабкі паролі, відкриті порти й не оновлюються.

Загроза іншим

Скомпрометований пристрій стає знаряддям розподілених атак на відмову в обслуговуванні (DDoS).

МЕТА ДОСЛІДЖЕННЯ

Мета, об'єкт і предмет роботи

Мета

Підвищити інформаційну безпеку систем дистанційного керування об'єктами на базі Raspberry Pi через розроблення серверної частини, побудованої за схемою сервера-посередника.

Об'єкт

Процеси віддаленого керування об'єктами в системах Інтернету речей та забезпечення їх інформаційної безпеки.

Предмет

Серверна частина - програмні засоби сервера-посередника для безпечного обміну між користувачем і Raspberry Pi.

ЗАВДАННЯ РОБОТИ

Шість задач для досягнення мети

1. Проаналізувати стан IoT, типові архітектури та притаманні їм загрози безпеки.
2. Обґрунтувати схему сервера-посередника для захисту від доступу та DDoS-атак.
3. Обґрунтувати вибір апаратної платформи, ОС і програмних засобів сервера.
4. Розробити серверне ПЗ для обміну за TCP та взаємодії з користувачем за HTTP.
5. Реалізувати засоби захищеного обміну даними між складовими системи.
6. Розгорнути, налаштувати, протестувати серверну частину й оцінити надійність.

ОБРАНЕ РІШЕННЯ

Схема із сервером-посередником

Пряме підключення

- Вузол отримує реальну адресу в мережі
- Доступний для атак ззовні
- Захист розосереджено по пристроях
- Один користувач - одна система

Сервер-посередник

✓ Керований вузол не має реальної IP-адреси

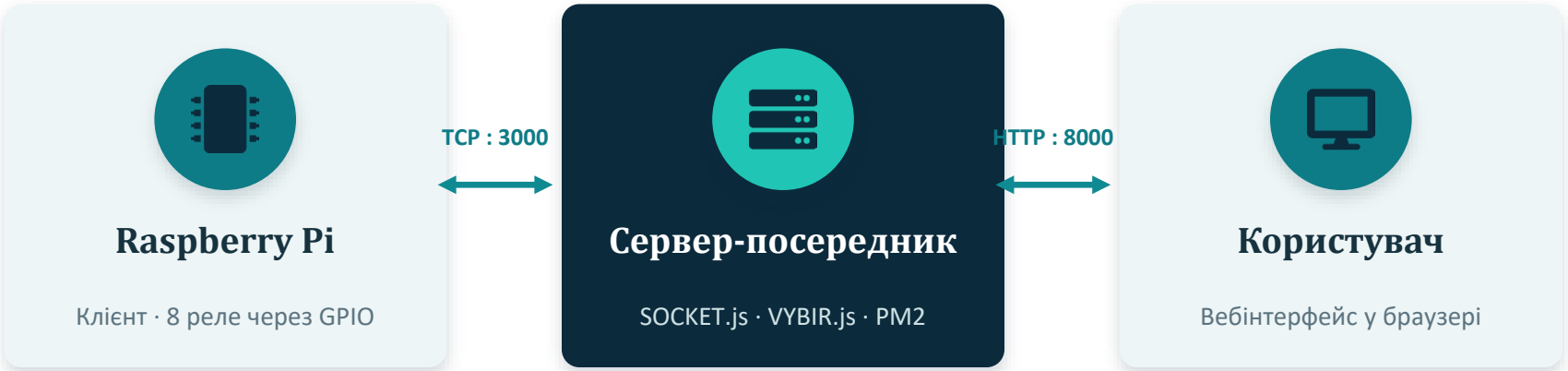
✓ Прихований від прямих атак з Інтернету

✓ Захист зосереджено в одному вузлі

✓ Один сервер - багато клієнтів (SaaS)

АРХІТЕКТУРА СИСТЕМИ

Як влаштована система керування



Серверна частина приймає стан об'єктів від мікрокомп'ютера, передає йому команди керування й надає користувачеві вебінтерфейс для спостереження та управління.

ОБҐРУНТУВАННЯ ВИБОРУ

Програмно-апаратні засоби сервера

OpenBSD

Операційна система з орієнтацією на безпеку та мінімальною поверхнею атак.

Node.js

Асинхронна модель вводу-виводу й єдина з клієнтом мова програмування.

PM2

Менеджер процесів із автоматичним перезапуском сервісів для безперервної роботи.

СЕРВЕРНЕ ПЗ

Два незалежні сервіси

SOCKET.js

TCP : 3000

- Обмін з мікрокомп'ютером за TCP
- Перевірка коректності даних
- Зберігання стану об'єктів

VYBIR.js

HTTP : 8000

- Вебінтерфейс користувача
- Повертає стан, приймає команди
- Опрацювання міждоменних запитів

Внутрішній обмін між сервісами - через спільні файли: незалежність сервісів і збереження стану між сеансами.

ЗАХИЩЕНИЙ ОБМІН

Криптографічний захист каналу

Шифр Вернама

Одноразовий блокнот - абсолютна, математично доведена стійкість шифрування.

Діффі-Геллман

Узгодження спільного ключа над полем Галуа - без передавання ключа каналом.

Чому це працює на практиці

- Криптоперетворення виконує однаковий код на сервері та мікрокомп'ютері - узгодженість обчислень.
- Керування потребує малого обсягу даних, тож абсолютно стійке шифрування не знижує швидкодії.

ТЕСТУВАННЯ

Перевірено за основними сценаріями

Доставлення команд

Команди користувача коректно досягали мікрокомп'ютера.

Синхронізація стану

Стан об'єктів узгоджувався між кількома користувачами.

Відхилення помилок

Некоректні дані відхилялися сервісом приймання.

Автовідновлення

Збій сервісу усувався перезапуском зі збереженням стану.

Усі тестові випадки пройдено успішно.

РЕЗУЛЬТАТ ЗАХИСТУ

Комплексний захист на чотирьох рівнях

Архітектурний

Приховування керованого вузла за сервером-посередником.

Криптографічний

Шифрування каналу та узгодження ключів без їх передавання.

Системний

Захищена ОС OpenBSD та обмеження доступу брандмауером.

Експлуатаційний

Автоматичне відновлення роботи сервісів через PM2.

РЕЗУЛЬТАТИ

Що зроблено в роботі

- Розроблено серверну частину захищеної системи IoT за схемою сервера-посередника.
- Реалізовано два сервіси: SOCKET.js (TCP) та VYBIR.js (HTTP) зі збереженням стану.
- Впроваджено захищений обмін на основі шифру Вернама та алгоритму Діффі-Геллмана.
- Серверну частину розгорнуто на OpenBSD, налаштовано й успішно протестовано.

Готово до моделі SaaS

Один сервер обслуговує багато незалежних клієнтських систем.

ПЕРСПЕКТИВИ

Напрями подальшого розвитку

Взаємна автентифікація

Запровадити перевірку автентичності обох сторін обміну даними.

Більше об'єктів

Розширити кількість керованих об'єктів понад вісім реле.

Журналювання

Додати журнал дій користувачів для аудиту та контролю.

Усі напрями не потребують зміни обраної архітектури й можуть впроваджуватися поступово.

ВИСНОВКИ

Мету роботи досягнуто

Розв'язано актуальну задачу підвищення інформаційної безпеки систем дистанційного керування на базі Raspberry Pi. Розроблена серверна частина підтвердила працездатність і відповідність висунутим вимогам.

- Усі шість завдань роботи виконано повністю.
- Захист забезпечено на чотирьох рівнях одночасно.
- Рішення придатне для надання послуги за моделлю SaaS.

Дякую за увагу!

Готовий відповісти на ваші запитання

Задорожний Іван Михайлович · КСМ-22

Керівник: к.т.н., доцент Вишняков В. М. · Київ, 2026