

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕКОНОМІКИ І МЕНЕДЖМЕНТУ**

Кафедра кібернетики та інформатики

КВАЛІФІКАЦІЙНА РОБОТА

освітній ступінь - «Бакалавр»

**на тему: Інформаційна технологія забезпечення
централізованого контролю безпеки використання USB-
накопичувачів в корпоративній мережі.**

Виконав:

студент спеціальності

126 «Інформаційні системи та технології»

ОП «Інформаційні системи та технології»

Шакотько Єгор Сергійович

Керівник:

В'юненко Олександр Борисович

к. е. н., доцент кафедри кібернетики та інформатики

Рецензент:

Ободяк Віктор Корнелійович

к. т. н., доцент кафедри кібербезпеки

Сумського державного університету

Суми – 2023

СУМСЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ

Факультет	Економіки і менеджменту
Кафедра	Кібернетики та інформатики
Освітній ступінь	«Бакалавр»
Спеціальність	126 «Інформаційні системи та технології» ОП «Інформаційні системи та технології»

Затверджую:
Завідувач кафедри _____ Світлана АГАДЖАНОВА

«_____» «_____» 2023 ____ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Шакотько Єгору Сергійовичу

1. Тема роботи: Інформаційна технологія забезпечення централізованого контролю безпеки використання USB-накопичувачів в корпоративній мережі.

2. База практичного дослідження: Сумський НАУ.

керівник роботи В'юненко Олександр Борисович, к.т.н., доцент

затверджена наказом по університету від _____ «_____» _____ 2023 р. № _____ -н

3. Строк подання студентом закінченого проекту (роботи) _____ «23» червня 2023 року

4. Вихідні дані до проекту (роботи):

Результати аналітичних досліджень зарубіжних та вітчизняних вчених, нормативні документи та чинне законодавство, Інтернет-джерела.

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

- виконати постановку задачі;
- вибрати середовище для реалізації програмного додатку;
- провести аналіз наявних рішень для вирішення проблеми;
- розробити проєкт удосконалення наявної мережі;
- розробити віртуальну модель модернізованої корпоративної мережі;
- розробити програмний продукт для перевірки підключень USB-пристроїв до абонентів корпоративної мережі.

6. Дата видачі завдання: _____ «_____» _____ 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту(роботи)	Строк виконання етапів проекту	Примітка
1	Визначення об'єкту, предмету дослідження, формулювання мети та задач кваліфікаційної роботи, складання плану	січень 2023 р.	виконано
2	Підбір та вивчення літературних джерел	січень 2023 р.	виконано
3	Узагальнення теоретичного матеріалу з обраної теми дослідження та представлення чорнового варіанту першого розділу кваліфікаційної роботи	січень 2023 р.	виконано
4	Збір та обробка фактичного матеріалу, узагальнення аналізу застосування досліджуваного питання в установі (переддипломна практика)	січень - березень 2023 р.	виконано
5	Оформлення теоретичної частини кваліфікаційної роботи, узагальнення аналітичної частини, захист звіту по переддипломній практиці	березень 2023 р.	виконано
6	Оформлення практичної частини кваліфікаційної роботи, розробка проекту модернізованої мережі, розробка програмного продукту для перевірки підключень USB-пристроїв до абонентів корпоративної мережі.	квітень 2023 р.	виконано
7	Підготовка висновків, пропозицій та узгодження з керівником	травень 2022 р.	виконано
8	Підготовка кінцевого варіанту кваліфікаційної роботи	до 3 червня 2023 р.	виконано
9	Опрацювання зауважень керівника, перевірка на автентичність	до 6 червня 2023 р.	виконано
10	Перевірка з академічної доброчесності кваліфікаційної роботи	9 червня 2023 р..	виконано
11	Представлення кваліфікаційної роботи на кафедрі, деканат	23 червня 2023 р..	виконано
12	Підготовка доповіді, презентації до кваліфікаційної роботи та її попередній захист	до 27 червня 2023 р..	виконано
13	Захист кваліфікаційної роботи	28 червня 2023 р.	виконано

Здобувач вищої освіти

Єгор ШАКОТЬКО

Керівник роботи

Олександр
В'ЮНЕНКО

Перевірку на автентичність проведено

Надія БАРАНИК

Дипломна робота допущена до захисту

Маргарита ЛИШЕНКО

АНОТАЦІЯ

Шакотько Є.С. Інформаційна технологія забезпечення централізованого контролю безпеки використання USB-накопичувачів в корпоративній мережі.

Кваліфікаційна робота зі спеціальності 126 «Інформаційні системи та технології» ОП Інформаційні системи та технології, СНАУ, Суми-2023р. – Рукопис.

Кваліфікаційна робота присвячена дослідженню актуальних питань, пов'язаних з організацією безпеки використання USB-накопичувачів у корпоративних мережах. У роботі проведений аналіз проблем пов'язаних з безпекою використання USB-накопичувачів. Наведене обґрунтування перспективних шляхів удосконалення системи контролю безпеки, зокрема використання інформаційних технологій.

Актуальність теми обумовлена зростанням загроз безпеці інформації в корпоративних мережах, пов'язаною з можливістю несанкціонованого доступу до даних через USB-накопичувачі. Проведений аналіз європейського досвіду в цій галузі та надані рекомендації щодо впровадження ефективної інформаційної технології для централізованого контролю безпеки.

Основними завданнями дослідження є вивчення та аналіз існуючих методів і засобів контролю безпеки використання USB-накопичувачів, розробка та реалізація інформаційної технології для централізованого контролю, а також оцінка ефективності впровадження запропонованої технології в мережу організації.

Результати дослідження сприятимуть покращенню безпеки використання USB-накопичувачів у корпоративних мережах, забезпеченню захисту конфіденційної інформації та запобіганню можливим загрозам безпеки. Розроблена інформаційна технологія може бути використана організаціями для підвищення рівня безпеки і забезпечення контролю над використанням USB-накопичувачів у корпоративних мережах.

Ключові слова: USB-накопичувач, організація безпеки у корпоративних мережах.

SUMMARY

Shakotko Y.S. Information Technology for Centralized Security Control of USB Device Usage in Corporate Networks.

Qualification work in the specialty 126 "Information Systems and Technologies," Study Program: Information Systems and Technologies, Sumy National Agrarian University, 2023 - Manuscript.

This qualification work focuses on the investigation of relevant issues related to the security of USB device usage in corporate networks. The study analyzes the problems associated with the security of USB devices and provides justification for prospective approaches to enhance security control systems, particularly through the utilization of information technology.

The significance of the topic is driven by the increasing threats to information security in corporate networks, primarily due to the potential unauthorized access to data through USB devices. The analysis of European experience in this field is conducted, and recommendations for implementing effective information technology for centralized security control are provided.

The main objectives of the research include studying and analyzing existing methods and tools for security control of USB device usage, developing and implementing an information technology solution for centralized control, and evaluating the effectiveness of implementing the proposed technology in the organization's network.

The research outcomes will contribute to improving the security of USB device usage in corporate networks, safeguarding confidential information, and mitigating potential security threats. The developed information technology can be employed by organizations to enhance security levels and exert control over USB device usage in corporate networks.

Keywords: USB device, security organization in corporate networks.

Зміст

ВСТУП	6
1.1 Огляд останніх досліджень і публікацій	8
1.2 Методи побудови програмного забезпечення та даних про стан корпоративної мережі	20
1.3 Аналіз наявних рішень для вирішення проблеми	23
РОЗДІЛ 2. ПРОЄКТУВАННЯ ЛОКАЛЬНОЇ МЕРЕЖІ ТА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ.....	28
2.1 Постановка задачі та визначення мети	28
2.2 Вибір методів реалізації та створення проєкту.....	32
2.3 Розробка віртуальної моделі локальної мережі.	42
РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ	50
3.1 Програмна реалізація клієнтської служби.....	50
3.2 Програмна реалізація інтерфейсу користувача	56
3.3 Використання програмного додатку	61
ВИСНОВКИ.....	65
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	66
ДОДАТКИ.....	68
Додаток А.....	69
Додаток Б	74
Додаток В.....	76
Додаток Г	80

ВСТУП

Співвідношення між доступністю, розміром та ємністю USB-накопичувачів ставить під загрозу інформаційну безпеку компаній. Недбале ставлення до цієї проблеми може призвести до значних фінансових втрат або навіть банкрутства. Зловмисники можуть використовувати підключення до USB-порту для крадіжки важливої інформації або встановлення шкідливого програмного забезпечення. З цією загрозою все більше компаній збільшують свої інвестиції в мережеві екрани, використовують надійні алгоритми шифрування, а також інші засоби та технології контролю, щоб захистити свої дані від крадіжок через Інтернет. Однак найчастіше зловмисниками стають самі працівники компанії, які крадуть інформацію про інвестиції, проектні документи, комерційні пропозиції, особисті дані та контакти клієнтів і багато іншого, з метою збагачення або помсти. USB-пристрої стають ідеальними знаряддями для втілення цих кримінальних намірів.

Відключення USB-портів є радикальним заходом безпеки, який може призвести до серйозних незручностей і навіть унеможливити підключення периферійних пристроїв до комп'ютера. Встановлення списку дозволених пристроїв за допомогою серійних номерів є добрим кроком для запобігання витоку інформації, але це не є достатнім. Серійні номери флеш-накопичувачів можна легко змінити на будь-який бажаний, що дозволяє зловмиснику отримати доступ до важливих даних без перешкод.

Важливим заходом безпеки є моніторинг підключень USB-пристроїв до корпоративних комп'ютерів. Централізоване відстеження підключень є ефективним способом розслідування витоку інформації через USB-канали, що дозволяє керівництву відділу безпеки або адміністратору безпеки реагувати своєчасно і встановити пристрій, дату, час і навіть комп'ютер, де сталася інцидент. Аналіз цих даних і виявлення факту підключення пристрою допомагають успішно встановити особу зловмисника і вжити необхідні заходи.

Об'єктом дослідження є локальна мережа на базі підприємства: Відокремлений структурний підрозділ «Глухівський агротехнічний фаховий коледж Сумського НАУ».

Метою цієї роботи є розробка системи моніторингу підключень USB-пристроїв для невеликих організацій, які не мають можливості придбати дорогі системи з непотрібними для цієї задачі функціями.

Для вирішення завдання були поставлені наступні задачі:

- Розробити проєкт для модернізації наявної мережі та створення програмного продукту.
- Розробити віртуальну модель модернізованої корпоративної мережі.
- Розробити програмний продукт для перевірки підключень USB-пристроїв до абонентів корпоративної мережі.

Актуальність даної теми зумовлена:

- USB-накопичувачі можуть бути використані для несанкціонованого викрадення, втрати або розголошення конфіденційних даних. Контроль за використанням USB-накопичувачів дозволяє організації попередити витіки даних, зменшити ризик крадіжок та зберегти цінну інформацію.

- USB-накопичувачі можуть бути заражені вірусами та шкідливими програмами, які можуть поширюватися в корпоративній мережі. Централізований контроль дозволяє сканувати USB-накопичувачі на наявність шкідливого коду перед їх використанням у системі, що допомагає запобігти поширенню загроз усередині мережі.

- USB-накопичувачі можуть бути використані для викрадення важливих даних з корпоративної мережі. За допомогою контролю за використанням USB-накопичувачів можна обмежити доступ до цієї функції для певних користувачів або встановити механізми шифрування для даних, що зберігаються на накопичувачах.

- У багатьох галузях, таких як фінансові установи, охорона здоров'я та галузі, пов'язані з особистою інформацією, існують обов'язкові норми щодо забезпечення безпеки даних.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

Забезпечення захисту локальних мереж. У сучасному цифровому світі, де технології швидко розвиваються, захист локальних мереж стає все більш актуальною проблемою. Локальна мережа - це мережа комп'ютерів та інших пристроїв, які пов'язані між собою у межах певної території, наприклад, офісу або підприємства. Забезпечення безпеки цих мереж має вирішальне значення для збереження конфіденційності, цілісності та доступності інформації.

Усвідомлення загроз. Першим кроком до ефективного захисту локальної мережі є усвідомлення потенційних загроз. Небезпека може прийти ззовні або зсередини організації. Хакери, зловмисники, віруси та зловживання дозволами співробітників - це лише деякі з чинників, які можуть поставити під загрозу безпеку мережі.

Використання захисних програмних засобів. Захист локальних мереж є однією з найважливіших проблем у сучасному світі, оскільки комп'ютерні системи стикаються з різноманітними загрозами та ризиками. Програмні засоби захисту локальних мереж відіграють ключову роль у забезпеченні безпеки, цілісності та конфіденційності інформації, яка передається через мережу.

Захист локальних мереж став надзвичайно важливою задачею у цифрову епоху, де комп'ютерні системи постійно піддаються загрозам та кібератакам. Програмні засоби захисту локальних мереж грають ключову роль у забезпеченні безпеки, цілісності та конфіденційності інформації, що передається через мережу. У цьому розділі ми розглянемо важливість захисту локальних мереж та загальні загрози, з якими вони стикаються.

Захист локальних мереж має велике значення для підприємств та організацій, які залежать від надійної та безпечної передачі даних. Локальна мережа є основним

каналом комунікації та обміну інформацією всередині організації, і недостатня безпека може призвести до різних проблем, включаючи втрату даних, порушення конфіденційності, збитки від кібератак та втрату репутації.

Локальні мережі піддаються різноманітним загрозам, включаючи:

а) віруси та шкідливі програми: Віруси, черв'яки, троянські програми та інші шкідливі програми можуть інфікувати комп'ютери в мережі, поширюючись через незахищені точки входу та завдаючи шкоди системам;

б) вторгнення та хакерські атаки: Хакери можуть намагатися проникнути в локальну мережу, використовуючи різні методи, такі як перехоплення даних, злам паролів або використання слабкостей в програмному забезпеченні;

в) соціальний інжиніринг: Атаки, пов'язані з соціальним інжинірингом, спираються на маніпулювання користувачами для отримання несанкціонованого доступу до мережі або конфіденційної інформації;

г) фішинг це вид шахрайства, де зловмисники намагаються здобути конфіденційну інформацію, шляхом використання підроблених веб-сайтів або електронних повідомлень, які здаються легітимними.

Програмні засоби захисту локальних мереж грають критичну роль у запобіганні та виявленні цих загроз. Вони забезпечують контроль доступу до мережі, моніторинг активності, виявлення та блокування шкідливих програм, а також шифрування комунікації для забезпечення конфіденційності.

Файрволи є одним з найважливіших програмних засобів захисту локальних мереж. Вони відповідають за контроль руху даних, блокування небезпечних з'єднань та шкідливого трафіку. Основні функції файрволів:

а) файрволи встановлюють правила, які регулюють доступ до локальної мережі. Вони можуть блокувати або дозволяти певні типи з'єднань на основі IP-адрес, портів або протоколів. Це дозволяє контролювати, хто має доступ до мережі та які ресурси вони можуть використовувати;

б) файрволи використовують сигнатури та евристичний аналіз для виявлення шкідливих програм, таких як віруси, черв'яки або троянські програми. Вони

можуть блокувати спроби входу таких програм в мережу або виявляти ізольовані комп'ютери, які можуть бути заражені;

в) файрволи можуть фільтрувати трафік, що проходить через мережу, для виявлення та блокування небезпечних або небажаних з'єднань. Вони можуть перевіряти пакети даних на відповідність встановленим правилам та політикам безпеки;

г) файрволи здатні реєструвати інформацію про вхідний та вихідний трафік, зберігати журнали активності та аудитувати події, пов'язані з мережею. Це допомагає виявити аномальну активність, вразливості та потенційні загрози.

Антивірусне програмне забезпечення є необхідною складовою для захисту локальних мереж від вірусів, шкідливих програм та інших кіберзагроз.

Антивірусне програмне забезпечення використовує сигнатури та алгоритми сканування для виявлення вірусів та шкідливих програм. Воно аналізує файли, електронні повідомлення та додатки, що надходять на комп'ютери в мережі, та блокує або очищає заражені файли.

Ефективне антивірусне програмне забезпечення повинне регулярно оновлювати свої вірусні бази даних, щоб розпізнавати нові шкідливі програми та віруси. Це дозволяє постійно покращувати рівень захисту та ефективно реагувати на нові загрози.

Антивірусне програмне забезпечення працює в реальному часі, постійно моніторячи активність системи та новий трафік. Воно блокує або ізолює потенційно небезпечні файли та програми, перешкоджаючи їх впливу на мережу.

Багато кіберзагроз походять зі шкідливих вкладень або поштових відправлень. Антивірусне програмне забезпечення може проводити сканування та фільтрацію електронних повідомлень, щоб виявляти та блокувати віруси, фішингові спроби та інші шкідливі елементи.

Виявлення інтранет-атак є важливим етапом у забезпеченні безпеки локальних мереж. Інтранет-атаки включають різні загрози, які можуть спричинити невідповідну роботу мережі, виток конфіденційної інформації та інші негативні наслідки.

Системи виявлення вторгнень є програмними засобами, які моніторять мережу та аналізують активність з метою виявлення потенційно шкідливих або неавторизованих дій. Вони спираються на виявлення підозрілих патернів, аномалій або підписів, що вказують на вторгнення або атаку в мережі. IDS можуть виявляти такі типи атак, як сканування портів, переповнення буфера, атаки типу "чоловік посередині" та інші.

Системи запобігання вторгненням, як правило, працюють в парі з системами виявлення вторгнень (IDS). Вони не тільки виявляють потенційні вторгнення, але й приймають заходи для блокування атак та захисту мережі. Системи запобігання вторгненням (IPS) можуть виконувати дії, такі як блокування IP-адрес, відключення підозрілих з'єднань, встановлення правил фільтрації пакетів та інші заходи безпеки.

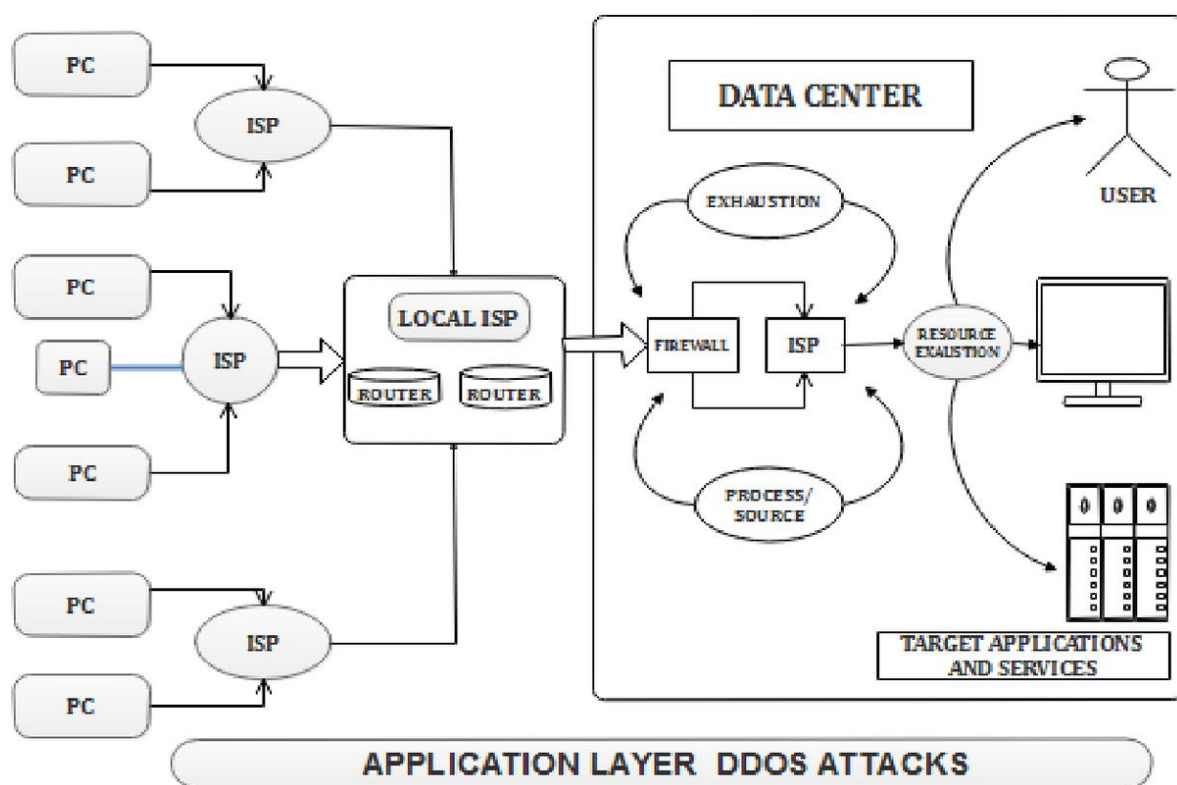


Рисунок 1.1 – Типова схема захисту Data-центру від DDoS-атаки

Джерело: [10]

Моніторинг мережевої активності включає в себе аналіз мережевого трафіку, журналів подій та інших інформаційних джерел для виявлення аномальних або підозрілих дій. Це може включати аналіз змін у паттернах трафіку, виявлення підозрілих з'єднань, аналіз системних журналів та інше. Моніторинг мережевої активності дозволяє оперативно виявляти та реагувати на потенційні загрози.

Захист від DDoS-атак. DDoS-атаки (розподілені атаки з відмовою в обслуговуванні) є одними з найпоширеніших видів кіберзагроз, які спрямовані на перенавантаження ресурсів мережі та обслуговувачів (див. рис. 1).

Для захисту локальних мереж від DDoS-атак можна використовувати наступні програмні засоби:

1) Cloudflare є одним з найпопулярніших сервісів захисту від DDoS-атак. Вони пропонують CDN (Content Delivery Network) разом із захистом від DDoS-атак на різних рівнях (Layer 3 до Layer 7). Cloudflare забезпечує захист від розподілених атак з використанням своїх мережевих ресурсів та аналізу трафіку.

Сервіс має такі можливості:

- надає широкий набір послуг, включаючи CDN та захист від DDoS-атак.
- глобальна мережа серверів дозволяє ефективно розподіляти та фільтрувати трафік.
- використовується масштабними компаніями та веб-сайтами різного розміру.
- має інтуїтивний інтерфейс та широкий спектр налаштувань.

2) Akamai є ще одним провайдером CDN, який пропонує великий набір послуг, у тому числі захист від DDoS-атак. Вони використовують свою глобальну мережу серверів для фільтрації та маршрутизації трафіку, забезпечуючи захист від різноманітних типів атак.

Сервіс має такі можливості:

- глобальна мережа серверів забезпечує швидку доставку контенту та фільтрацію трафіку.
- має багато додаткових функцій та інтеграцію з різними платформами.
- використовується багатьма великими підприємствами та медіа-компаніями.

3) Arbor Networks: Arbor Networks є провідним постачальником рішень для виявлення та захисту від DDoS-атак. Вони пропонують апаратні та програмні рішення для моніторингу та забезпечення безпеки мережі, включаючи виявлення та відвертання DDoS-атак.

Сервіс має такі можливості:

- спеціалізується на виявленні та захисті від DDoS-атак.
- пропонує апаратні та програмні рішення для моніторингу та захисту мережі.
- має великий досвід у боротьбі зі складними атаками та зловживанням протоколів.
- часто використовується провайдерами мережевих послуг та організаціями з високим рівнем безпеки.

4) Radware - це ще одна компанія, що спеціалізується на рішеннях для захисту від DDoS-атак. Вони пропонують різноманітні продукти, включаючи апаратні пристрої та програмне забезпечення, які забезпечують моніторинг, виявлення та захист від DDoS-атак.

Сервіс має такі можливості:

- спеціалізується на рішеннях для захисту від DDoS-атак.
- пропонує апаратні пристрої та програмне забезпечення для виявлення та захисту від атак.
- має широкий набір функціональних можливостей та гнучкі налаштування.
- широко використовується в секторі фінансових послуг та електронної комерції.

Фільтрація трафіку є ефективним методом захисту від DDoS-атак. Вона полягає відокремленні легітимного трафіку від шкідливого, блокуванні IP-адрес атакуючих систем або використанні інших методів для фільтрації небажаного трафіку. Фільтри можуть бути встановлені на мережевих комутаторах, маршрутизаторах або на спеціальних апаратних пристроях.

Деякі програмні засоби захисту від DDoS-атак працюють на рівні програмного забезпечення. Вони використовують різні техніки для виявлення та

блокування великого обсягу небажаного трафіку, наприклад, використання аналізу заголовків пакетів, контролю аутентифікації або перевірки поведінки трафіку.

Розподілені системи миттєвого реагування (CDN) можуть забезпечити захист від DDoS-атак шляхом розподілення трафіку на різні сервери та географічні регіони. Вони мають розгалужену мережу серверів, що дозволяє розподіляти навантаження та фільтрувати небажаний трафік, забезпечуючи надійний доступ до веб-ресурсу навіть під час DDoS-атак.

Ефективний захист від DDoS-атак передбачає постійне моніторинг мережі та реагування на атаки. Це може включати виявлення аномалій у мережевій активності, аналіз журналів подій, спостереження за показниками продуктивності мережі та прийняття необхідних заходів для запобігання або пом'якшення наслідків атаки.

Фізична безпека. Забезпечення безпеки локальної мережі (LAN) є важливим аспектом для будь-якої організації, незалежно від її розміру або сфери діяльності. При розгляді аспектів безпеки мережі часто акцентують увагу на кібербезпеці, такі як захист від хакерських атак та зловживань, але фізична безпека так само важлива для запобігання несанкціонованому доступу до мережевої інфраструктури та захисту від фізичних загроз. У цій статті ми розглянемо деякі основні аспекти фізичної безпеки локальної мережі та важливі кроки, які слід вживати для її забезпечення.

Перш ніж говорити про безпеку самої мережі, важливо забезпечити фізичний доступ до приміщення, де знаходиться мережеве обладнання. Це може включати використання фізичних замків, контролю доступу, системи відеоспостереження та обмеження доступу лише для авторизованого персоналу.

Серверні кімнати є локальної мережі, тому вони повинні бути особливо захищені. Забезпечення фізичного безпеки серверних кімнат може включати встановлення багаторівневих систем контролю доступу, використання вогнегасників та детекторів задимлення, регулярну перевірку стану обладнання та систем аварійного живлення.

Кабелі, що використовуються для підключення комп'ютерів та мережевого обладнання, повинні бути захищені від несанкціонованого доступу. Вони можуть бути закриті в спеціальних кабельних каналах або підлогових системах, які унеможливають фізичний доступ до них.

Встановлення системи відеоспостереження є важливим елементом фізичної безпеки мережі. Вона дозволяє виявити будь-які некоректні дії або несанкціонований доступ до приміщень з мережевим обладнанням. Відеокамери слід розміщувати в стратегічних місцях, таких як входи, коридори та серверні кімнати, і підключати до центральної системи моніторингу.

КРИПТОГРАФІЧНИЙ ЗАХИСТ

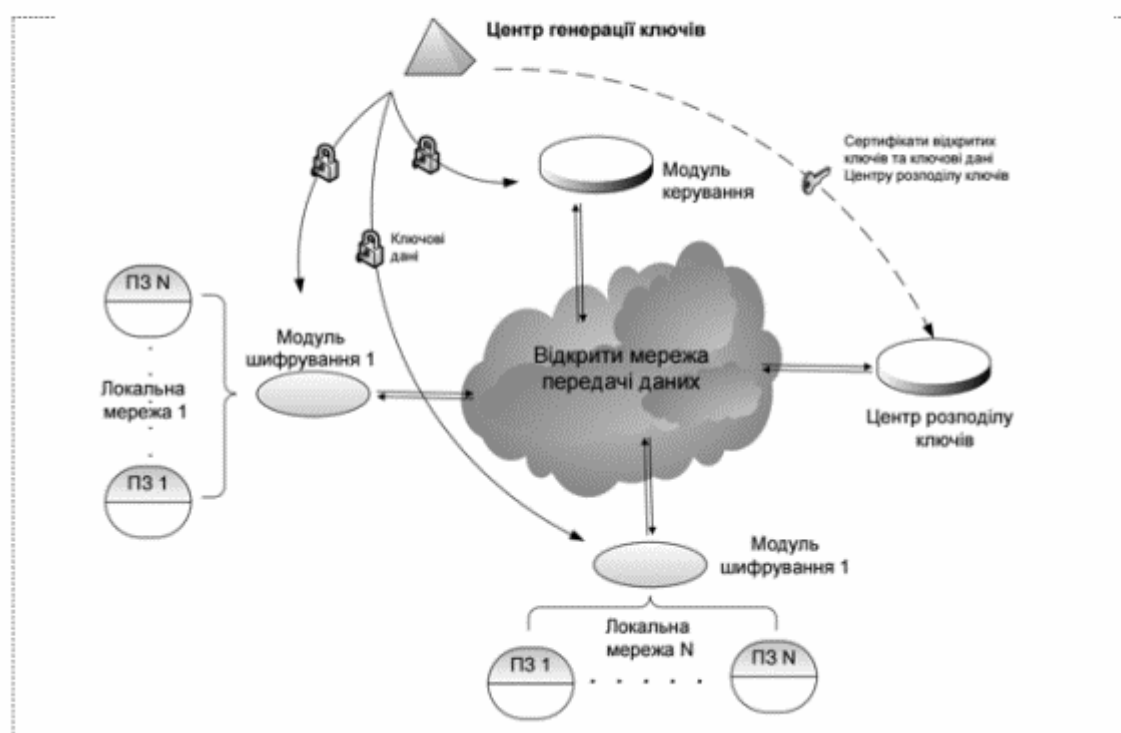


Рисунок 1.2 – Приклад криптографічного захисту окремого абоненту мережі

Джерело: [7]

Фізична безпека мережі також повинна враховувати потенційні фізичні загрози, такі як природні катастрофи (пожежа, повінь) або нещасні випадки (короткозамикання, перегрів). Важливо мати плани невідкладних заходів, резервне

живлення та резервне копіювання даних, а також проводити регулярні перевірки систем безпеки.

Шифрування інформації. У сучасному цифровому світі, де передача та обробка інформації стають все більш важливими завданнями для багатьох організацій, захист конфіденційності та цілісності даних стає надзвичайно важливим. Одним з найефективніших методів захисту інформації є шифрування[1].

Шифрування - це процес перетворення звичайного тексту (зрозумілого для людини) в зашифрований вигляд (незрозумілий без ключа). У шифруванні використовуються різні алгоритми та ключі для забезпечення безпеки даних. Найпоширеніші методи шифрування включають симетричне шифрування, де один і той же ключ використовується як для шифрування, так і для розшифрування, та асиметричне шифрування, де використовуються два ключі: публічний та приватний (див. рис. 2).

Шифрування даних в транспортному каналі є важливим аспектом забезпечення безпеки в локальній мережі. Використання протоколів шифрування, таких як рівень захищених сокетів (SSL)/ захист на транспортному рівні (TLS), дозволяє захистити передачу даних між клієнтом та сервером. Це особливо важливо при передачі конфіденційної інформації, такої як паролі, фінансові дані або особисті дані користувачів.

Крім шифрування даних під час їх передачі, важливо також забезпечити шифрування даних на зберіганні. Це можна досягти шляхом використання шифрованих файлових систем або шифрування окремих файлів та папок. Шифрування даних на зберіганні допомагає запобігти несанкціонованому доступу до даних в разі втрати або крадіжки носіїв інформації.

Ключі є невід'ємною частиною шифрування інформації. Ефективне управління ключами є ключовим аспектом забезпечення безпеки даних. Ключі повинні бути довіреними, довгими та складними для вгадування. Крім того, слід забезпечити їх безпечне зберігання та обмін між відповідними сторонами.

Шифрування не є статичним заходом безпеки. Інформаційні технології постійно змінюються, і нові методи атак можуть виникати з часом. Тому важливо

регулярно оновлювати шифрування та проводити аудит його ефективності. Це дозволить виявити можливі слабкі місця та забезпечити належний рівень безпеки.

Навчання співробітників. Співробітники організації - це слабе місце в системі захисту локальних мереж. Необхідно навчати співробітників про правила безпеки, які стосуються використання паролів, обробки електронної пошти та інших аспектів інформаційної безпеки. Створення політики безпеки та її регулярне оновлення є також важливими кроками.

Використання та захист USB-накопичувачів. У сучасному світі USB-накопичувачі стали невід'ємною частиною нашого цифрового життя. Вони забезпечують легкий доступ до збережених даних та зручне переміщення інформації з одного пристрою на інший. У цій статті ми розглянемо різні аспекти використання USB-накопичувачів у сучасному світі та їх вплив на безпеку даних.

Однією з найбільших переваг USB-накопичувачів є їх переносність та зручність. Вони мають компактний розмір і легко поміщаються в кишеню або сумку. Це дозволяє користувачам з легкістю переміщувати дані між різними пристроями, такими як комп'ютери, ноутбуки, смартфони та планшети.

USB-накопичувачі є зручним засобом для зберігання та резервного копіювання даних. Вони можуть вміщувати значну кількість інформації, включаючи документи, фотографії, відео та музику. Це дозволяє зберігати особисті файли, робочі документи або навіть всю оперативну систему відновлення на випадок аварії.

USB-накопичувачі також часто використовуються для обміну даними між користувачами. Вони дозволяють легко передавати файли, документи та іншу інформацію між колегами, друзями та родичами. Це особливо зручно, коли ви хочете поділитися великими файлами, які можуть бути складні для передачі через електронну пошту або хмарні сервіси.

Незважаючи на всі переваги, USB-накопичувачі також мають потенційні загрози для безпеки даних. Вони можуть бути джерелом поширення вірусів, шкідливого програмного забезпечення та інших шкідливих програм.

Неконтрольоване використання USB-накопичувачів може призвести до втрати чутливих даних або порушення конфіденційності.

Щоб забезпечити безпеку даних при використанні USB-накопичувачів, важливо дотримуватись наступних заходів безпеки:

- а) використовуйте надійне антивірусне програмне забезпечення та перевіряйте USB-накопичувачі на наявність шкідливих програм.
- б) не використовуйте невідомі або недостовірні USB-накопичувачі.
- в) забороніть автоматичне виконання файлів з USB-накопичувачів.
- г) зашифруйте важливі дані на USB-накопичувачі, щоб запобігти несанкціонованому доступу у разі втрати чи крадіжки.

Усім своїм перевагам та зручностям USB-накопичувачі також потребують обережного та відповідального використання. З дотриманням необхідних заходів безпеки, вони можуть бути надійним та зручним засобом збереження та переміщення даних.

Розглянемо різні засоби захисту інформації на USB-накопичувачах:

- Використання паролів. Одним з найпростіших, але ефективних способів захисту даних на USB-накопичувачах є використання паролів. Багато USB-накопичувачів мають функцію захисту паролем, яка дозволяє зашифрувати дані та вимагати введення пароля перед доступом до них. Важливо використовувати сильні паролі, які складаються з комбінації великих і малих літер, цифр і символів. Крім того, рекомендується регулярно змінювати паролі для забезпечення додаткової безпеки.

- Шифрування даних. Ще одним ефективним способом захисту інформації на USB-накопичувачах є шифрування даних. Шифрування дозволяє перетворити дані в незрозумілу форму, що забезпечує їх конфіденційність у разі втрати або крадіжки накопичувача. Існують різні програми індивідуального шифрування, які дозволяють зашифрувати файли та папки на USB-накопичувачі. Також існують спеціальні шифровані USB-накопичувачі, які автоматично шифрують дані при їх записі на накопичувач і розшифровують при читанні.

- Використання захищених програм. Деякі USB-накопичувачі мають вбудовані захищені програми, які дозволяють створювати захищені віртуальні простори на накопичувачі. Ці простори можуть бути захищені паролем або шифруванням, і вони дозволяють зберігати конфіденційну інформацію в окремому захищеному середовищі. Такі програми часто надають можливість захищеної навігації, керування правами доступу та автоматичного блокування накопичувача після певного періоду бездіяльності.

- Віддалене видалення даних. Якщо ви втратили свій USB-накопичувач або він був викрадений, важливо мати можливість віддалено видалити дані з пристрою. Деякі програми та служби дистанційного керування дозволяють віддалено виконувати команди знищення даних на накопичувачі. Це може бути корисно, щоб запобігти несанкціонованому доступу до конфіденційної інформації.

Забезпечення належного рівня захисту інформації на USB-накопичувачах є важливим завданням. Застосування паролів, шифрування даних, використання захищених програм та віддалене видалення даних - це лише кілька засобів, які можуть допомогти забезпечити безпеку даних.

Окрім захисту інформації на USB-накопичувачах, треба захистити інформацію від шкідливих USB-накопичувачів, бо вони можуть бути потенційними джерелами загроз для безпеки вашого персонального комп'ютера і конфіденційної інформації. Розглянемо деякі засоби захисту інформації на персональному комп'ютері від шкідливих USB-накопичувачів.

Один з найважливіших кроків у захисті вашого комп'ютера від шкідливих USB-накопичувачів - це встановлення та оновлення надійного антивірусного програмного забезпечення. Антивірусна програма здатна виявляти та блокувати потенційно небезпечні файли та програми, які можуть бути передані через USB-накопичувачі. Вона також може аналізувати накопичувачі на наявність вірусів і шкідливих програм під час підключення до комп'ютера.

Багато шкідливих програм використовують функцію автозапуску, щоб виконуватися автоматично після підключення USB-накопичувача. Важливо налаштувати ваш комп'ютер таким чином, щоб автоматичний запуск програм з

USB-накопичувачів був блокований. Це можна зробити шляхом налаштування опцій автозапуску в операційній системі або використання спеціальних програм, які контролюють цю функцію.

USB-фільтри - це пристрої або програми, які дозволяють контролювати доступ до USB-портів вашого комп'ютера. Вони можуть бути встановлені як апаратні фільтри, що підключаються до USB-порту, або програмні фільтри, які працюють на рівні операційної системи. Вони дозволяють встановлювати політики доступу, блокувати підозрілі USB-пристрої та виконувати інші заходи безпеки.

Для захисту конфіденційної інформації, яка зберігається на USB-накопичувачі, варто розглянути можливість використання криптографічних засобів шифрування. Деякі програми дозволяють створювати захищені контейнери або зашифровувати файли та папки на накопичувачі. Це забезпечує додатковий рівень захисту, оскільки навіть у разі втрати накопичувача інформація залишається зашифрованою та недоступною для злоумисників.

Регулярне оновлення операційної системи і програмного забезпечення на вашому комп'ютері також є важливим для забезпечення безпеки. Виробники надають оновлення, які виправляють виявлені уразливості і забезпечують більш надійний захист. Важливо мати оновлення встановлені на вашому комп'ютері, щоб уникнути можливих атак, пов'язаних з USB-накопичувачами.

1.2 Методи побудови програмного забезпечення та даних про стан корпоративної мережі

Побудова програмного забезпечення є складним та важливим процесом, що вимагає систематичного підходу та використання відповідних методів. Розглянемо декілька основних методів побудови програмного забезпечення, які допомагають забезпечити якість, ефективність та успішність розробки програмних продуктів.

Каскадна модель розробки є одним з найбільш традиційних та широко використовуваних методів. Вона передбачає послідовний підхід до розробки, де кожна фаза проекту виконується послідовно: аналіз вимог, проектування,

реалізація, тестування та впровадження. Каскадна модель дозволяє чітко визначити кожен етап проекту та планувати роботу, але може бути менш гнучкою в умовах, коли вимоги можуть змінюватися протягом процесу розробки.

Ітеративний та інкрементальний підхід передбачає поділ процесу розробки на ітерації або фази. Кожна ітерація включає в себе етапи аналізу, проектування, реалізації та тестування, але з фокусом на конкретну частину функціоналу. Кожна ітерація додає новий функціонал або поліпшення до програмного продукту. Цей підхід дозволяє забезпечити більшу гнучкість та швидкість реагування на змінні вимоги, а також забезпечити більш ранню зворотну зв'язок зі замовником.

Розробка на основі компонентів передбачає використання готових компонентів або модулів для побудови програмного продукту. Цей метод дозволяє використовувати готовий функціонал інших розробників, що прискорює процес розробки, поліпшує якість та забезпечує перевикористання коду. Важливим аспектом розробки на основі компонентів є правильний вибір компонентів, їх адаптація до потреб проекту та забезпечення сумісності між компонентами.

Гнучкі методології розробки, такі як Scrum, Kanban або Extreme Programming (XP), стали популярними в останні роки. Вони передбачають ітеративний підхід, наголошують на самоорганізації команди, регулярному зворотному зв'язку зі замовником та адаптації до змінних вимог. Гнучкі методології підходять для проектів, де вимоги можуть змінюватися протягом процесу розробки, та забезпечують більшу прозорість та взаємодію в команді.

DevOps - це підхід до розробки програмного забезпечення, який поєднує розробку та операції. DevOps став популярним в контексті швидкого розгортання та доставки програмного забезпечення. Цей підхід передбачає автоматизацію процесів розробки, тестування, впровадження та моніторингу, що допомагає скоротити час доставки та покращити якість програмного продукту.

Ці методи побудови програмного забезпечення є лише кількома з багатьох існуючих підходів. Вибір підходу залежить від конкретних умов та вимог проекту. Часто розробники комбінують різні методи, використовуючи їх переваги для досягнення найкращих результатів.

Важливо також зазначити, що успіх побудови програмного забезпечення залежить не лише від використаних методів, але й від команди розробників, їх досвіду, співпраці та професіоналізму. Здійснення регулярного контролю, тестування та зворотного зв'язку є також важливими аспектами для досягнення якісного програмного продукту.

Методи побудови подій та даних про стан корпоративної мережі. Корпоративна мережа є нерозривною складовою сучасного бізнесу. Її ефективна робота та безпека є ключовими факторами для забезпечення продуктивності та неперервності роботи організації. Однак, у зв'язку зі зростаючою складністю мережевих інфраструктур, необхідно мати методи побудови подій та даних про стан корпоративної мережі для забезпечення її ефективного управління та захисту. Розглянемо декілька методів, які допомагають побудувати події та дані про стан корпоративної мережі.

Журналювання подій є важливим інструментом для збору і реєстрації подій, що відбуваються в корпоративній мережі. Цей метод передбачає ведення журналів подій з різних мережевих пристроїв, таких як маршрутизатори, комутатори, файрволи тощо. Журнали подій містять інформацію про зміни конфігурації, підключення та відключення пристроїв, спроби несанкціонованого доступу, помилки та інші події. Аналіз журналів дозволяє виявити проблеми, виявити загрози безпеці та вжити відповідних заходів.

Моніторинг стану мережі полягає в постійному відстеженні параметрів та характеристик мережевих пристроїв, сервісів та додатків. Цей метод дозволяє виявляти аномальні або некоректні поведінку, прогнозувати навантаження та вчасно реагувати на проблеми. Моніторинг може бути здійснений за допомогою спеціальних програмних засобів, які забезпечують збір, аналіз та візуалізацію даних про стан мережі.

Протоколи мережевого управління, такі як SNMP (Simple Network Management Protocol), дозволяють здійснювати моніторинг та управління мережевими пристроями. Вони надають можливість зчитувати статус, статистику та конфігурацію пристроїв, а також виконувати дії, наприклад, перезавантаження

або зміну конфігурації. Використання протоколів мережевого управління спрощує процес контролю та управління корпоративною мережею.

Системи управління подіями та інформацією про безпеку (SIEM) є потужним інструментом для збору, аналізу та сповіщення про події, пов'язані з безпекою мережі. SIEM-системи інтегрують дані з різних джерел, включаючи журнали подій, мережеві пристрої, системи виявлення вторгнень та інші, та забезпечують аналіз та кореляцію цих подій для виявлення загроз безпеці. Вони також надають можливість швидкого реагування на інциденти безпеки та вжиття відповідних заходів.

1.3 Аналіз наявних рішень для вирішення проблеми

Під час аналізу наявних рішень, було виявлено п'ять програмних продуктів, які можуть виконати задачу моніторингу підключень USB-пристроїв повністю або частково. Давай розглянемо їх детальніше.

USB Canary. Ця утиліта написана на мові програмування Python і наразі працює тільки на операційній системі Linux, але розробник працює над версіями для Windows і Mac OS. Принцип роботи утиліти досить простий: коли комп'ютер заблокований, а його власник покинув робоче місце, USB Canary спостерігає за USB-роз'ємами і виявляє будь-яку активність. Якщо хтось намагатиметься підключити пристрій до комп'ютера або, навпаки, відключити його, USB Canary повідомить власника ПК через SMS-повідомлення, використовуючи Twilio API, або через відповідне сповіщення в Slack-каналі. Цей канал можуть переглядати колеги користувача або системний адміністратор. Крім того, утиліту можна налаштувати на певні години, що дозволить виявляти підключення в неробочий час. USB Canary працює навіть у випадку, коли комп'ютер заблокований.

Ця утиліта має ряд недоліків. Відсутність версії для операційної системи Windows, яка є найпоширенішою у світі, є одним з основних недоліків цього продукту. Така відсутність обмежує кількість потенційних користувачів. Крім того, утиліта має обмежений функціонал для моніторингу підключень USB-пристроїв у

великих компаніях. Адміністраторам безпеки потрібно відстежувати велику кількість комп'ютерів, і без централізованого аналізу подій це майже неможливо зробити ефективно. Висновок полягає в тому, що функціонал USB Canary є достатнім для невеликих компаній, які використовують Linux, або для персонального використання, що дозволить повідомити користувача про спробу викрасти файл під час його відсутності.

USB-Deview. Це програма, яка не лише відображає інформацію про USB-пристрої, підключені на момент запуску, але і всі пристрої, що підключалися до комп'ютера у минулому. Завдяки USB-Deview можна переглянути список всіх USB-пристроїв, включаючи назву, короткий опис, тип, виробника та серійний номер. Додаткові відомості включають дату останнього використання, поточний статус підключення та можливість безпечного відключення. Крім того, отримана інформація може бути експортована в формати txt, HTML або XML.

Утиліта відображає всю цю інформацію в зручній табличній формі. Користувач може налаштувати кількість та розташування стовпців і сортувати їх за потребою.

У меню "Файл" USBDeview є опції для взаємодії з USB-пристроями, такі як включення або відключення окремих пристроїв або відключення всіх одночасно. Також користувач може налаштувати вивід стовпців відповідно до своїх потреб.

Крім того, програма USB-Deview допомагає очистити системний реєстр від непотрібних записів, що покращує швидкість його роботи.

Проте, подібно до USB Canary, недоліком USB-Deview є відсутність можливості централізованого перегляду подій. Без такого централізованого моніторингу USB-підключень час реагування на витік інформації затримується, що може призвести до значних збитків для компанії.

Хоча USB-Deview детально відображає інформацію про USB-пристрої, відсутність можливості переглянути історію підключень обмежує її ефективність у моніторингу. Відображення тільки останніх підключень ускладнює встановлення повної хронології дій користувача і може ускладнити доказ провини зловмисника.

CleverControl є рішенням для моніторингу дій співробітників на комп'ютерах. Цей інструмент допомагає підтримувати оптимальну продуктивність працівників, підвищувати ефективність роботи та запобігати незаконним діям, таким як витік інформації, крадіжка особистих даних та несанкціонований доступ. CleverControl надає користувачам можливість віддаленого контролю та моніторингу різних комп'ютерних дій в режимі реального часу, включаючи веб-серфінг, пошук в Інтернеті, чати, електронну пошту, соціальні мережі та обмін файлами. Вона також відстежує взаємодію з іншими пристроями та сховищами, такими як принтери та USB-накопичувачі.

CleverControl є хмарною платформою для моніторингу, до якої можна отримати доступ через веб-аккаунт. Для його використання не потрібен власний сервер, а його налаштування є простим і не вимагає досвіду ІТ-фахівця.

Однак, CleverControl має обширний функціонал, що може бути зайвим для деяких користувачів і призводити до додаткових витрат.

Надання сервісу у форматі хмарного рішення може бути як перевагою, так і недоліком. З одного боку, це звільняє від необхідності мати власний сервер та адміністратора, який забезпечуватиме роботу системи. З іншого боку, при використанні цього сервісу, компанія передає всі свої дані на сервери CleverControl, що може призвести до потенційного ризику витоку інформації через цей сервіс.

Крім того, для використання CleverControl потрібна щомісячна підписка, без можливості одноразової покупки продукту та самостійного розгортання його на власних серверах.

Solarwinds Security Event Manager. Менеджер подій безпеки (SEM) - це активне рішення SIEM для моніторингу, яке автоматично виявляє, сповіщає та реагує на підозрілу поведінку на мережевих пристроях, серверах, робочих станціях і додатках різних постачальників. SEM є завантажувальним програмним комплексом, який швидко розгортається й забезпечує оперативний аналіз загроз і кореляцію подій у реальному часі, що дозволяє більш швидко реагувати на кібератаки. Цей програмний продукт має широкий функціонал, який включає 58

випадків використання. Давайте тепер детальніше розглянемо випадок використання, який вирішує задачу моніторингу підключень USB-пристроїв.

За допомогою функції USB Security Analyzer Detects and Responds to Potential Threats можна підвищити безпеку USB-порту, використовуючи попередньо встановлені правила активної відповіді або створюючи власні правила. Правила активної реакції призначені для блокування USB-пристроїв, коли доступ до USB становить загрозу, наприклад, якщо пристрій занесено до чорного списку. Використовуючи USB Defender, можна налаштувати правила активної відповіді, щоб автоматично перевіряти, чи належить пристрій до білого списку USB-пристроїв, визначеного користувачем. Якщо виявлено несанкціоноване підключення до USB-порту, вживаються запобіжні заходи, такі як автоматичне відключення USB-пристрою, зупинка процесів за ідентифікатором або іменем, блокування IP-адрес і вимкнення машини. USB Defender також реєструє події, пов'язані з USB-пристроями, підключеними до агентів SEM за замовчуванням. Ці події можуть бути проаналізовані для отримання додаткової інформації для створення більш складних чорних списків. SEM пропонує як вбудовані, так і настраювані шаблони звітів.

Однак, на відміну від функції USB Security Analyzer Detects and Responds to Potential Threats, Solarwinds Security Event Manager має ще 57 інших функцій моніторингу та управління різними аспектами системи, які для кінцевого користувача можуть бути надлишковими для вирішення поставленої задачі. Хоча користувач може не встановлювати непотрібні йому функції, це не впливає на кінцеву ціну підписки, яка залежить від кількості вузлів, на яких буде встановлений програмний продукт. На момент написання ціна річної підписки починається від 4 805 доларів США за 30 вузлів, а ціна обслуговування понад 100 вузлів становить 14 694,86 доларів США, що робить цей продукт досить дорогим для невеликих або навіть середніх компаній.

Звідси ми можемо зробити висновок, що цей продукт є хорошим вибором, переважно для великих компаній з високими вимогами до інформаційної безпеки.

Таблиця 1.1 – Порівняльна характеристика наявних рішень

Назва критерію	Назва рішення				
	USB Canary	USB-Deview	Clever Control	SEM	Alient Vault
Можно розвернути на власних ПК	+	+	-	+	-
Централізований перегляд подій	-	-	+	+	+
Можливість перегляду історії	-	-	+	+	+
Перегляд даних про USB-прист.	-	Не повна	+	+	+
Має недостатній функціонал	+	+	-	-	-
Містить зайві функції	-	-	+	+	+
Підтримка Windows	-	+	+	+	+
Зрозумілий інтерфейс	-	+	+	+	+
Ціна	Безкоштовно	Безкоштовно	Від 180\$/рік	Від 3,805\$ за 30 вузлів/рік	1075\$ \ місяць

Джерело : [5]

Платформа AlienVault є потужним і уніфікованим рішенням, яке дозволяє організаціям швидко виявляти загрози і реагувати на них, комбінуючи різноманітні функції безпеки. Вона включає в себе виявлення активів, управління вразливостями, виявлення вторгнень, моніторинг поведінки, SIEM та управління журналами. AlienVault розроблений для захисту хмарної, локальної та гібридної інфраструктури, забезпечуючи постійний аналіз загроз і ефективну реакцію на інциденти. Вона також підтримує такі сервіси, як Office 365, Google Suite, Microsoft Azure Cloud та Amazon Web Services. За допомогою зручного графічного інтерфейсу та візуалізації даних, включаючи діаграми, графіки, карти та інші графічні елементи, AlienVault надає користувачам зручні інструменти для аналізу та управління безпекою.

Проте, варто відзначити, що AlienVault працює на базі хмари, що означає, що користувачам потрібно надсилати свої корпоративні дані на сервери AlienVault для моніторингу USB-пристроїв та забезпечення безпеки.

РОЗДІЛ 2

ПРОЄКТУВАННЯ ЛОКАЛЬНОЇ МЕРЕЖІ ТА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ

2.1 Постановка задачі та визначення мети

Метою проекту є розробка сучасної локальної мережі та створення системи моніторингу підключень USB-пристроїв для проведення аудиту кібербезпеки.

Розробка системи орієнтована на використання в коледжі, який мають потребу забезпечити контроль за USB-пристроями та не має змоги придбати дорогі системи з непотрібними для вирішення цієї задачі функціями. Система зменшить час реагування на витік інформації через USB-накопичувачі та створить доказову базу, яка допоможе визначити особу порушника та довести його провину.

Отримано декілька завдань:

- проаналізувати наявний стан мережі;
- розробити віртуальну модель удосконаленої мережі та перевірити її на працездатність;
- надати рекомендації що до фізичного удосконалення мережі та обґрунтувати їх;
- надати рекомендації що до програмного удосконалення мережі та обґрунтувати їх;
- розробити власний програмний продукт для перевірки USB портів абонентів мережі для покращення захищеності мережі та витоку інформації з неї.

Розроблювана інформаційна система має такі вимоги:

- відслідковувати події підключення на корпоративних комп'ютерах;
- забезпечити централізований збір та збереження подій, для подальшого аналізу;
- мати інтуїтивно зрозумілий інтерфейс для зручного перегляду та подій;
- забезпечувати можливість пошуку потрібних подій;

- здатність розгорнутися локально;

Задачі, які потрібно вирішити для досягнення зазначених вимог:

- здійснити аналіз предметної області;
- дослідити документацію Windows;
- обрати засоби для розробки системи;
- налаштувати середовище для розробки програмного коду;
- розробити службу клієнт, для спостереження за USB-пристроями.

Коротка характеристика підприємства.

Назва: Відокремлений структурний підрозділ «Глухівський агротехнічний фаховий коледж Сумського НАУ», Україна, Сумська область, м.Глухів, вул. Терещенків, 36.

Заклад здійснює свою діяльність відповідно до Конституції та законів України, указів Президента, постанов, указів і розпоряджень Кабінету Міністрів України, нормативних актів Міністерства освіти і науки України, Статуту Сумського національного аграрного університету, Положення про організацію освітнього процесу у Глухівському агротехнічному фаховому коледжі Сумського національного аграрного університету.

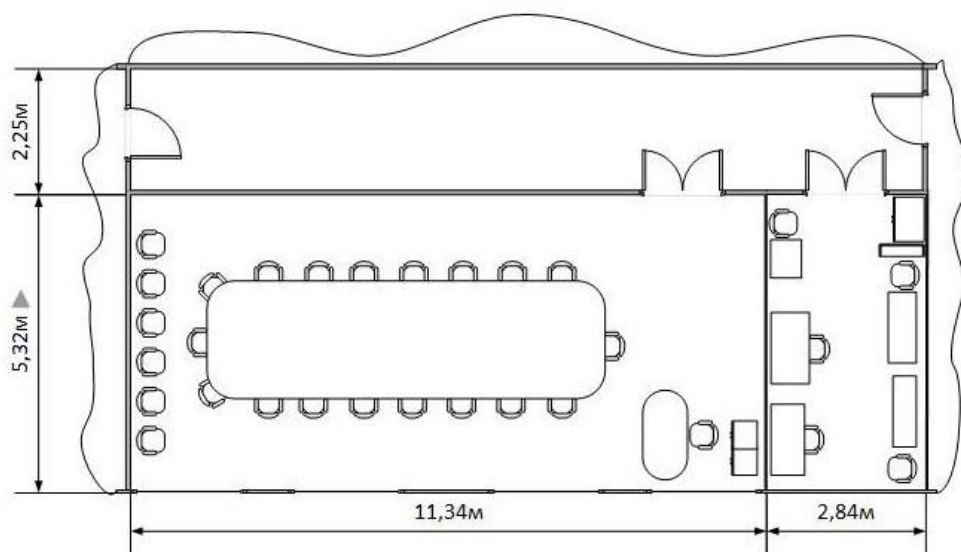


Рисунок 2.1 - План комп'ютерної лабораторії організації – креслення у масштабі

Джерело: запропоновано автором

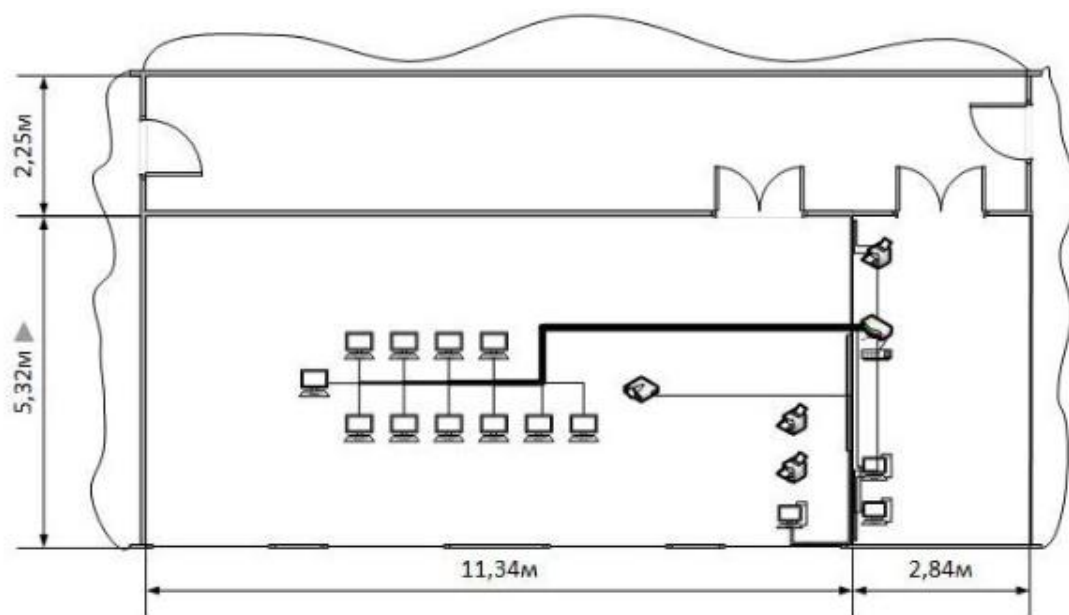


Рисунок 2.2 - Фрагмент комп'ютерної мережі організації, креслення у масштабі

Джерело: запропоновано автором

Загальна характеристика комп'ютерної мережі підприємства

Топологія мережі - дерево.

Тип мережі: Fast Ethernet.

Обладнання для підключення: В даній мережі використовується кабель вита пара 5е категорії, з 4 парами без екранування, вартість ~540 грн, у мережі використовується маршрутизатор TP-LINK WR841N та комутатор D-LINK DGS-1024C.

Тип ліній зв'язку: оптичне волокно, бездротові мережі, коаксіальний кабель, вита пара, супутниковий зв'язок.

Вартість ліній зв'язку та обладнання мережі: вита пара категорії 5е.

1) Обладнання :

комутатор- 2 466 грн.

маршрутизатор- 669 грн.

2) Лінії зв'язку:

Ціна за 1 метр 5е = 3,59 грн.. Ціна 150 м-503 грн.

Разом: $2466,00 + 669,00 + 503 = 3638$ грн.

Код, який використовується в мережі: 4В/5В.

Протоколи мережі : TCP/IP, HTTP, DHCP, DNS, FTP, MAC, SMTP.

Методи керування обміном: централізований випадковий.

Масштабування мережі до мережі можливо підключити фізично 5 абонентів та 30 по Wi-Fi.

Максимальна довжина лінії зв'язку, що сполучає усі концентратори: 1.5 м.

Максимальна довжина лінії зв'язку між комутатором і маршрутизатором: 1.5м.

Максимальна довжина ліній зв'язку між абонентом та концентратором: 13м.

Пікове навантаження : при включених всіх абонентах, мережа здатна забезпечити передачу без збоїв і максимальну швидкість передачі, ~60Мбіт/с, в середньому на абонента припадає близько 15Мб/с.

Швидкість передачі: 100Мб/с, на одного абонента ~5.88Мб/с.

Прикладне програмне забезпечення мережі: Radmin Server, UserGate, Google Chrome, Mozilla Firefox, Microsoft Office, Photoshop.

Кількість абонентів: 17.

Основні напрямки використання мережі, її призначення.

Мережа виконує велике коло завдань, такі як: обмін даними у мережі між абонентами, робота з єдиною електронною державною базою, обробка документації, реєстрація вступників до навчального закладу. Також у коледжі, одна з аудиторій використовується для проведення різного роду семінарів, вебінарів, які дозволяють обмінюватися досвідом не лише з іншими навчальними закладами в Україні, але й закордонними партнерами з Німеччини, за рахунок інтерактивної дошки, проектору, сучасних моноблоків. Мережа, також, дозволяє вести навчальний процес, в цьому знову ж таки, допомагає обладнання аудиторії та актуальні презентації. Не можливі б були й різноманітні тренінги для вчителів та студентів.

Технологія Ethernet — самий доступний і якісний стандарт локальних мереж на сьогоднішній день. Загальна кількість мереж, що працюють по протоколу Ethernet в даний час, оцінюється в 5 мільйонів, а кількість комп'ютерів з встановленими мережними адаптерами Ethernet - у 50 мільйонів. Цей протокол

дозволяє в кожний момент часу лише один сеанс передачі в логічному сегменті мережі. При появі двох і більше сеансів передачі одночасно виникає колізія, яка фіксується станцією, що ініціює передачу. Станція аварійно зупиняє процес і очікує закінчення поточного сеансу передачі, а потім знову намагається повторити передачу. Ethernet-мережі функціонують на швидкостях 10 Мбіт/с, Fast Ethernet — на швидкостях 100 Мбіт/с, Gigabit Ethernet—на швидкостях 1000 Мбіт/с, 10 Gigabit Ethernet — на швидкостях 10 Гбіт/с. В кінці листопада 2006 року було прийняте рішення про початок розробок наступної версії стандарту з досягненням швидкості 100 Гбіт/с (100 Gigabit Ethernet).

Переваги мережі коледжу: простота установки. Добре відома і найбільш поширена мережева технологія. Невисока вартість мережевих карт. Можливість реалізації з використанням різних типів кабелю і схем прокладки кабельної системи.

Недоліки мережі коледжу: зниження реальної швидкості передачі даних під час використання мережі всіма абонентами одночасно, аж до її повної зупинки через конфлікти в середовищі передачі даних. Труднощі пошуку несправностей: при обриві кабелю відмовляє весь сегмент ЛОМ, і локалізувати несправний вузол або ділянку мережі досить складно.

2.2 Вибір методів реалізації та створення проєкту

Для реалізації створення віртуальної моделі локальної мережі була обрана середовище моделювання Cisco Packet Tracer. Для реалізації системи моніторингу підключень USB-пристроїв були вибрані середовище програмування Microsoft Visual Studio, а також мова програмування C#.

Частина системи, яка відстежує події підключень USB-пристроїв, буде реалізована у вигляді Windows служби. Для реалізації Windows служби широко використовується платформа .NET Framework, розроблена компанією Microsoft. Ця платформа дозволяє запускати додатки на операційній системі Windows, написані на мовах, таких як C# або Visual Basic.

C# є сучасною об'єктно-орієнтованою типізованою мовою програмування, розробленою компанією Microsoft. Вона призначена для використання у Common Language Infrastructure, яка включає виконуваний код та середовище виконання. У порівнянні з мовами програмування C та C++, у розробника не потрібно піклуватися про звільнення пам'яті, що сприяє зменшенню кількості помилок і полегшує процес розробки. C# має обширну документацію, яку можна знайти на веб-сайті Microsoft.

Розробка моделі варіантів використання. Завданням діаграми варіантів використання є опис можливих сценаріїв використання системи, програмного продукту або послуги. Ця діаграма є однією з ключових складових процесу розробки програмного забезпечення, оскільки вона допомагає розробникам краще зрозуміти потреби та очікування користувачів.

Діаграма варіантів використання складається з акторів, варіантів використання та зв'язків між ними. Актори - це люди, системи або інші сутності, які взаємодіють з системою. Варіанти використання - це конкретні дії, які виконуються акторами, щоб досягти певної мети. Зв'язки між акторами та варіантами використання показують, як актори взаємодіють з системою для досягнення своїх мет.

Діаграми варіантів використання можуть бути корисним інструментом для комунікації між командами розробників та користувачами. Вони допомагають уточнити вимоги та очікування користувачів та дозволяють розробникам краще зрозуміти, як користувачі взаємодіють з системою.

Щоб створити ефективну діаграму варіантів використання, слід детально вивчити бізнес-процеси та потреби користувачів. Крім того, слід враховувати можливість розширення та модифікації системи в майбутньому.

Відповідно до багатьох досліджень, використання діаграм варіантів використання дозволяє знизити ризики та підвищити якість програмного забезпечення. Крім того, ці діаграми є корисним інструментом для навчання та опису функціональностей системи або програмного продукту.

Наприклад, діаграми варіантів використання можуть бути використані для вивчення потреб користувачів та визначення функцій, які необхідні для задоволення цих потреб. Крім того, вони можуть бути використані для ідентифікації можливих проблем взаємодії користувачів з системою та вирішення цих проблем на ранніх етапах розробки.

Одним з важливих аспектів розробки діаграм варіантів використання є врахування потреб користувачів з різним рівнем технічної освіти та досвіду використання комп'ютерів. Для цього можна використовувати різні техніки представлення інформації, такі як візуальні елементи, текстові описи та ілюстрації.

Одним з найпоширеніших інструментів для створення діаграм варіантів використання є програмне забезпечення UML (Unified Modeling Language). UML надає стандартний набір символів та нотацій для побудови діаграм варіантів використання, що дозволяє розробникам та користувачам легко зрозуміти та інтерпретувати ці діаграми.

Узагальнюючи, діаграма варіантів використання є важливим інструментом для розробки програмного забезпечення, оскільки вона допомагає визначити потреби та очікування користувачів та забезпечує зв'язок між акторами та варіантами використання. Діаграми варіантів використання можуть бути використані для зниження ризиків та підвищення якості програмного забезпечення, а також для навчання та опису функціональностей системи або програмного продукту. Для їх ефективного використання важливо враховувати потреби користувачів з різним рівнем технічної освіти та досвіду використання комп'ютерів. Нарешті, програмне забезпечення UML є одним з найпоширеніших інструментів для створення діаграм варіантів використання, що дозволяє розробникам та користувачам легко зрозуміти та інтерпретувати ці діаграми.

Важливо зазначити, що діаграма варіантів використання не є універсальним інструментом та не підходить для всіх видів програмного забезпечення. Наприклад, для деяких систем може бути більш ефективним використання діаграм класів або діаграм послідовності. Однак, діаграма варіантів використання є важливим інструментом, який може бути використаний в більшості проектів з розробки

програмного забезпечення. Для створення діаграм використання було вибрано програмне забезпечення Microsoft Visio.

Microsoft Visio - це програмне забезпечення для створення діаграм, схем, планів та інших візуальних представлень. Воно дозволяє користувачам легко створювати, редагувати і ділитися професійними діаграмами за допомогою готових шаблонів та інструментів.

Visio має широкий набір функцій, які допомагають представляти різні типи даних, включаючи організаційні структури, мережі, схеми баз даних, блок-схеми, процеси, проектні плани та багато іншого. Користувачі можуть створювати власні діаграми з нуля або використовувати вбудовані шаблони для швидкого старту.

Visio надає можливість використовувати різні форми, лінії, кольори, текстові блоки та інші елементи для налаштування зовнішнього вигляду діаграми. Крім того, він підтримує підключення до зовнішніх джерел даних, що дозволяє створювати діаграми, що оновлюються автоматично на основі актуальних даних.

Visio також забезпечує можливість співпрацювати з іншими користувачами, дозволяючи ділитися діаграмами та працювати над ними одночасно в режимі реального часу. Користувачі можуть експортувати діаграми в різні формати файлів, такі як PDF, зображення, документи Microsoft Office і багато іншого.

Створено діаграму варіантів використання для системи «Удосконалена мережа»(див. рис. 2.3).



Рисунок 2.3 - Діаграма варіантів використання для системи «Удосконалена мережа».

Джерело: запропоновано автором

Створено діаграму варіантів використання для системи «Програма USB моніторингу» (див. рис. 2.4).

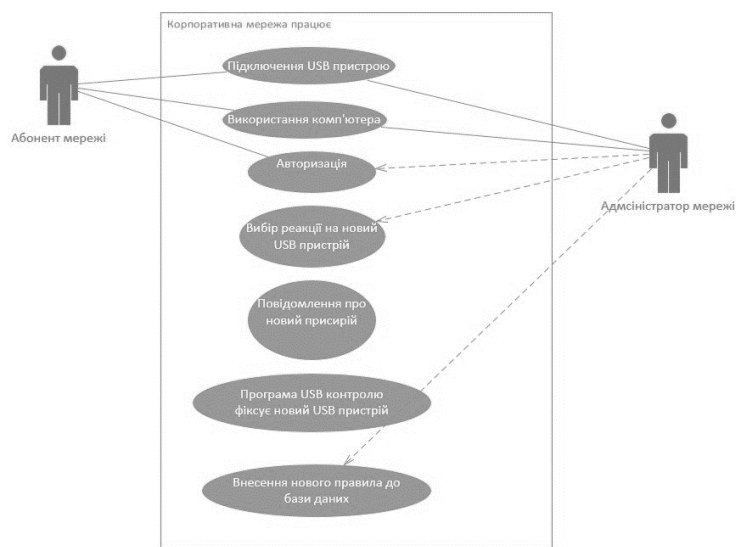


Рисунок 2.4 - Діаграма варіантів використання для системи «Програма USB моніторингу»

Джерело: запропоновано автором

Створено діаграму варіантів використання для системи «Обробка та повідомлення нових USB пристроїв» (див. рис. 2.5).

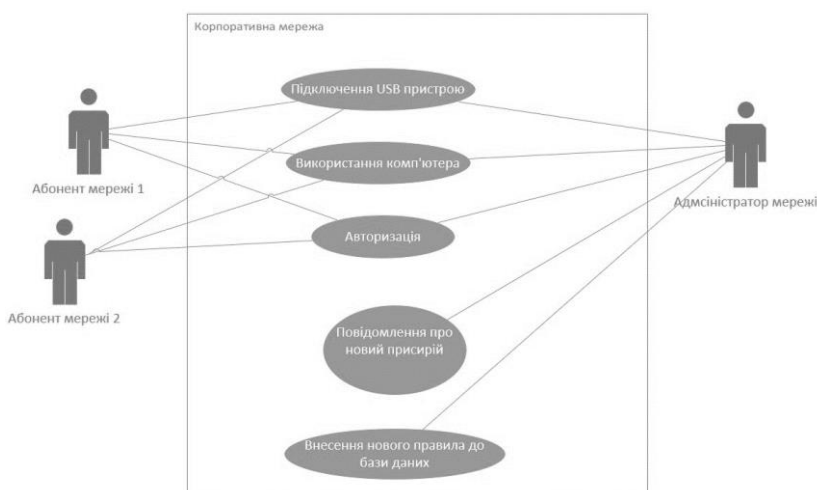


Рисунок 2.5 - Діаграма варіантів використання для системи «Обробка та повідомлення нових USB пристроїв»

Джерело: запропоновано автором

Діаграма варіантів використання є потужним інструментом для розробки програмного забезпечення, який дозволяє визначити потреби користувачів та забезпечує зв'язок між акторами та варіантами використання. Це може допомогти знизити ризики та підвищити якість програмного забезпечення, а також навчити та описати функціональності системи або програмного продукту.

Розробка мережевого графіка проєкту. Мережева модель - це форма графічного відображення змісту, тривалості та послідовності виконання заходів щодо реалізації планів будь-якого характеру і призначення, а також потреб в економічних ресурсах. На відміну від простих лінійних графіків і табличних розрахунків мережеві методи планування дозволяють розробляти та оптимізувати розвиток складних виробничих систем в аспекті їх довгострокового використання. Мережевий графік - це метод планування та контролю робіт за допомогою графічного зображення залежності між задачами[2].

Розглянемо основні компоненти мережевого графіка. Він складається з вузлів, що представляють задачі, та дуг, що показують залежність між ними. У мережевому графіку важливо розуміти три елементи: початок, тривалість та завершення. Ці елементи допомагають зрозуміти часові обмеження та послідовність виконання робіт.

Один з найбільш відомих методів мережевого графіка - це метод СРМ (Critical Path Method), який використовується для визначення критичного шляху проєкту та оцінки часу виконання проєкту. Критичний шлях - це послідовність задач, що має найбільший вплив на завершення проєкту. Оцінка часу виконання допомагає керівникам проєктів зрозуміти, скільки часу потрібно для виконання кожної задачі, та збільшити ефективність проєкту.

Інший метод мережевого графіка - це метод PERT (Program Evaluation and Review Technique), який використовується для оцінки ризику проєкту та покращення його планування. Цей метод використовується для визначення часу виконання кожної задачі та визначення можливих затримок в проєкті. PERT використовує статистичні методи для визначення часу виконання кожної задачі та

її можливих затримок. Це допомагає менеджерам проектів покращити планування та зменшити ризики, пов'язані з проектом.

Для ефективного використання мережевого графіка в проектному менеджменті необхідно дотримуватися кількох правил. Перш за все, необхідно правильно визначити послідовність виконання задач та залежності між ними. Друге правило полягає в правильному розподілі ресурсів, таких як людські ресурси та матеріальні ресурси, між задачами. Третє правило - це регулярне оновлення мережевого графіка під час виконання проекту для врахування змін в плані.

Щоб бути ефективним використанням мережевого графіка в проектному менеджменті, необхідно також звернути увагу на деякі звички, що допоможуть уникнути помилок та зберегти час. Перш за все, необхідно дотримуватися графіку та контролювати прогрес виконання проекту. Друге - це забезпечення взаємодії між всіма учасниками проекту та дотримання стандартів комунікації. Третє - це регулярний аналіз виконання проекту та вирішення проблем, що виникають.

Усі ці правила та звички допомагають ефективно використовувати мережевий графік в проектному менеджменті та досягати поставлених цілей.

Однак, варто зазначити, що успіх використання мережевого графіка залежить від точності визначення часу виконання кожної задачі та її залежностей від інших задач. Неправильне визначення цих параметрів може призвести до неправильного планування та затримок в проекті. Тому, перед використанням мережевого графіка, необхідно провести детальний аналіз проекту та визначити всі можливі ризики та фактори, які можуть впливати на виконання проекту. Спочатку був розроблений мережевий графік робіт над новою мережею (див. рис. 2.6).

Загалом, використання мережевого графіка в проектному менеджменті є важливим елементом планування та контролю проекту. Цей інструмент дозволяє менеджерам проектів ефективно використовувати ресурси та зменшити ризики, пов'язані з проектом. Однак, для успішного використання мережевого графіка, необхідно правильно визначити послідовність виконання задач, розподілити ресурси та дотримуватися правил та звичок, пов'язаних з його використанням.

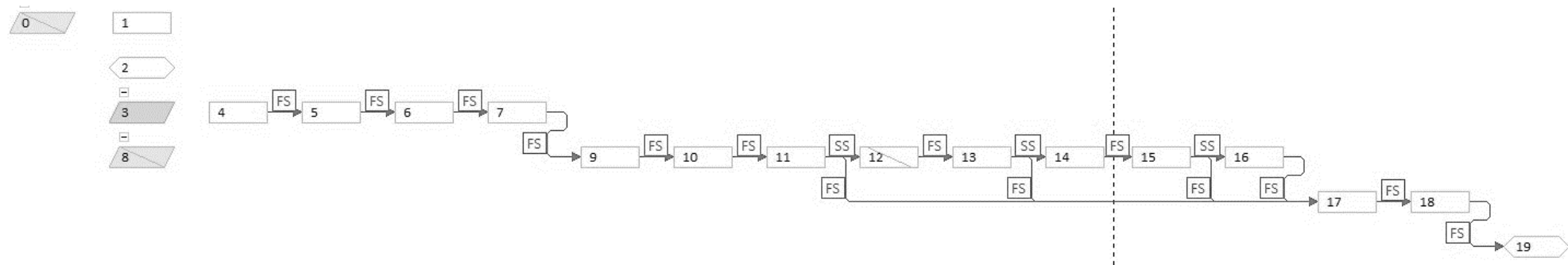


Рисунок 2.6 - Мережевий графік робіт над новою мережею

Джерело: запропоновано автором

Далі був створений мережевий графік робіт над розробкою програмного забезпечення. (див. рис. 2.7)

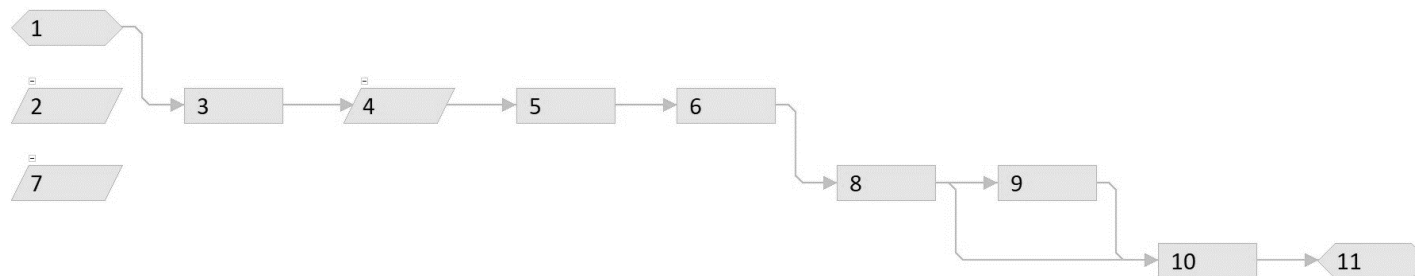


Рисунок 2.7 - Мережевий графік робіт над розробкою програмного забезпечення

Джерело: запропоновано автором

Розробка діаграми Ганта. Задача діаграми Ганта полягає в тому, щоб показати послідовність виконання завдань і проектних етапів у вигляді графіка. Це може бути корисно при плануванні проектів і візуалізації взаємозв'язків між задачами та їх взаємодією в часі. Діаграма Ганта дозволяє керівникам проектів, а також всім, хто працює над проектом, краще розуміти, які задачі є найважливішими, як довго вони займуть, і коли вони повинні бути виконані [7].

Діаграма Ганта може бути корисна при плануванні різних типів проектів, включаючи будівництво, програмне забезпечення та управління проектами. Вона дозволяє керівникам проектів краще контролювати процес виконання проекту та виявляти потенційні проблеми з дотриманням термінів виконання задач.

Для того, щоб використання діаграми Ганта було найбільш ефективним, потрібно дотримуватися певних правил. Спочатку необхідно визначити всі задачі, що повинні бути виконані, та їх тривалість. Потім необхідно визначити залежності між завданнями та встановити послідовність їх виконання. Після цього можна побудувати діаграму Ганта, яка допоможе візуалізувати процес виконання проекту та відстежити прогрес.

Мережевий графік а також діаграма Ганта були створені за допомогою програмного засобу Microsoft Project.

Microsoft Project має широкий набір функцій, які допомагають користувачам створювати графіки Ганта, розклади завдань, таблиці ресурсів, діаграми залежностей та інші елементи проектного планування. Користувачі можуть встановлювати залежності між завданнями, розподіляти ресурси, встановлювати терміни, присвоювати пріоритети і виконувати інші дії, щоб управляти проектом ефективно.

Програма надає можливість стежити за виконанням проекту в реальному часі, відслідковувати прогрес завдань, контролювати витрати та бюджет, вирішувати конфлікти ресурсів та здійснювати інші аналітичні дії. Користувачі також можуть створювати звіти та ділитися ними з іншими учасниками проекту.

Microsoft Project підтримує інтеграцію з іншими продуктами Microsoft, такими як Microsoft Excel, Microsoft Teams та Microsoft SharePoint, що дозволяє спрощувати спілкування та співпрацювати з іншими учасниками проєкту.

За допомогою створених діаграм Ганта вдалося відстужити критичний шлях проєкту, встановити всі зв'язки між запланованими роботами і оптимізувати час виконання а також фінансові ресурси проєкту.

2.3 Розробка віртуальної моделі локальної мережі.

Підприємство використовує застарілі методи захисту інформації у локальній мережі. Щоб це виправити, необхідно використовувати більш сучасний комутатор який здатен не тільки швидко обробляти інформацію але й використовувати сучасні та надійні методи кодування інформації.

Наявне обладнання не дозволяє швидко реагувати системному адміністратору на несанкціонований доступ до мережі, або навпаки проникнення сторонніми особам до інформації в середині мережі через заражені вірусом файли скачані з інтернету або активовані з зовнішніх USB або будь-яких інших накопичувачів. Для протидії цьому системному адміністратору необхідний швидкий доступ до управління а також обміном інформацією з будь-яким абонентом мережі.

Проаналізовано наявний стан мережі, зроблено наступні висновки:

По-перше, як і всі мережі, ця залежність від місцевого електроживлення. Якщо мережа стає знеструмленою, то вона не працездатна.

По-друге, послідовне підключення обладнання, яке в будь-який момент може вийти із ладу.

По-третє, складність розширення системи, додаткові витрати на розширення. Мережа має мало додаткових резервних каналів для підключення нових абонентів.

Щоб виправити дані недоліки необхідно виконати:

а) спланувати та розмістити резервну систему живлення мережі;

б) позбавитися від послідовного підключення обладнання. Можливість підключення нового обладнання, яке буде працювати в майбутньому, і не буде доставляти незручності користувачам;

в) побороти складність розширення системи, врегулювати додаткові витрати на розширення. Зробити додаткові резервні канали для підключення абонентів.

На підприємстві було поставлено задачу розробити нову мережу з використанням мінімальних грошових ресурсів а також з урахуванням попередніх недоліків мережі. Спочатку була розроблена діаграма робіт над локальною мережею (див. рис. 2.10).

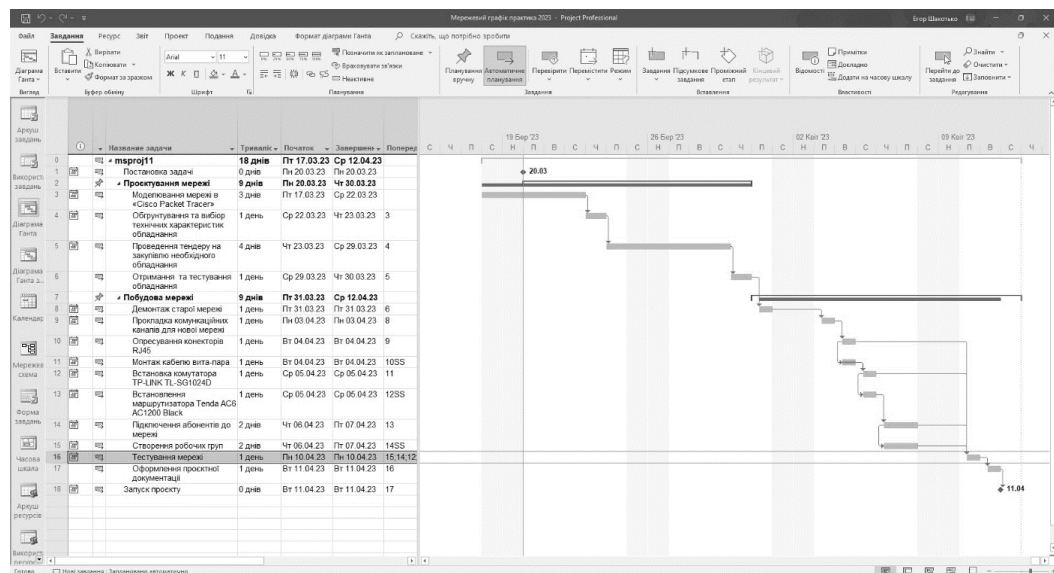


Рисунок 2.10 - Роботи впровадження локальної мережі

Джерело: запропоновано автором

Далі до проекту були додані робітники(див. рис. 2.11) а також задані ресурси (див. рис. 2.12).

msproj11	40 днів	Пт 17.03.23	Пт 12.05.23			15 550,00р.
Постановка задачі	0 днів	Пн 20.03.23	Пн 20.03.23			0,00р.
Проектування мережі	9 днів	Пн 20.03.23	Чт 30.03.23			4 000,00р.
Моделювання мережі в «Cisco Packet Tracer»	3 днів	Пт 17.03.23	Ср 22.03.23		Системний адміністратор	1 200,00р.
Обґрунтування та вибір технічних характеристик обладнання	1 день	Ср 22.03.23	Чт 23.03.23	3	Системний адміністратор Фінанси	400,00р.
Проведення тендеру на закупівлю необхідного обладнання	4 днів	Чт 23.03.23	Ср 29.03.23	4	Системний адміністратор Фінанси	1 600,00р.
Отримання та тестування обладнання	1 день	Ср 29.03.23	Чт 30.03.23	5	Системний адміністратор	800,00р.
Побудова мережі	31 днів	Пт 31.03.23	Пт 12.05.23			11 550,00р.
Демонтаж старої мережі	1 день	Пт 31.03.23	Пт 31.03.23	6	Технік-монтажник	400,00р.
Прокладка комунікаційних каналів для нової мережі	1 день	Пн 03.04.23	Пн 03.04.23	8	Технік-монтажник	400,00р.
Опресування конекторів RJ45	1 день	Вт 04.04.23	Вт 04.04.23	9	Технік-монтажник	400,00р.
Монтаж кабелю вита-пара	1 день	Вт 04.04.23	Вт 04.04.23	10SS	Технік-монтажник	400,00р.
Встановка комутатора TP-LINK TL-SG1024D	1 день	Ср 05.04.23	Ср 05.04.23	11	Технік-монтажник	400,00р.
Встановлення маршрутизатора Tenda AC6 AC1200 Black	1 день	Ср 05.04.23	Ср 05.04.23	12SS	Технік-монтажник	400,00р.
Підключення абонентів до мережі	2 днів	Чт 06.04.23	Пт 07.04.23	13	Технік-монтажник	800,00р.
Створення робочих груп	11 днів	Чт 06.04.23	Чт 20.04.23	14SS	Системний адміністратор	2 350,00р.
Тестування мережі	14 днів	Пт 21.04.23	Ср 10.05.23	15;14;12;10	Системний адміністратор	5 600,00р.
Оформлення проектної документації	1 день	Чт 11.05.23	Чт 11.05.23	16	Системний адміністратор	400,00р.
Запуск проєкту	0 днів	Чт 11.05.23	Чт 11.05.23	17		0,00р.

Рисунок 2.11 – Перелік завдань, виконавців та ресурсів проєкту

Джерело: запропоновано автором

Ім'я ресурсу	Тип	Одиниця вимірювання матеріалів	Ініціали	Група	Ма оди	Звич. ставка	Понад. ставка	Витрати	Нараху	Основний календар
Технік-монтаж	Робота		Т		200%	50,00р./год	0,00р./год	0,00р.	Пропорц	Стандартний
Мережеве обладнання	Матеріал		М			0,00р.		0,00р.	Пропорц	
Фінанси	Витрати		Ф						Пропорц	
Системний адміністратор	Робота		С		100%	50,00р./год	0,00р./год	0,00р.	Пропорц	Стандартний

Рисунок 2.12 – Аркуш заданих ресурсів проєкту

Джерело: запропоновано автором

Нова мережа була розроблена за допомогою програми «Cisco Packet Tracer». Cisco Packet Tracer - це симулятор мережевих пристроїв, розроблений компанією Cisco Systems. Він призначений для вивчення та експериментів з мережевими технологіями без реального обладнання.

Packet Tracer надає користувачам можливість створювати віртуальні мережі, включаючи роутери, комутатори, кінцеві пристрої та інші мережеві компоненти. Користувачі можуть підключати ці пристрої, налаштовувати їх, створювати

мережеві топології, налаштовувати параметри мережі, встановлювати протоколи, налаштовувати безпеку та багато іншого.

Один з основних функціональних блоків Packet Tracer - це віртуальний робочий стіл, де користувачі можуть перетягувати й встановлювати мережеві пристрої з бібліотеки, налаштовувати їх параметри та встановлювати з'єднання. Крім того, програма має інструменти для моніторингу та аналізу мережевого трафіку, що дозволяє вивчати роботу мережі та виявляти проблеми.

Packet Tracer також підтримує симуляцію різних мережевих протоколів, таких як IP, DHCP, DNS, OSPF, EIGRP, VLAN та інші. Користувачі можуть спостерігати за протоколами, їхніми взаємодіями та впливом на мережу.

Важливо відзначити, що Packet Tracer призначений для навчання та експериментів, і його можливості можуть бути обмеженими порівняно з реальним обладнанням. Однак, він є корисним інструментом для вивчення мережевих технологій та практичного використання в академічних та навчальних середовищах.

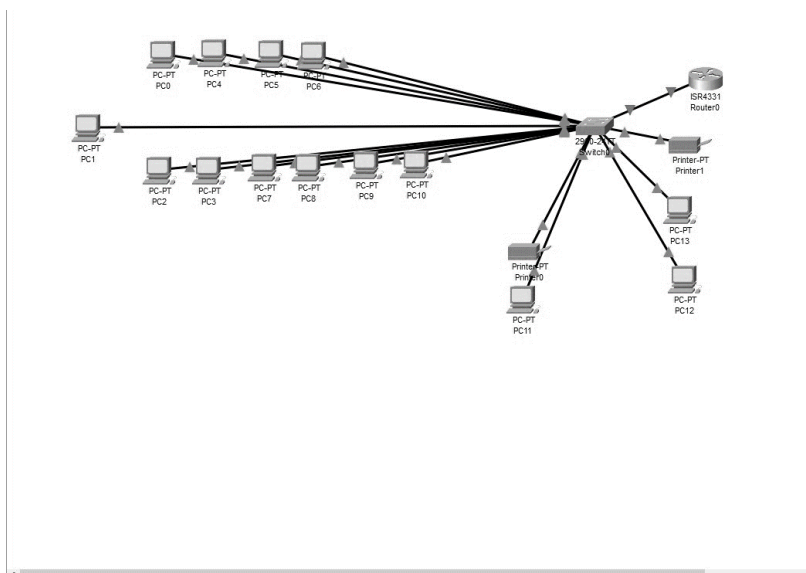


Рисунок 2.13 - Робоче вікно з розробленою мережею «Cisco Packet Tracer»

Нова мережа була протестована за допомогою програми «Cisco Packet Tracer». Всі абоненти мережі підключені до комутатора і без помилок передають дані між собою та мережевим принтером (див. рис. 2.13).

Для готової реалізації мережі пропонується виконати:

а) для початку, треба оновити апаратуру комп'ютерів, щоб вони відповідали вимогам підприємства. Для мінімального підняття ефективності працездатності абонентів і їх комфортнішої роботи пропоную встановити SSD. Пропонується вибір твердотільного накопичувача – Kingston SSDNow A400 (див. рис. 2.14, табл. 2.1);

Таблиця 2.1 - Характеристика SSD-накопичувача Kingston SSDNow A400

Параметр	Характеристика
Розробник	Kingston
Об'єм	240 Гб
Форм-фактор	2,5
Інтерфейс підключення	SATAIII

Джерело: запропоновано автором

б) Для збільшення швидкості передачі даних, пропоную замінити наявний маршрутизатор на сучасний, який здатний працювати зі стандартом Gigabit Ethernet, наприклад, такий як Tenda AC6 AC1200 Black (див. рис. 2.15), що має два діапазони роботи Wi-Fi, і підтримує Gigabit Ethernet це дозволить мережі стати більш стійкою до потенційних навантажень і забезпечить більшу середню швидкість передачі на одного абонента.

Таблиця 2.2 - Характеристика маршрутизатора Tenda AC6 AC1200 Black

Параметр	Характеристика
Розробник	Tenda
Тип антени	Зовнішня/незнімна
Частота роботи Wi-Fi	2,4 ГГц, 5 ГГц
Тип IEEE 802.11	a (54 Мбіт/с), b (11 Мбіт/с), g (54 Мбіт/с), n (300 Мбіт/с), ac (867 Мбіт/с)
Інтерфейс	RJ45-LAN
Кількість LAN-портів	3

Джерело: запропоновано автором

Також необхідно встановити більш сучасний комутатор, який буде використовувати різні методи шифрування та кодування інформації. Пропоную взяти TP-LINK TL-SG1024D (див. рис. 2.16).



Рисунок 2.15 - маршрутизатор Tenda AC6 AC1200 Black

Джерело: [13]



Рисунок 2.16 - комутатор «TP-LINK TL-SG1024D»

Джерело: [14]

Таблиця 2.3 - Характеристика комутатора TP-LINK TL-SG1024D

Параметр	Характеристика
Розробник	TP-LINK
Комутаційна спроможність	48 Гбіт/с
Швидкість передачі пакетів	35.7 млн. пакетів/с
Таблиця MAC адрес	8K
Буфер пам'яті пакетів	512 Kб
Стандарти та протоколи	IEEE 802.3i, IEEE 802.3u, IEEE 802.3ab, IEEE 802.3x

Джерело: запропоновано автором

в) Наступним кроком, пропоную підвищити цифрову безпеку організації, для цього, необхідно використовувати програмні та апаратні засоби захисту інформації. До апаратних можна віднести встановлення сигналізації, відеоспостереження, що дозволить негайно зреагувати на потенційне втручання до аудиторії. Говорячи про програмні засоби, то пропоную включити шифрування дисків за допомогою засобів Windows - Bitlocker, що дозволить запобігти втраті інформації з робочої станції у разі її викрадення; поставити на всі комп'ютери антивіруси, наприклад, Bitdefender, який має високий рівень по виявленню вірусів, різноманітні додаткові функції, і важливий фактор – майже не впливає на швидкість роботи системи; також пропоную встановити VPN, що дозволить би забезпечити безпеку підключення до глобальної мережі, паролей та конфіденційної інформації організації, наприклад SharkVPN.

г) Для подолання проблеми системи резервного живлення мережі, пропоную використати генератор, або мобільну батарею до якої доєднати апаратуру мережі а також ключових та необхідних абонентів. Пропоную використати мобільну батарею BLUETTI EB3A Portable Power Station (див. рис. 2.17) яка має значний обсяг батареї а також наскрізне живлення, що дає змогу під'єднати мобільну батареї до мережі і не вимикати від неї.

Таблиця 2.4 – Характеристика мобільної батареї BLUETTI EB3A

Параметр	Характеристика
Розробник	BLUETTI
Тип акумуляторів	Літій-залізо-фосфатні (LiFePO4)
Вихідні інтерфейси	1 x USB Type A DC-роз'єм USB Type C
Вихідна потужність (загальна), Вт	600
Функції зарядки	Power Delivery QC швидка зарядка 2.0 Qi бездротова зарядка
Особливості	Можливість наскрізної зарядки акумулятора

Джерело: запропоновано автором



Рисунок 2.17 - мобільна батарея «BLUETTI EB3A»

Джерело: [14]

При використанні всіх запропонованих апаратних та програмних засобів мережа подолає більшість недоліків. Швидкість та ефективність передачі даних в середині мережі збільшиться. З'явиться система автономного живлення а також фізичного захисту інформації та апаратури мережі.

РОЗДІЛ 3

РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ

3.1 Програмна реалізація клієнтської служби

Спочатку був реалізований основний функціонал - можливість відстежувати підключення або відключення USB-накопичувача. Після детального аналізу різних методів вирішення поставленої задачі було обрано використання вбудованого в операційну систему Windows інструмента «Windows Management Instrumentation» (WMI). WMI дозволяє відстежувати події, що виникають під час змін у внутрішніх об'єктах[16].

У нашому випадку ми будемо відстежувати зміни у об'єкті Win32_USBHub. Параметри зазнають змін під час кожного підключення або відключення USB-пристрою[5].

Для відстежування змін WMI використовує клас ManagementEventWatcher. Для реалізації буде використовуватися середовище програмування Visual Studio а також мова програмування C#[9]. Visual Studio - це інтегроване середовище програмування, розроблене компанією Microsoft. Воно надає розширений набір інструментів для створення різноманітних програм, веб-сайтів, мобільних додатків та інших програмних рішень. Visual Studio підтримує багато мов програмування, включаючи C#, Visual Basic, C++, JavaScript, Python та багато інших. Воно має потужний редактор коду зі змінними кольоровими підсвічуваннями, автодоповненням, вбудованими довідниками та іншими функціями, що полегшують роботу з кодом. Середовище надає можливості для управління версіями за допомогою інтегрованої системи керування версіями, такої як Git, інтегрованого відлагоджувача для пошуку й усунення помилок у коді, а також інструментів для побудови, тестування і розгортання програмного забезпечення[11].

Visual Studio також підтримує розробку веб-додатків, включаючи підтримку ASP.NET та веб-фреймворків, таких як Angular і React. Воно надає інструменти для візуального редагування сторінок, каскадних таблиць стилів (CSS) та скриптів.

Крім того, Visual Studio має велику екосистему додатків та розширень, що дозволяють розширити його функціональність і пристосувати його до потреб користувача.

Для перевірки підключених та відключених USB-пристроїв до комп'ютера створимо власну Windows службу «USBMonitoringService».

Визначимо поетапні дії для початку відслідковування змін:

Відкриваємо Visual Studio та на етапі створення нового проєкту обираємо створення «(Служба Windows .Net Framework)»

У вікні «Service1.cs» необхідно додати всі необхідні імена полів за допомогою «Управління пакетів NuGet».

Ці рядки імпортують необхідні простори імен, які використовуються в коді (див. рис. 3.1).

```
using System;  
using System.IO;  
using System.ServiceProcess;  
using System.Management;  
using System.Diagnostics;
```

Рисунок 3.1 - Фрагмент програмного коду «Поля імен програмного коду»

Джерело: запропоновано автором

Цей код визначає простір імен USBMonitoringService і клас USBMonitoringService, який успадковує від ServiceBase. Це основний клас служби, де розміщені всі методи та функціональність, пов'язані з моніторингом USB-подій (див. рис. 3.2).

```
namespace USBMonitoringService
{
    Ссылка: 3
    public partial class USBMonitoringService : ServiceBase
    {
```

Рисунок 3.2 – Фрагмент програмного коду «Клас USBMonitoringService»

Джерело: запропоновано автором

У цих рядках визначаються константи для шляху до файлу журналу (LogFilePath), назви подієвого журналу (EventLogName) та джерела подій (EventLogSource). Також визначаються приватні змінні _logWriter, _usbEventWatcher та _eventLog, які використовуються для ведення журналу та моніторингу USB-подій (див. рис. 3.3).

```
private const string LogFilePath = @"D:\USBMonitoringService\usb_devices.txt";
private const string EventLogName = "USBMonitoringServiceLog";
private const string EventLogSource = "USBMonitoringService";

private StreamWriter _logWriter;
private ManagementEventWatcher _usbEventWatcher;
private EventLog _eventLog;
```

Рисунок 3.3 – Фрагмент програмного коду «Константи для шляху до файлу журналу»

Джерело: запропоновано автором

Цей конструктор встановлює назву служби (ServiceName) на "USBMonitoringService" і дозволяє службі бути зупиненою (CanStop = true) (див. рис. 3.4).

```
public USBMonitoringService()
{
    ServiceName = "USBMonitoringService";
    CanStop = true;
}
```

Рисунок 3.4 – Фрагмент програмного коду «Встановлення назви служби»

Джерело: запропоновано автором

Цей метод викликається при запуску служби. Він викликає методи `InitializeLogWriter()`, `InitializeEventLog()` і `StartUSBEventMonitoring()`, які виконують ініціалізацію файлу журналу, подієвого журналу та початок моніторингу USB-подій відповідно (див. рис. 3.5).

```
protected override void OnStart(string[] args)
{
    InitializeLogWriter();
    InitializeEventLog();
    StartUSBEventMonitoring();
}
```

Рисунок 3.5 – Фрагмент програмного коду «Методи `InitializeLogWriter()`, `InitializeEventLog()` і `StartUSBEventMonitoring()`»

Джерело: запропоновано автором

Цей метод викликається при зупинці служби. Він викликає методи `StopUSBEventMonitoring()` і `CloseLogWriter()`, які зупиняють моніторинг USB-подій і закривають файл журналу відповідно (див. рис. 3.6).

```
protected override void OnStop()
{
    StopUSBEventMonitoring();
    CloseLogWriter();
}
```

Рисунок 3.6 – Фрагмент програмного коду «Методи `StopUSBEventMonitoring()` і `CloseLogWriter()`»

Джерело: запропоновано автором

Цей метод перевіряє, чи існує файл журналу за заданим шляхом (`LogFilePath`). Якщо файл не існує, він створює новий файл за допомогою `File.Create(LogFilePath)` (див. рис. 3.7).

```
private void InitializeLogWriter()
{
    if (!File.Exists(LogFilePath))
    {
        File.Create(LogFilePath).Close();
    }
}
```

Рисунок 3.7 – Фрагмент програмного коду «Метод InitializeEventLog()»

Джерело: запропоновано автором

У цьому методі перевіряється, чи існує подієвий журнал з вказаним джерелом подій (EventLogSource). Якщо журнал не існує, він створюється за допомогою EventLog.CreateEventSource(EventLogSource, EventLogName). Потім створюється об'єкт EventLog з назвою журналу (EventLogName) і встановлюється джерело подій (EventLogSource) (див. рис. 3.8).

```
private void InitializeEventLog()
{
    if (!EventLog.SourceExists(EventLogSource))
    {
        EventLog.CreateEventSource(EventLogSource, EventLogName);
    }

    _eventLog = new EventLog(EventLogName);
    _eventLog.Source = EventLogSource;
}
```

Рисунок 3.8 – Фрагмент програмного коду «Метод перевірки, чи існує подієвий журнал з вказаним джерелом подій (EventLogSource)»

Джерело: запропоновано автором

У цьому методі створюється запит WQL для моніторингу подій зміни USB-пристроїв. Створюється об'єкт ManagementEventWatcher з цим запитом, а потім реєструється обробник подій USBEventArrived для подій, що прибули. Нарешті, моніторинг USB-подій розпочинається за допомогою методу Start() (див. рис. 3.9).

У цьому методі обробляються отримані події USB. Він отримує тип події, ім'я пристрою та час події. Потім він записує ці дані у файл журналу за допомогою StreamWriter. Також він записує ці дані у подієвий журнал Windows за допомогою

_eventLog.WriteEntry(). Обробка події включає створення запису в файлі журналу та запис події в журнал подій Windows (див. рис. 3.10).

```
private void StartUSBEventMonitoring()
{
    var query = new WqlEventQuery("SELECT * FROM Win32_DeviceChangeEvent WHERE EventType = 2 or EventType = 3");
    _usbEventWatcher = new ManagementEventWatcher(query);
    _usbEventWatcher.EventArrived += USBEventArrived;
    _usbEventWatcher.Start();
}
```

Рисунок 3.9 – Фрагмент програмного коду «Метод створення запиту WQL для моніторингу подій зміни USB-пристроїв»

Джерело: запропоновано автором

```
private void USBEventArrived(object sender, EventArrivedEventArgs e)
{
    var eventType = e.NewEvent.GetPropertyValue("EventType").ToString();
    var deviceName = e.NewEvent.GetPropertyValue("DeviceName").ToString();
    var timestamp = DateTime.Now;

    using (_logWriter = new StreamWriter(LogFilePath, true))
    {
        _logWriter.WriteLine($"USB Device Monitoring Log ({DateTime.Now})");
        _logWriter.WriteLine("-----");
        _logWriter.WriteLine($"Event Type: {eventType}");
        _logWriter.WriteLine($"Device Name: {deviceName}");
        _logWriter.WriteLine($"Timestamp: {timestamp}");
        _logWriter.WriteLine("-----");
    }

    _eventLog.WriteEntry($"USB Device Monitoring Log ({DateTime.Now})\nEvent Type: {eventType}\nDevice Name: {deviceName}");
}
```

Рисунок 3.10 – Фрагмент програмного коду «Метод обробки отриманих подій USB»

Джерело: запропоновано автором

Цей метод зупиняє моніторинг USB-подій, викликаючи метод Stop() для _usbEventWatcher. Також він викликає метод Dispose() для звільнення ресурсів, пов'язаних з _usbEventWatcher (див. рис. 3.11).

```
private void StopUSBEventMonitoring()
{
    _usbEventWatcher?.Stop();
    _usbEventWatcher?.Dispose();
}
```

Рисунок 3.11 – Фрагмент програмного коду «Метод зупинки моніторингу USB-подій»

Джерело: запропоновано автором

Цей метод закриває файл журналу, викликаючи метод `Close()` для `_logWriter`. Також він викликає метод `Dispose()` для звільнення ресурсів, пов'язаних з `_logWriter` (див. рис. 3.12).

```
private void CloseLogWriter()
{
    _logWriter?.Close();
    _logWriter?.Dispose();
}
```

Рисунок 3.12 – Фрагмент програмного коду «Метод закриття файлу журналу»

Джерело: запропоновано автором

Повний код програми наведено у додатку Б. Використовуючи всі ці методи ми отримали власну Windows службу «USBMonitoringService», яка перевіряє всі підключені або відключені USB-пристрої до комп'ютера під-час виконання служби. Всі дані записуються в окремий лог де їх перевіряє користувач інформаційної системи.

3.2 Програмна реалізація інтерфейсу користувача

Після створення служби «USBMonitoringService» необхідно розробити зручну та зрозумілу екранну форму для користувача інформаційної системи.

Цей код є частиною Windows Forms додатку і містить обробники подій для різних кнопок та методи для взаємодії зі службою Windows.

Цей код визначає простір імен `USBProgramWindowsFormsApp1`, а в ньому розміщений клас `Form1`, який є головною формою додатку. У конструкторі `Form1` створюється об'єкт `ServiceController`, який представляє службу Windows з назвою "USBMonitoringService" (див. рис. 3.13) .

```

namespace USBProgramWindowsFormsApp1
{
    Ссылка 3
    public partial class Form1 : Form
    {
        private ServiceController usbMonitoringService;
        Ссылка 1
        public Form1()
        {
            InitializeComponent();
            usbMonitoringService = new ServiceController("USBMonitoringService");
        }
    }
}

```

Рисунок 3.13 – Фрагмент програмного коду «Конструктор Form1»

Джерело: запропоновано автором

Цей метод виконується при натисканні на кнопку з button1. Він запускає процес командного рядка (CMD) і виконує команду InstallUtil.exe, яка встановлює службу Windows. Змінні installUtilPath та servicePath визначають шляхи до виконуваних файлів InstallUtil.exe та USBMonitoringService.exe відповідно. Рядок команди "{installUtilPath} \"{servicePath}\"" передає шлях до USBMonitoringService.exe у команду InstallUtil.exe. Результат виконання команди виводиться у вікно повідомлень (MessageBox). Після завершення роботи з процесом командного рядка, він закривається та звільняється від ресурсів (див. рис. 3.14).

```

private void button1_Click(object sender, EventArgs e)
{
    string installUtilPath = @"C:\Windows\Microsoft.NET\Framework64\v4.0.30319\InstallUtil.exe";

    string servicePath = @"C:\visualtyudio\USBMonitoringService\USBMonitoringService\bin\Debug\USBMonitoringService.exe";

    Process process = new Process();

    try
    {
        process.StartInfo.FileName = "cmd.exe";
        process.StartInfo.UseShellExecute = false;
        process.StartInfo.RedirectStandardInput = true;
        process.StartInfo.RedirectStandardOutput = true;
        process.StartInfo.CreateNoWindow = true;

        process.Start();
    }
}

```

Рисунок 3.14 – Фрагмент програмного коду «Кнопка button1»

Джерело: запропоновано автором

Цей метод виконується при натисканні на кнопку з `button3`. Він перевіряє статус служби `usbMonitoringService`. Якщо служба не зупинена, вона зупиняється за допомогою `usbMonitoringService.Stop()` (див. рис. 3.15).

Після зупинки служби виконується очікування її статусу за допомогою `usbMonitoringService.WaitForStatus(ServiceControllerStatus.Stopped)`. Відповідне повідомлення виводиться у вікні повідомлень (`MessageBox`).

```
private void button3_Click(object sender, EventArgs e)
{
    try
    {
        if (usbMonitoringService.Status != ServiceControllerStatus.Stopped)
        {
            usbMonitoringService.Stop();
            usbMonitoringService.WaitForStatus(ServiceControllerStatus.Stopped);
            MessageBox.Show("Службу успішно зупинено.");
        }
        else
        {
            MessageBox.Show("Службу успішно зупинено.");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Помилка при зупинці служби: " + ex.Message);
    }
}
```

Рисунок 3.15 – Фрагмент програмного коду «Кнопка `button3`»

Джерело: запропоновано автором

Цей метод виконується при натисканні на кнопку з `button2`. Він перевіряє статус служби `usbMonitoringService`. Якщо служба не запущена, вона запускається за допомогою `usbMonitoringService.Start()`.

Після запуску служби виконується очікування її статусу за допомогою `usbMonitoringService.WaitForStatus(ServiceControllerStatus.Running)`. Відповідне повідомлення виводиться у вікні повідомлень (`MessageBox`) (див. рис. 3.16).

```

private void button2_Click(object sender, EventArgs e)
{
    try
    {
        if (usbMonitoringService.Status != ServiceControllerStatus.Running)
        {
            usbMonitoringService.Start();
            usbMonitoringService.WaitForStatus(ServiceControllerStatus.Running);
            MessageBox.Show("Служба успішно запущена.");
        }
        else
        {
            MessageBox.Show("Служба успішно запущена.");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Помилка під час запуску служби: " + ex.Message);
    }
}

```

Рисунок 3.16 – Фрагмент програмного коду «Кнопка button2»

Джерело: запропоновано автором

Цей метод виконується при натисканні на кнопку з button4. Він відкриває файл за допомогою `Process.Start(filePath)`, де `filePath` - шлях до файлу. У випадку помилки відкриття файлу, виводиться відповідне повідомлення у вікні повідомлень (`MessageBox`) (див. рис. 3.17).

```

private void button4_Click(object sender, EventArgs e)
{
    string filePath = @"D:\USBMonitoringService\usb_devices.txt";

    try
    {
        Process.Start(filePath);
    }
    catch (Exception ex)
    {
        MessageBox.Show("Помилка при відкритті файлу: " + ex.Message);
    }
}

```

Рисунок 3.17 – Фрагмент програмного коду «Кнопка button4»

Джерело: запропоновано автором

Цей метод виконується при натисканні на кнопку з button5. Він створює процес командного рядка (CMD) і виконує команду `sc delete {serviceName}`, де

{serviceName} - назва служби, яку потрібно видалити. Результат виконання команди виводиться у вікно повідомлень (MessageBox). Після завершення роботи з процесом командного рядка, він закривається та звільняється від ресурсів (див. рис. 3.18).

```
private void button5_Click(object sender, EventArgs e)
{
    string serviceName = "USBMonitoringService";

    Process process = new Process();

    try
    {
        process.StartInfo.FileName = "cmd.exe";
        process.StartInfo.UseShellExecute = false;
        process.StartInfo.RedirectStandardInput = true;
        process.StartInfo.RedirectStandardOutput = true;
        process.StartInfo.CreateNoWindow = true;

        process.Start();

        process.StandardInput.WriteLine($"sc delete {serviceName}");
        process.StandardInput.WriteLine("exit");

        process.WaitForExit();

        string output = process.StandardOutput.ReadToEnd();

        if (output.Contains("Службу успішно видалено"))
        {
            MessageBox.Show("Службу успішно видалено");
        }
        else
        {
            MessageBox.Show("Службу успішно видалено");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Помилка: {ex.Message}");
    }
    finally
    {
        process.Close();
        process.Dispose();
    }
}
```

Рисунок 3.18 – Фрагмент програмного коду «Кнопка button5»

Джерело: запропоновано автором

Отже, готова Windows Form має наступний вигляд (див. рис. 3.19):

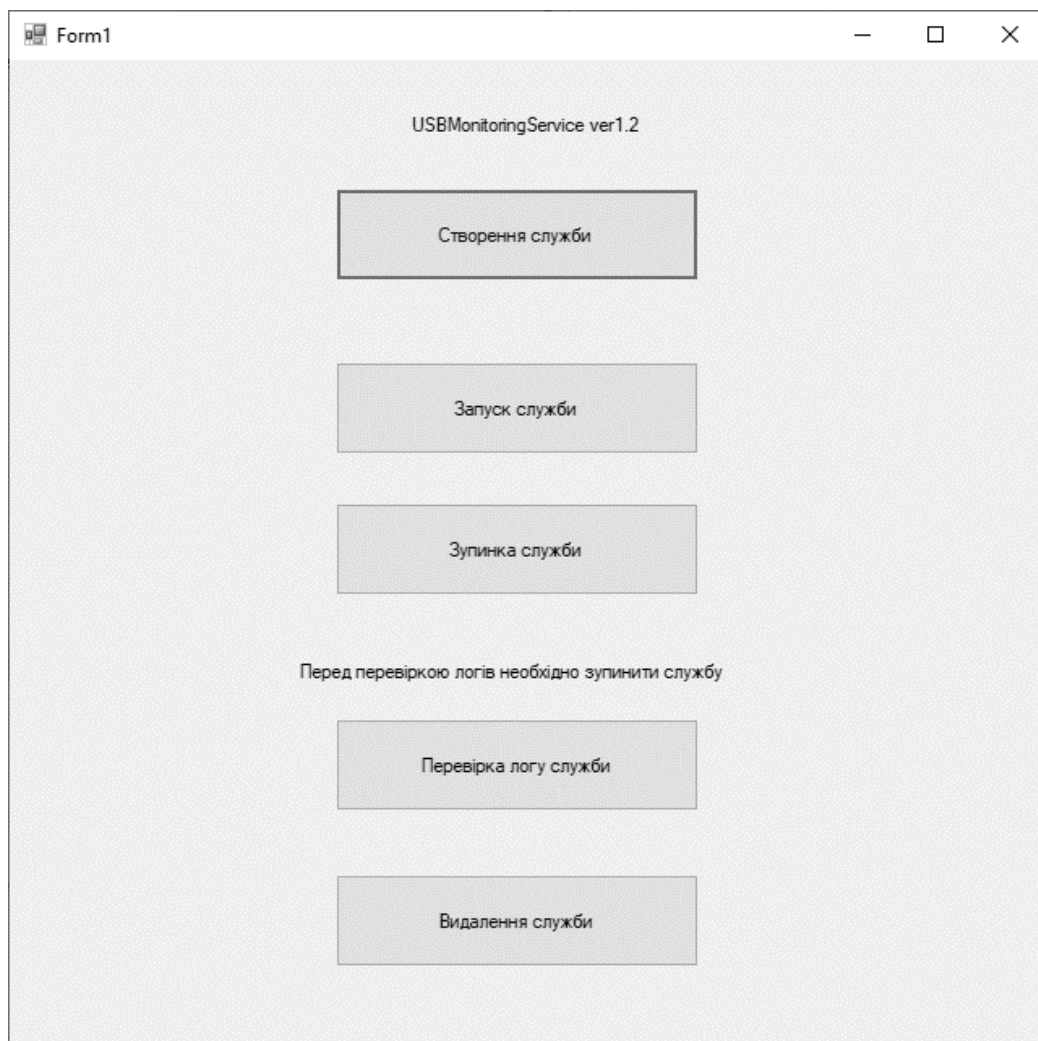


Рисунок 3.19 – Знімок екранної форми програми «USBMonitoringService»

Джерело: запропоновано автором

Повний код програми дивись у додатку В. Була розроблена зручна та зрозуміла екранна форма користувача, яка дозволяє: створити службу, запустити службу, зупинити службу, перевірити лог служби, видалити службу за необхідністю.

3.3 Використання програмного додатку

Розглянемо поетапно всі варіанти використання програмного додатку.

Запускаємо програму «USBMonitoringService.exe» від ім'я адміністратора.

У відкритій екранній формі обираємо кнопку «Створення служби», яка

автоматично створить службу «USBMonitoringService» на комп'ютері. Після цього в журналі «Служби» операційної системи з'явиться нова служба «USBMonitoringService»(див. рис. 3.20).

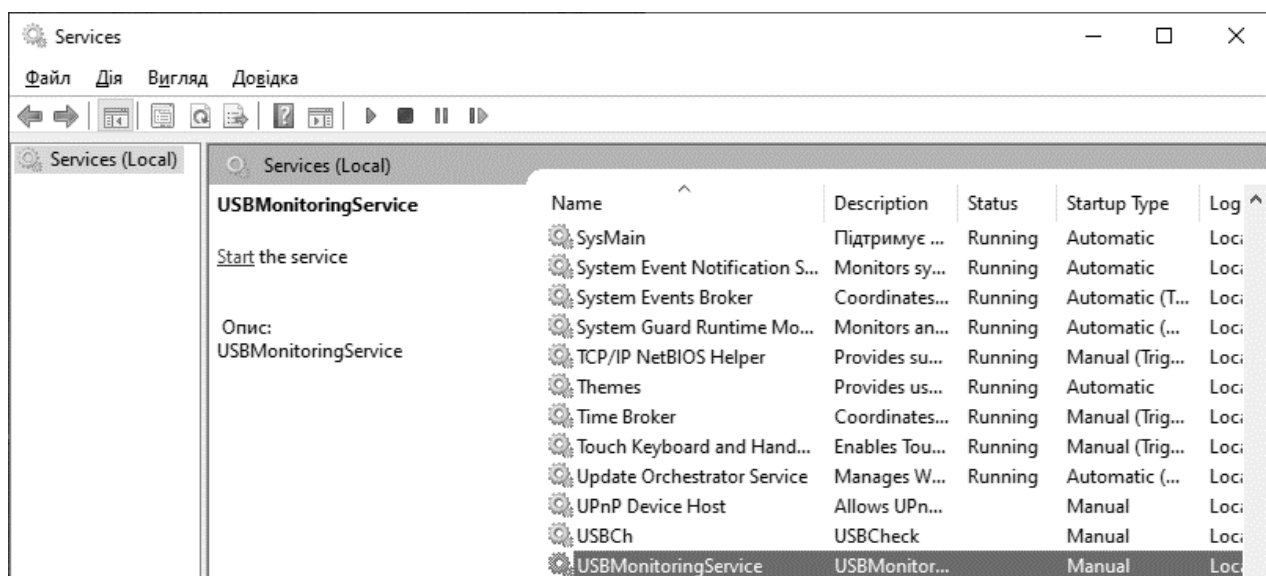


Рисунок 3.20 – Знімок журналу «Служби» після встановлення служби «USBMonitoringService»

Джерело: запропоновано автором

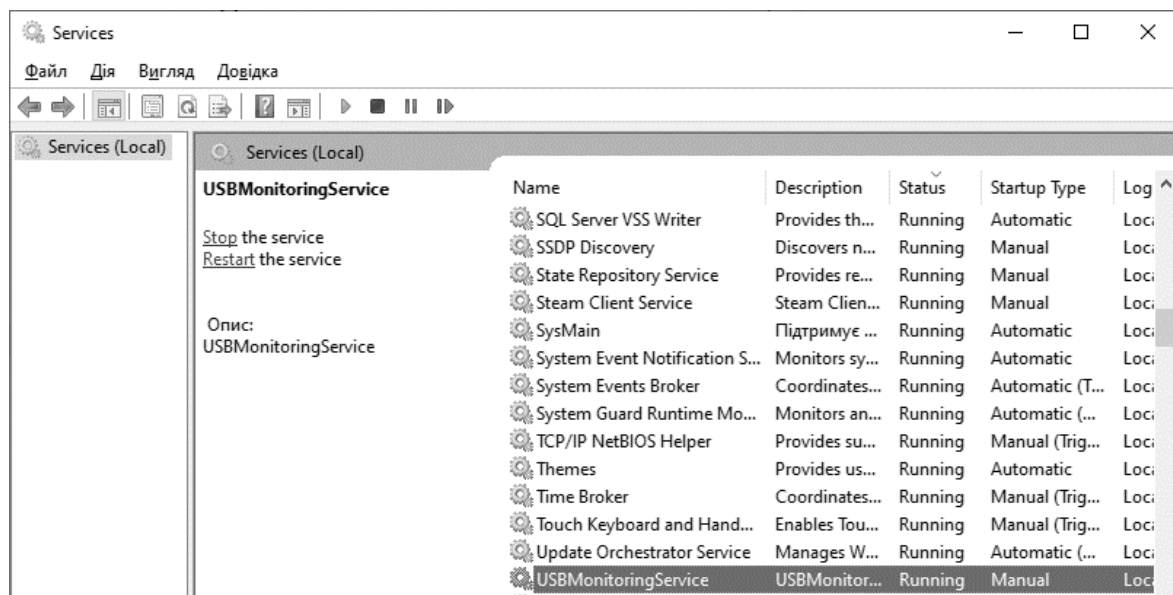


Рисунок 3.21 – Знімок журналу «Служби» після запуску служби «USBMonitoringService»

Джерело: запропоновано автором

Після успішного встановлення служби нам необхідно її запустити, для цього використовуючи відкриту програму натискаємо кнопку «Запуск служби». Після цього в журналі «Служби» операційної системи служба «USBMonitoringService» змінить статус на виконується (Running)(див. рис. 3.21).

Після успішного запуску служби вона буде реєструвати всі підключені та від'єднанні USB – пристрої. Коли користувач захоче перевірити записи служби, необхідно її зупинити, для цього використовуючи відкриту програму натискаємо кнопку «Зупинка служби». Якщо переглянути журнал «Служби» ми помітимо, що служба «USBMonitoringService» змінить статус(див. рис. 3.22).

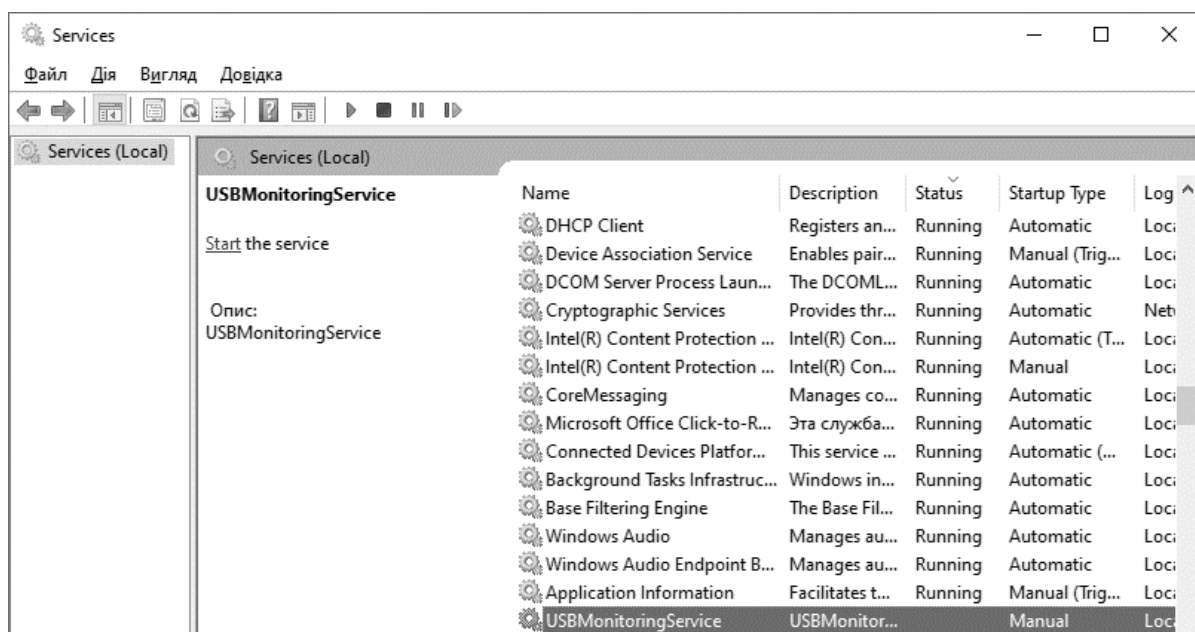


Рисунок 3.22 – Знімок журналу «Служби» після зупинки служби «USBMonitoringService»

Джерело: запропоновано автором

Далі користувач має змогу переглянути лог служби за допомогою кнопки «Перевірка логу служби». Після відкриття файлу з інформацією про всі підключені та від'єднанні USB – пристрої під час роботи служби «USBMonitoringService»(див. рис. 3.23).

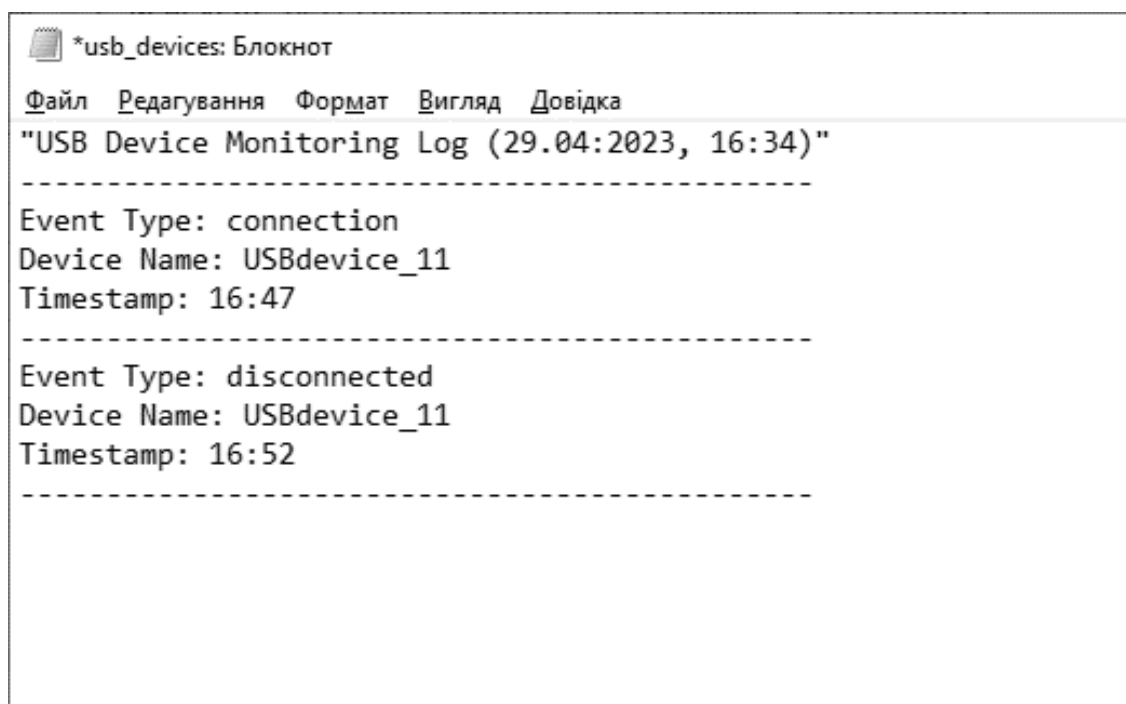


Рисунок 3.23 – Знімок логу служби «USBMonitoringService».

Джерело: запропоновано автором

Як бачимо з рисунку 3.23 під час роботи служби «USBMonitoringService» було виявлено дві дії з USB-пристроєм «USBdevice_11», а саме його підключення о 16:47 та від'єднанні о 16:52. Таким чином користувач інформаційної системи може зручно переглянути який і коли USB-пристрій був підключений до комп'ютера.

Далі користувач має змогу видалити службу з журналу «Служби» за допомогою кнопки «Видалення служби». Після служба «USBMonitoringService» автоматично видалиться з журналу.

Якщо перевірити журнал «Служби» помітимо що служби «USBMonitoringService» була видалена і більше не відображається у списку).

Таким чином будь-який програмного продукту може зручно відслідковувати всі підключені та від'єднанні USB – пристрої до комп'ютера, навіть якщо він сам не був присутнім за робочим місцем. Це дозволяє підвищити контроль за USB-пристроями що дає змогу швидко та оперативною реагувати на нові, невідомі та потенційно небезпечні USB-пристрої.

ВИСНОВКИ

В кваліфікаційній роботі виконано аналітичний огляд досліджень, що-до визначеної предметної області, проаналізовано сучасні рішення, сформульовані загальні підходи до розробки програмного забезпечення.

Створено проєкт модернізації наявної корпоративної мережі. До нього входять: діаграма використань, діаграма Ганта та мережевий графік всіх наявних робіт за допомогою програми Microsoft Project. Окремо було створено віртуальну модель нової корпоративної мережі за допомогою програми Cisco Packet Tracer. Наступним кроком було запропоновано нове фізичне обладнання для локальної мережі, за допомогою якого вона подолає більшість власних недоліків. Швидкість та ефективність передачі даних в середині мережі збільшиться. З'явиться система автономного живлення а також фізичного захисту інформації та апаратури мережі.

Створено програмний застосунок для відстежування підключення або відключення USB-накопичувача на комп'ютері абонента локальної мережі за допомогою середовища програмування Visual Studio Code використовуючи вбудованого в операційну систему Windows інструмента «Windows Management Instrumentation» (WMI). Для відстежування змін WMI був використаний клас Management Event Watcher. Реалізований код програмного застосунку на мові програмування C#. Програма проводить моніторинг підключених USB – пристроїв за допомогою власної Windows служби. Всі дані про підключені або від'єднані USB – пристрої, під час роботи служби, заносяться до окремого текстового файлу. Для швидкого та зручного користування службою була розроблена екранна форма за допомогою вбудованого в Visual Studio Code додатку Windows Forms.

Використовуючи всі удосконалення мережі а також розроблену інформаційну систему, організація зможе підвищити рівень швидкості та ефективності передачі інформації, її захист а також контроль підключення USB – пристроїв у локальній мережі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Задерейко О. В. Комп'ютерні мережі: навчальний посібник / О. В. Задерейко, Н. І. Логінова, А. А. Толокнов. – Одеса, 2022. 249 с.
2. Захист комп'ютерних мереж URL – <https://magistrweb.wordpress.com/home-2/protection/>. (Дата звернення 23.01.2023)
3. Кіщак І.Т., Шебаніна О.В., Ключник А.В., Федосова А.О. Управління проектами: навчально-методичний посібник для самостійного вивчення дисципліни. – Миколаїв: 2018. 352 с.
4. Матеріали науково-практичної конференції викладачів, аспірантів та студентів Сумського НАУ (25-28 квітня 2023 р.). – Суми, 2023. 642 с.
5. Система моніторингу підключення USB-пристроїв для проведення аудиту кібербезпеки URL – https://essuir.sumdu.edu.ua/bitstream-download/123456789/79077/1/Tatarinov_bac_rob.pdf;jsessionid=4A9954949266E2050EAFE999CED94C22. (Дата звернення 12.02.2023)
6. Технології XXI сторіччя: Збірник тез за матеріалами 28-ої міжнародної науковопрактичної конференції (23-25 листопада 2022 р.). Ч.2. – Суми: СНАУ, 2022. 193 с.
7. Управління ІТ-проектами. [Книга 1. Стандарти, моделі та методи управління проектами] – Львів: «Новий Світ- 2000». 2017. 550 с.
8. About WMI URL – <https://docs.microsoft.com/en-us/windows/win32/wmisdk/about-wmi>. (Дата звернення 22.04.2023)
9. Faisal P. Create A Windows Service In C# URL - <https://www.csharpcorner.com/article/create-windows-services-in-c-sharp/>.
10. GRIDINSOFT URL – <https://gridinsoft.ua/ddos>. (Дата звернення 27.01.2023)
11. ManagementEventWatcher Class URL – <https://docs.microsoft.com/enus/dotnet/api/system.management.managementeventwatcher?view=dotnet-plat-ext3.1>. (Дата звернення 17.03.2023)

- | | | | |
|---|--------------------------------|-----|---|
| 12. | Rozetka | URL | — |
| https://hard.rozetka.com.ua/ua/kingston_sa400s37_240g/p14606570/?gclid=CjwKCAjwyqWkBhBMEiwAp2yUFvBOlR73KtSsV-Oc2mH6EeOLtG12zSKK8agwZVaW2knkanknJzW_9hoCIxQQA vD_BwE . (Дата звернення 09.03.2023) | | | |
| 13. | Rozetka | URL | — |
| https://rozetka.com.ua/ua/12834243/p12834243/?gclid=CjwKCAjwyqWkBhBMEiwAp2yUFqX4ceJO1MLoywF4yZRE9cMkDHhbFHqiMEqpFRLZK80gSDX8e4ACFBoCtSkQAvD_BwE . (Дата звернення 10.03.2023) | | | |
| 14. | Rozetka | URL | — |
| https://rozetka.com.ua/ua/tp_link_tl_sg1024d/p203087/?gclid=CjwKCAjwyqWkBhBMEiwAp2yUFnn2rVZ32js4tanXsT_XRH-K73q9BnH2uJT6lQuKRSX9KrzLGFUMHxoCDBsQAvD_BwE . (Дата звернення 11.03.2023) | | | |
| 15. | Rozetka | URL | — |
| https://rozetka.com.ua/ua/364980675/p364980675/?gclid=CjwKCAjwyqWkBhBMEiwAp2yUFoOoaiaX_6-efvfeG5ke66ZBq_H8nPQob3cVRE8MWYAKj4ohtjB3_RoCWnUQAvD_BwE . (Дата звернення 12.03.2023) | | | |
| 16. | WMI for Detection and Response | URL | - |
| https://www.hsd1.org/?abstract&did=795007 . | | | |

ДОДАТКИ

Додаток А

ТЕХНІЧНЕ ЗАВДАННЯ
на розробку інформаційної системи
«Програмний-додаток організації “ Глухівський агротехнічний фаховий
коледж Сумського НАУ ”»

ПОГОДЖЕНО:

доцент кафедри кібернетики та інформатики

Здобувач вищої освіти групи ІСТ _____

_____ (ПІБ)

Суми 2023

1. Призначення й мета створення програмного додатку

1.1 Призначення програмного додатку

Програмний додаток для перевірки підключень USB-пристроїв до абонентів корпоративної мережі.

1.2 Мета створення програмного додатку

Розробка системи моніторингу підключень USB-пристроїв для невеликих організацій, які не мають можливості придбати дорогі системи з непотрібними для цієї задачі функціями.

1.3 Цільова аудиторія

Системний-адміністратор мережі організації.

2 Вимоги до програмного додатку

2.1 Вимоги до програмного додатку в цілому

2.1.1 Вимоги до структури й функціонування програмного додатку

Програмний-додаток має бути створений на мові програмування C# у середовищі програмування Visual Studio.

2.1.2 Вимоги до персоналу

Від персоналу не має вимагатися особливих технічних навичок для підтримки й експлуатації програмного додатку, окрім загальних навичок роботи з персональним комп'ютером.

2.1.3 Вимоги до збереження інформації

Уся інформація надана у програмного додатку буде зберігатися у окремому текстовому файлі.

2.1.4 Вимоги до розмежування доступу

Розроблюваний програмний додаток має бути доступний тільки системному-адміністратору.

2.2 Структура програмного додатку

2.2.1 Загальна інформація про структуру програмного додатку

Структура програмного додатку являє собою набір кнопок навігації.

Існують кнопки:

Створення служби - автоматично створить службу «USBMonitoringService» на комп'ютер.

Запуск служби – після використання кнопки в журналі «Служби» операційної системи служба «USBMonitoringService» змінить статус на виконується (Running).

Зупинка служби - служба «USBMonitoringService» змінить статус.

Перевірка логу служби - відкриється файл з інформацією про всі підключені та від'єднанні USB – пристрої під час роботи служби «USBMonitoringService».

Видалення служби - служба «USBMonitoringService» автоматично видалиться з журналу.

2.2.2 Навігація

Відповідно до бажаного замовником дизайну програмного додатку, для навігації, було створено просте і зрозуміле меню. Меню необхідне для швидкого переміщення користувача по усім доступним функціям програми.

2.2.3 Наповнення програмного додатку (контент)

Для моніторингу USB-пристроїв у локальній було обрано використання вбудованого в операційну систему Windows інструмента «Windows Management Instrumentation» (WMI).

Для створення екранної форми меню була обрана Windows Forms.

Вся інформація про підключені та від'єднанні USB-пристрої вноситься до окремого текстового документа.

2.2.4 Дизайн та структура додатку

Стиль програмного додатку має бути простим, зрозумілим для сприйняття.

Основою мають бути кнопки керування програмою, програмний-додаток має бути інтуїтивно зрозумілим для використання.

2.3 Вимоги до функціонування системи

2.3.1 Потреби користувача

Потреби користувача, визначені на основі рішення замовника, представлені у таблиці А.1.

Таблиця А.1 – Потреби користувача

ID	Потреби користувача	Джерело
UN-01	Запуск служби	Системний Адміністратор
UN-02	Зупинка служби	Системний Адміністратор
UN-03	Перегляд логу служби	Системний Адміністратор
UN-04	Видалення служби	Системний Адміністратор

2.3.2 Функціональні вимоги

На основі потреб користувача були визначені такі функціональні вимоги:

- моніторинг USB-пристроїв;
- запуск служби;
- видалення служби;
- встановлення служби;
- перевірка логу служби.

Наводяться усі функціональні вимоги.

2.3.3 Системні вимоги

Даний розділ визначає, розподіляє та вказує на системні вимоги, визначені розробником. Їх перелік наведений в таблиці А.2.

Таблиця А.2 – Системні вимоги

ID	Системні вимоги	Пріоритет	Опис
SR-01	Наявність модуля запису USB-пристроїв	M	Надає можливість клієнту перевірити лог програми
SR-02	Встановлення служби	M	Встановлює службу на локальний комп'ютер
SR-03	Автоматичний запуск служби	S	Служба автоматично запускається на комп'ютері під час його включення
SR-04	Видалення служби	M	Служба видаляється при використанні програми
SR-05	База даних з інформацією про USB-пристроїв	C	Відповідає за заповнення бази даних програмного додатку новими записами

Умовні позначення в таблиці А.2:

Must have (M) – вимоги, які повинні бути реалізовані в системі;

Should have (S) – вимоги, які мають бути виконані, але вони можуть почекати своєї черги;

Could have (C) – вимоги, які можуть бути реалізовані, але вони не є центральною ціллю проекту.

2.4 Вимоги до видів забезпечення

1.1.1 Вимоги до інформаційного забезпечення

Реалізація програмного додатку відбувається з використанням: Windows 11\10\7, C#, Visual Studio 2022.

2.4.2 Вимоги до лінгвістичного забезпечення

Програмний додаток має бути виконаний українською мовою.

2.4.3 Вимоги до програмного забезпечення

Програмне забезпечення клієнтської частини повинне задовольняти наступним вимогам: програмний додаток має працювати на операційних системах Windows 11\10\7

3 Склад і зміст робіт зі створення web-додатку

Докладний опис етапів роботи зі створення web-додатку наведено в таблиці А.3.

Таблиця А.3 – Етапи створення web-додатку

№	Склад і зміст робіт	Строк розробки (у робочих днях)
1	Постановка цілей необхідних для досягнення певного результату	1 день
2	Складання технічного завдання	3 дні
3	Створення основного коду програмного додатку	5 дні
4	Створення макету дизайну програмного додатку	1 день
5	Перевірка працездатності програмного додатку	1 день
6	Завершення роботи	1 день
	Загальна тривалість робіт	12 днів

Додаток Б

Код USBMonitoringService

Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.ServiceProcess;
using System.Text;
using System.Threading.Tasks;

namespace USBMonitoringService
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        static void Main()
        {
            ServiceBase[] ServicesToRun;
            ServicesToRun = new ServiceBase[]
            {
                new USBMonitoringService()
            };
            ServiceBase.Run(ServicesToRun);
        }
    }
}
```

Service1.cs

```
using System;
using System.IO;
using System.ServiceProcess;
using System.Management;
using System.Diagnostics;

namespace USBMonitoringService
{
    public partial class USBMonitoringService : ServiceBase
    {
        private const string LogFilePath = @"D:\USBMonitoringService\usb_devices.txt";
        private const string EventLogName = "USBMonitoringServiceLog";
        private const string EventLogSource = "USBMonitoringService";

        private StreamWriter _logWriter;
        private ManagementEventWatcher _usbEventWatcher;
        private EventLog _eventLog;

        public USBMonitoringService()
        {
            ServiceName = "USBMonitoringService";
            CanStop = true;
        }
        protected override void OnStart(string[] args)
        {
            InitializeLogWriter();
            InitializeEventLog();
        }
    }
}
```

```

        StartUSBEventMonitoring();
    }
    protected override void OnStop()
    {
        StopUSBEventMonitoring();
        CloseLogWriter();
    }

    private void InitializeLogWriter()
    {
        if (!File.Exists(LogFilePath))
        {
            File.Create(LogFilePath).Close();
        }
    }
    private void InitializeEventLog()
    {
        if (!EventLog.SourceExists(EventLogSource))
        {
            EventLog.CreateEventSource(EventLogSource, EventLogName);
        }
        _eventLog = new EventLog(EventLogName);
        _eventLog.Source = EventLogSource;
    }
    private void StartUSBEventMonitoring()
    {
        var query = new WqlEventQuery("SELECT * FROM Win32_DeviceChangeEvent WHERE EventType = 2 or EventType = 3");
        _usbEventWatcher = new ManagementEventWatcher(query);
        _usbEventWatcher.EventArrived += USBEventArrived;
        _usbEventWatcher.Start();
    }
    private void USBEventArrived(object sender, EventArrivedEventArgs e)
    {
        var eventType = e.NewEvent.GetPropertyValue("EventType").ToString();
        var deviceName = e.NewEvent.GetPropertyValue("DeviceName").ToString();
        var timestamp = DateTime.Now;

        using (_logWriter = new StreamWriter(LogFilePath, true))
        {
            _logWriter.WriteLine($"USB Device Monitoring Log ({DateTime.Now})");
            _logWriter.WriteLine("-----");
            _logWriter.WriteLine($"Event Type: {eventType}");
            _logWriter.WriteLine($"Device Name: {deviceName}");
            _logWriter.WriteLine($"Timestamp: {timestamp}");
            _logWriter.WriteLine("-----");
        }
        _eventLog.WriteEntry($"USB Device Monitoring Log ({DateTime.Now})\nEvent Type: {eventType}\nDevice Name: {deviceName}\nTimestamp: {timestamp}", EventLogEntryType.Information);
    }
    private void StopUSBEventMonitoring()
    {
        _usbEventWatcher?.Stop();
        _usbEventWatcher?.Dispose();
    }

    private void CloseLogWriter()
    {
        _logWriter?.Close();
        _logWriter?.Dispose();
    }
}

```

```
}
```

Додаток В

Програмний код інтерфейсу користувача

Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace USBProgramWindowsFormsApp1
{
    internal static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Form1.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.ServiceProcess;

namespace USBProgramWindowsFormsApp1
{
    public partial class Form1 : Form
    {
        private ServiceController usbMonitoringService;
        public Form1()
        {
            InitializeComponent();
            usbMonitoringService = new ServiceController("USBMonitoringService");
        }
    }
}
```

```

private void label1_Click(object sender, EventArgs e)
{

}

private void button1_Click(object sender, EventArgs e)
{

    string installUtilPath = @"C:\Windows\Microsoft.NET\Framework64\v4.0.30319\InstallUtil.exe";

    string servicePath =
@"C:\visualtyudio\USBMonitoringService\USBMonitoringService\bin\Debug\USBMonitoringService.exe";

    Process process = new Process();

    try
    {
        process.StartInfo.FileName = "cmd.exe";
        process.StartInfo.UseShellExecute = false;
        process.StartInfo.RedirectStandardInput = true;
        process.StartInfo.RedirectStandardOutput = true;
        process.StartInfo.CreateNoWindow = true;

        process.Start();

        process.StandardInput.WriteLine($"{installUtilPath} \"{servicePath}\"");
        process.StandardInput.WriteLine("exit");

        process.WaitForExit();

        string output = process.StandardOutput.ReadToEnd();

        if (output.Contains("Встановлення служби успішно завершено"))
        {
            MessageBox.Show("Встановлення служби успішно завершено");
        }
        else
        {
            MessageBox.Show("Встановлення служби успішно завершено");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Помилка: {ex.Message}");
    }
    finally
    {
        process.Close();
        process.Dispose();
    }
}

private void button3_Click(object sender, EventArgs e)
{
    try
    {

```

```

        if (usbMonitoringService.Status != ServiceControllerStatus.Stopped)
        {

            usbMonitoringService.Stop();
            usbMonitoringService.WaitForStatus(ServiceControllerStatus.Stopped);
            MessageBox.Show("Службу успішно зупинено.");
        }
        else
        {
            MessageBox.Show("Службу успішно зупинено.");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Помилка при зупинці служби: " + ex.Message);
    }
}

```

```

private void button2_Click(object sender, EventArgs e)
{
    try
    {

        if (usbMonitoringService.Status != ServiceControllerStatus.Running)
        {

            usbMonitoringService.Start();
            usbMonitoringService.WaitForStatus(ServiceControllerStatus.Running);
            MessageBox.Show("Служба успішно запущена.");
        }
        else
        {
            MessageBox.Show("Служба успішно запущена.");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Помилка під час запуску служби: " + ex.Message);
    }
}

```

```

private void button4_Click(object sender, EventArgs e)
{
    string filePath = @"D:\USBMonitoringService\usb_devices.txt";

    try
    {
        Process.Start(filePath);
    }
    catch (Exception ex)
    {
        MessageBox.Show("Помилка при відкритті файлу: " + ex.Message);
    }
}

```

```

private void button5_Click(object sender, EventArgs e)
{
    string serviceName = "USBMonitoringService";

```

```

Process process = new Process();

try
{
    process.StartInfo.FileName = "cmd.exe";
    process.StartInfo.UseShellExecute = false;
    process.StartInfo.RedirectStandardInput = true;
    process.StartInfo.RedirectStandardOutput = true;
    process.StartInfo.CreateNoWindow = true;

    process.Start();

    process.StandardInput.WriteLine($"sc delete {serviceName}");
    process.StandardInput.WriteLine("exit");

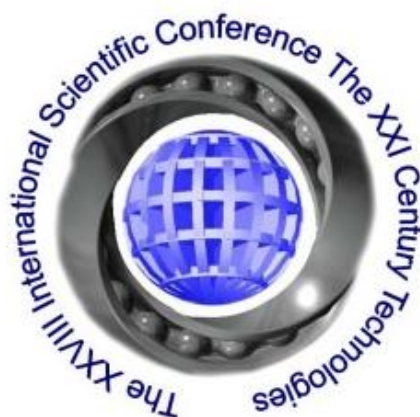
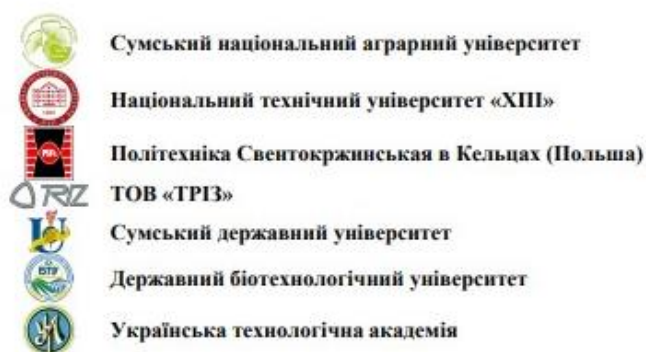
    process.WaitForExit();

    string output = process.StandardOutput.ReadToEnd();

    if (output.Contains("Службу успішно видалено"))
    {
        MessageBox.Show("Службу успішно видалено");
    }
    else
    {
        MessageBox.Show("Службу успішно видалено");
    }
}
catch (Exception ex)
{
    MessageBox.Show($"Помилка: {ex.Message}");
}
finally
{
    process.Close();
    process.Dispose();
}
}
}

```

Додаток Г



ТЕХНОЛОГІЇ ХХІ СТОРІЧЧЯ

Збірник тез за матеріалами 28^{ої} міжнародної науково-
практичної конференції
(23-25 листопада 2022 р.)

Частина 2

Секції: «Прогресивні технології в економіці»,
«Прогресивні технології в екології та енергоефективність»,
«Прогресивні технології у педагогіці»

Суми – 2022

ЗМІСТ

СЕКЦІЯ «ПРОГРЕСИВНІ ТЕХНОЛОГІЇ В ЕКОНОМІЦІ»

<i>Самошкіна І.Д., доцент, СНАУ, м. Суми, Україна</i> МІЖНАРОДНИЙ ДОСВІД РОЗВИТКУ АГРАРНОГО СТРАХУВАННЯ В УКРАЇНІ	3
<i>Омельчук А.І., магістрант, Агаджанова С.В., к.т.н., доцент, Суми, СНАУ, Україна</i> ЗАКОНОДАВЧІ ЗАСАДИ ФУНКЦІОНУВАННЯ СИСТЕМИ НАДАННЯ СОЦІАЛЬНИХ ПОСЛУГ ДЛЯ СОЦІАЛЬНИХ ГРУП НАСЕЛЕННЯ.....	5
<i>Омельчук А.І., магістрант, Агаджанова С.В., к.т.н., доцент, Суми, СНАУ, Україна</i> ВДОСКОНАЛЕННЯ СИСТЕМИ НАДАННЯ СОЦІАЛЬНИХ ПОСЛУГ СПЕЦІАЛЬНИМ ГРУПАМ НАСЕЛЕННЯ ЗАСОБАМИ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ.....	6
<i>Басанець Д.В., студент, СНАУ, Суми, Україна</i> ІНФОРМАЦІЙНА ПІДСИСТЕМА ОБЛІКУ КАДРОВОГО ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ КОЛЕДЖУ ВСП «ГАФК СНАУ».....	7
<i>Нечипоренко В.В., к.с.-г.н., доцент, СНАУ</i> СТРАТЕГІЧНІ НАПРЯМИ СТАЛОГО РОЗВИТКУ АГРАРНОГО СЕКТОРУ ЕКОНОМІКИ	8
<i>Шакотько А. М., студент, Толбатов А.В., к.т.н., доцент, СНАУ, Суми, Україна</i> ІНФОРМАЦІЙНА ПІДСИСТЕМА МАТЕРІАЛЬНОГО ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ВСП «ГАФК СНАУ»	9
<i>Шакотько Є.С., молодший спеціаліст, Сумський національний аграрний університет</i> СУЧАСНІ НАПРЯМКИ СТРАХУВАННЯ КІБЕРРИЗИКІВ	11
<i>Білик Г.М., магістрант, Сумський національний аграрний університет, Україна</i> ОСОБЛИВОСТІ ЗЕМЛЕКОРИСТУВАННЯ У РОЗВИНЕНИХ КРАЇНАХ СВІТУ В ЧАСТИНІ ЕКСПЛУАТАЦІЇ СІЛЬСЬКОГОСПОДАРСЬКИХ УГІДЬ	12
<i>Грошовик Р.Ю. студент, Пасько Н.Б., доцент, Сумський НАУ</i> АВТОМАТИЗАЦІЯ ПОГОСПОДАРСЬКОГО ОБЛІКУ ТЕРИТОРІАЛЬНИХ ГРОМАД	15
<i>Кальченко Я., магістрант, СНАУ, Суми, Україна</i> ОЦІНКА ЕФЕКТИВНОСТІ ЗБУТОВОЇ ДІЯЛЬНОСТІ АГРАРНОГО ПІДПРИЄМСТВА	19
<i>Колеснікова Ю.С., магістрант, Сумський національний аграрний університет, Україна</i> ОСОБЛИВОСТІ ФОРМУВАННЯ МАРКЕТИНГОВОЇ СТРАТЕГІЇ ПЕРСОНАЛУ У ВЕТЕРИНАРНИХ КЛІНІКАХ	21
<i>Колосов Р.В., магістрант, Сумський національний аграрний університет, Україна</i> ПРОБЛЕМИ ПРОГНОЗУВАННЯ ВИКОРИСТАННЯ ЗЕМЕЛЬНИХ РЕСУРСІВ В УКРАЇНІ	24
<i>Колпаков В.В., магістрант, Сумський національний аграрний університет, Україна</i> МАРКЕТИНГ ВІДНОСИН ЯК ВАЖЛИВИЙ ЕЛЕМЕНТ КОНКУРЕНТНОЇ МАРКЕТИНГОВОЇ СТРАТЕГІЇ	26
<i>Лисенко К., магістрант, Сумський національний аграрний університет, Україна</i> МЕТОДОЛОГІЯ ТА АЛГОРИТМ ПРОВЕДЕННЯ ЗЕМЕЛЬНИХ ТОРГІВ	29
<i>Макарова В.В., д.е.н., доцент, Сумський національний аграрний університет, Україна</i> МЕТОДОЛОГІЧНІ ПІДХОДИ ДО ОЦІНКИ ЕФЕКТИВНОСТІ МАРКЕТИНГОВОЇ ДІЯЛЬНОСТІ В СОЦІАЛЬНИХ МЕРЕЖАХ.....	31
<i>Могилка А., магістрант, Сумський національний аграрний університет, Україна</i> ЕКОЛОГО-ЕКОНОМІЧНІ ТЕНДЕНЦІЇ ВИКОРИСТАННЯ	

УДК 004.49:368.8

Шакотько Є.С., молодший спеціаліст, Сумський національний аграрний університет

СУЧАСНІ НАПРЯМКИ СТРАХУВАННЯ КІБЕРРИЗИКІВ

Вступ. Враховуючи поширення цифрових послуг – відповідальність організацій які зберігають, обробляють та використовують персональні дані значно збільшується. Зростання числа кібератак, фішингових програм та несанкціонованого доступу до даних, а також супутній ризик шкоди для репутації, штрафів та втрати інтелектуальної власності змушує багато організацій почати звертатися до страхування кіберризиків (СК), щоб захистити себе від фінансових збитків, пов'язаних з атакою.

Процедура «кіберстрахування» для України є доволі новим та мало дослідженим та попри це сьогодні воно набуває актуальності, бо українські підприємства потребують захисту від кібератак. Причини, через що розвиток СК в Україні є низьким: неякісні ІТ-інфраструктури; неліцензійне програмне забезпечення; важкість доведення що підприємство постраждало від кібератаки; високовартість СК, бо страховики не мають чіткої статистики кібератак і не можуть правильно розрахувати страхові тарифи. Навіть за наявності покриття у організації, що зазнала атаки, може виявитися, що через недогляд у їх безпеці вони отримують відмову в оплаті страхування. Покладатися виключно на СК для захисту інформаційної системи організації, даних клієнтів та суспільної репутації – ризикована ставка.

Виклад основного матеріалу дослідження. Впроваджуючи різні стратегії кібербезпеки, фахівці підприємства можуть підвищити рівень організації щодо відповідності вимогам страховиків, при цьому знизивши ймовірність того, що їм доведеться звернутися за цим покриттям.

Швидке зростання числа кібератак, спрямованих проти компаній практично будь-якого розміру та галузі призводить до фінансових, репутаційних та конфіденціальних втрат. Проблема дотримання вимог СК полягає в тому, що успішно порушуються правила кібербезпеки і це призводить до зупинення діяльності компанії, втрати репутації та значних штрафів на додаток до фінансових втрат які безпосередньо пов'язані з атакою. СК створено для компенсування організаціям фінансових втрат у разі кібератаки чи витоку даних.

Втрата СК, зазвичай, виникає через наступні проблеми: системні пошкодження; переривання бізнесу; втрата конфіденційності; фінансові штрафи; договірні порушення; відновлення даних.

Однак, щоб претендувати на покриття, організації повинні відповідати списку вимог, який зростає. Ставки, термін зберігання та інші заходи контролю залежать від профілю ризику компанії. Навіть, коли організація має право на отримання полісу і купує його, якщо їй не вдасться забезпечити надійний захист, її постачальник може відмовитися покривати претензію коли станеться інцидент.

Якщо організація не дотримується вимог кібербезпеки, визначених страховою компанією, її страховий поліс може опинитися під загрозою. Крім майже повсюдного попиту на багатфакторну автентифікацію (MFA), право на покриття та оплату часто включає такі вимоги:

1. Резервне копіювання та аварійне відновлення.

2. Виявлення шкідливої активності в кінцевих точок (EDR). Встановіть антивірусні рішення для захисту кінцевих точок від несанкціонованих програм, вірусів та інших атак.
3. Управління ідентифікацією та аутентифікацією (IAM). Провести Авторизацію та аутентифікацію користувачів, а також дотримуйтесь політики мінімальних привілеїв.
4. Управління привілейованим доступом (PAM). Слідкуйте за привілейованими обліковими записами, щоб виявити підозрілу поведінку та швидко виявляти скомпрометовані облікові записи.
5. Управління патчами. Постійно впроваджуйте виправлення та оновлення.

Висновок. Багато організацій звертаються до закордонних страхових компаній через те, що сучасні напрямки страхування кіберризиків в Україні не є сформованим та потребують подальшої розробки та впровадження. Але щоб отримати СК треба дотримуватися суворих вимог кібербезпеки постачальника страхових послуг. Тільки тоді організації можуть претендувати на отримання СК або скоротити витрати на улагодження проблем після кібератаки і залишатися захищеними у разі витоку даних.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ

МАТЕРІАЛИ

науково-практичної конференції
викладачів, аспірантів та студентів
Сумського НАУ

(25-28 квітня 2023 р.)

Рудакова Д.О. РОЛЬ ЕКОЛОГІЧНОГО МАРКЕТИНГУ В ФОРМУВАННІ КОНКУРЕНТНИХ ПЕРЕВАГ ДЕРЖАВНОГО ПІДПРИЄМСТВА	339
Базарний С.О. УПРАВЛІННЯ ТОВАРНОЮ ПОЛІТИКОЮ НА ПІДПРИЄМСТВАХ МАЛОГО БІЗНЕСУ	340
Сергієнко В.Г. КАДРОВА БЕЗПЕКА ЯК ОСНОВНИЙ ЕЛЕМЕНТ ЕКОНОМІЧНОЇ БЕЗПЕКИ ПІДПРИЄМСТВА	341
Бельська А.О. РЕКЛАМНА ДІЯЛЬНІСТЬ НА ПІДПРИЄМСТВАХ РОЗДРІБНОЇ ТОРГІВЛІ ТА СПОСОБИ ЇЇ ВДОСКОНАЛЕННЯ	342
Дундук Д.С. РОЛЬ І ЗНАЧЕННЯ КРЕДИТУ В РОЗВИТКУ ЕКОНОМІКИ УКРАЇНИ	343
В'юненко О.Б. СУЧАСНІ ПРОБЛЕМИ ІНФОРМАЦІЙНОЇ КІБЕРГІПЕНІ	344
Баталова А.Б. ГЕОІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ЯК ІНСТРУМЕНТ ЕКОЛОГІЧНИХ ДОСЛІДЖЕНЬ	345
Маяк А.О. ЦИФРОВІ ІНСТРУМЕНТИ ДЛЯ ПУБЛІКАЦІЇ ТА ПРОДАЖУ МУЗИКИ	346
Бусенко І.В. РОЗРОБКА ВІДЕОІГОР НА БАЗІ ТЕХНОЛОГІЇ P2E	347
Крамський А.С. ChatGPT – ТВІЙ МЕНТОР У НАВЧАННІ	348
Гервольський М.Д. АНАЛІТИЧНИЙ АНАЛІЗ РІЗНИХ ВИДІ ХОСТИНГІВ	349
Потапенко М.Т. АВТОМАТИЗОВАНА ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ СПОРТИВНОГО КОМПЛЕКСУ	350
Єременко О.В. РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ПЕРЕВІРКИ ПРАКТИЧНИХ РОБІТ ІЗ ВИБІРКИ ДАНИХ: ШЛЯХИ І ТЕХНОЛОГІЇ	351
Мотлюк М.В. А/В ТЕСТУВАННЯ ЯК МЕТОД МАРКЕТИНГОВИХ ДОСЛІДЖЕНЬ В ІТ	352
Олефіренко Д.В. ПРОГНОЗУВАННЯ КІЛЬКОСТІ ВЕЛИКОЇ РОГАТОЇ ХУДОБИ В УКРАЇНІ З ВИКОРИСТАННЯМ ЕКОНОМЕТРИЧНИХ МЕТОДІВ	353
Бабуров Д.В. ПРОГНОЗУВАННЯ КІЛЬКОСТІ СІЛЬСЬКОГОСПОДАРСЬКОЇ ПТИЦІ В УКРАЇНІ З ВИКОРИСТАННЯМ ЕКОНОМЕТРИЧНИХ МЕТОДІВ	354
Шакотько А.М. ІНФОРМАЦІЙНА ПІДСИСТЕМА МАТЕРІАЛЬНОГО ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ВСП «ГАФК СНАУ»	355
Басанець Д.В. ІНФОРМАЦІЙНА ПІДСИСТЕМА ОБЛІКУ КАДРОВОГО ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ КОЛЕДЖУ ВСП «ГАФК СНАУ»	356
Хіманова А.О. ПРОБЛЕМИ УПРАВЛІННЯ ІТ ПРОЕКТАМИ СУЧАСНИХ НАВЧАЛЬНИХ СИСТЕМ	357
Сіденко А.С. ПРОБЛЕМИ ПОБУДОВИ САЙТІВ ДЛЯ СИСТЕМ ДИСТАНЦІЙНОГО НАВЧАННЯ	358
Шакотько Є.С. СУЧАСНІ АЛГОРИТМИ АНАЛІЗУ ПРОДУКТИВНОСТІ ТА МОНІТОРИНГУ КОМП'ЮТЕРНИХ МЕРЕЖ	359
Грошовик Р.Ю. ОБГРУНТУВАННЯ НЕОБХІДНОСТІ АВТОМАТИЗАЦІЇ ВЕДЕННЯ ПОГОСПОДАРСЬКОГО ОБЛІКУ ТЕРИТОРІАЛЬНИХ ГРОМАД	360
Наконечний О.Б. АНАЛІЗ ПРОБЛЕМИ АВТОМАТИЗАЦІЇ РОЗРАХУНКУ БАЛАНСУ ГУМУСУ І ПОЖИВНИХ РЕЧОВИН В ҐРУНТІ ТА ПРОПОЗИЦІЇ ЩОДО ЇЇ ВИРІШЕННЯ	361
Мартинченко С.В. ОБРОБКА ВЕЛИКИХ ДАНИХ: ПРОБЛЕМИ І ШЛЯХИ ЇХ ВИРІШЕННЯ	362
Zhang Yuanyu THE KEY FACTORS IN PERSONAL DEVELOPMENT FROM THE PERSPECTIVE OF ENTERPRISES	363
Qin Fan IMPROVEMENT OF THE RISK MANAGEMENT SYSTEM AT THE ENTERPRISE	364
Jiang Maoji ORGANIZE THE CONCEPT AND METHOD OF BUSINESS PROCESS IMPROVEMENT	365
Zhang Buyao DISCUSSION ON THE MANAGEMENT OF CAREER PROMOTION OF ENTERPRISE EMPLOYEES	366
Cui Xiaoping THE IMPACT OF INNOVATIVE TRANSFORMATIONS IN THE AGRO-INDUSTRIAL SPHERE ON THE ACHIEVEMENT OF THE SUSTAINABLE DEVELOPMENT GOALS	367
Zhao Jing CONCEPTUAL APPROACHES TO MANAGING THE BEHAVIOR OF THE COMPANY'S PERSONNEL ON THE BASIS OF CORPORATE CULTURE	368
Гулько Б. В. МЕХАНІЗМ ДЕРЖАВНОГО РЕГУЛЮВАННЯ В СФЕРІ СТАЛОГО РОЗВИТКУ	369
Маслак О.М., Санін Р.С. ІНСТРУМЕНТИ ДЕРЖАВНОГО РЕГУЛЮВАННЯ ФЕРМЕРСЬКИХ ГОСПОДАРСТВ УКРАЇНИ В УМОВАХ ВОЄННОГО СТАНУ	370
Ніколаєнко Т.С. НАПРЯМИ РОЗВИТКУ ПУБЛІЧНОГО АДМІНІСТРУВАННЯ ОСВІТИ НА РІВНІ ТЕРИТОРІАЛЬНОЇ ГРОМАДИ	371
Гребченко Т.А., Стеценко Б.М. КОМУНІКАЦІЙНИЙ ЗВ'ЯЗОК ОРГАНІВ ДЕРЖАВНОГО УПРАВЛІННЯ ІЗ СУСПІЛЬСТВОМ	372
Майданевич С.В. АДМІНІСТРАТИВНО-ПРОЦЕДУРНІ ЗАСАДИ РЕГУЛЮВАННЯ В БАНКІВСЬКІЙ СФЕРІ	373
Chonglong Bi MANAGEMENT OF ORGANIZATIONAL CULTURE AS A FACTOR OF INCREASING COMPETITIVENESS	374
Тимошенко М.І. ПРИНЦИПИ СТВОРЕННЯ ТА УПРАВЛІННЯ КЛАСТЕРНИМИ ТА НАДКЛАСТЕРНИМИ ЛІКАРНЯМИ	375

СУЧАСНІ АЛГОРИТМИ АНАЛІЗУ ПРОДУКТИВНОСТІ ТА МОНІТОРИНГУ КОМП'ЮТЕРНИХ МЕРЕЖ

Шакотько Є. С., студ. 2 курсу ФЕІМ, спец. «Інформаційні системи та технології»
 Науковий керівник: к.е.н., доц. В'юненко О.Б.
 Сумський НАУ

Алгоритми продуктивності та моніторингу комп'ютерних мереж - це набір програмних засобів, які дозволяють контролювати роботу комп'ютерної мережі та підтримувати її продуктивність на високому рівні. Застосування таких алгоритмів є дуже важливим для будь-якої компанії, яка має велику мережу комп'ютерів. Це допомагає зменшити час, витрачений на пошук та усунення неполадок у мережі, а також збільшити її продуктивність.

Одним з найбільш популярних алгоритмів продуктивності є алгоритм WMI (Windows Management Instrumentation). Цей алгоритм дозволяє отримувати інформацію про стан комп'ютерної мережі та її компонентів, таких як процесори, оперативна пам'ять, диски, мережеві інтерфейси тощо. WMI дозволяє збирати цю інформацію на регулярній основі та використовувати її для аналізу продуктивності мережі. WMI найчастіше використовується для автоматизації адміністративної роботи та отримання доступу до даних без безпосередньої роботи з операційною системою.

Іншим популярним алгоритмом продуктивності є SNMP (Simple Network Management Protocol). Цей алгоритм дозволяє отримувати інформацію про стан мережевих пристроїв, таких як маршрутизатори, комутатори та інші. SNMP дозволяє збирати цю інформацію та використовувати її для моніторингу стану мережі, а також для виявлення проблем та їх подальшого усунення. Загалом, Ммодель SNMP складається з чотирьох ключових компонентів: а) керовані пристрої; б) станції управління (NMS); в) керуюча інформація; г) протокол управління.

Керований пристрій є мережевим вузлом, який реалізує SNMP-інтерфейс, який дозволяє односторонній (тільки для читання) або двосторонній (читання і запис) доступ до інформації, специфічної для вузла. Керовані пристрої обмінюються вузловими даними з NMS. Станція управління мережею виконує додатки, які контролюють і управляють керованими пристроями. NMS забезпечують основну частину ресурсів обробки і пам'яті, необхідних для управління мережею. Одна або кілька NMS можуть існувати в будь-якій керованій мережі.

Крім алгоритмів продуктивності, також існують алгоритми моніторингу мережі, які дозволяють відстежувати роботу окремих комп'ютерів та програм на них. Наприклад, алгоритм моніторингу з використанням NetFlow. Цей алгоритм дозволяє вимірювати трафік мережі та збирати статистику про використання різних мережевих пристроїв. За допомогою NetFlow можна визначити, які додатки та протоколи використовуються в мережі, а також з'ясувати, які з них відносяться до критичних та можуть впливати на продуктивність мережі. Функції та призначення NetFlow: а) моніторинг мережі; б) моніторинг програмного забезпечення; в) моніторинг користувачів; г) аналіз безпеки мережі.

- Моніторинг мережі. Дані протоколу NetFlow дозволяють виконувати моніторинг стану мережі. За допомогою програм, які виконують обробку накопичених даних від NetFlow колекторів, можливо проводити оцінку трафіку при цьому виявляти проблеми, що виникають в мережі.

- Моніторинг програмного забезпечення. Дані протоколу NetFlow дозволяють мережним адміністраторам мати точну інформацію про програми які на даний час працюють або використовуються в мережі, виявляти з них ті, що створюють проблеми в роботі мережі або перенавантажують їх.

- Моніторинг користувачів. Дані протоколу NetFlow дозволяють збирати детальну інформацію про кожного користувача мережі, про те які мережні ресурси він використовує на даний момент, яке програмне забезпечення, з якими IP – адресами працює та інше.

- Аналіз безпеки мережі. Під час використання протоколу NetFlow можливо ідентифікувати та класифікувати атаки по відмові в обслуговуванні (DDoS атаки), виявляти трафік, що генерується вірусами та троянськими програмами у реальному часі.

Залежно від розміру мережі та потреб в моніторингу, можна використовувати як безкоштовні, так і платні алгоритми продуктивності та моніторингу мережі. Наприклад, безкоштовний алгоритм Wireshark є дуже популярним серед мережевих інженерів та адміністраторів, але не завжди може забезпечити повний моніторинг мережі.

Платні алгоритми, такі як SolarWinds Network Performance Monitor або PRTG Network Monitor, мають більш широкий функціонал та можливості, але вимагають великої фінансової витрати. Тому перед вибором алгоритму слід ретельно оцінити всі переваги та недоліки кожного з них.

Важливою проблемою є розробка критеріїв вибору алгоритмів аналізу продуктивності та моніторингу комп'ютерних мереж, які дозволяють контролювати роботу мережі та вчасно виявляти можливі проблеми з продуктивністю. Кожен з них має свої особливості та переваги, тому перед вибором алгоритму варто звернути увагу на конкретні потреби та завдання компанії. Аналіз сучасних алгоритмів продуктивності та моніторингу комп'ютерних мереж є актуальною задачею для будь-якої компанії, яка використовує складну комп'ютерну мережу. Правильний вибір алгоритму може допомогти ефективно контролювати роботу мережі та вчасно виявляти можливі проблеми з продуктивністю.