Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

# КВАЛІФІКАЦІЙНА РОБОТА

*бакалавр*
Освітній рівень

Система забезпечення функціонування конвергентної обчислювальної мережі підприємства на основі використання «хмарних» технологій
Назва теми

*КвРКІ. 180224.18.02.01 ПЗ*
Шифр

Галузь знань *12 «Інформаційні технології»*
Шифр, назва

Спеціальність *123 «Комп'ютерна інженерія»*
Шифр, назва

Освітня програма *«Комп'ютерна інженерія»*
Назва

Виконав: студент IV курсу, група КІ-18-2 _____ Бойко В.О.
Підпис

Керівник доц., к. т. н, доцент кафедри Кб _____ Орленко В.С.
Підпис, дата

Нормоконтролер ст. викладач кафедри Кб 08.06.22 Мостовий С.В.
Підпис, дата

До захисту допускаю:

Зав. кафедри Кб, к.т.н., доц _____ Кльоц Ю.П.
Підпис

«08» 06 2022р.

Хмельницький, 2022

| Формат | Зона | Позиц | Позначення | Найменування | Кільк. | Прим. |
|---|---|---|---|---|---|---|
| | | | | | | |
| А4 | | 1 | | Завдання на кваліфікаційну роботу | 1 | |
| | | | | | | |
| А4 | | 2 | | Анотація | 1 | |
| | | | | | | |
| А4 | | 3 | КвРКІ. 180224.18.02.01ПЗ | Система забезпечення функціонування | 1 | |
| | | | | конвергентної обчислювальної мережі | | |
| | | | | підприємства на основі використання | | |
| | | | | «хмарних» технологій | | |
| | | | | Пояснювальна записка | | |
| | | | | | | |
| А1 | | 4 | КвРКІ. 180224.18.02.01Е8 | Система забезпечення функціонування | 1 | |
| | | | | конвергентної обчислювальної мережі | | |
| | | | | підприємства на основі використання | | |
| | | | | «хмарних» технологій | | |
| | | | | Конвергентна мережа | | |
| | | | | підприємства | | |
| | | | | | | |
| А1 | | 5 | КвРКІ. 180224.18.02.01Е8 | Система забезпечення функціонування | 1 | |
| | | | | конвергентної обчислювальної мережі | | |
| | | | | підприємства на основі використання | | |
| | | | | «хмарних» технологій | | |
| | | | | Структура системи захисту даних | | |
| | | | | в «хмарному» середовищі | | |

КвРКІ. 180224.18.02.01 ВП

| Зм. | Арк. | № Докум. | Підп. | Дата |
|---|---|---|---|---|
| Розробив | | Бойко В.О. | | 06.06 |
| Перев. | | Орленко В.С. | | 06.06 |
| Н. контр. | | Мостовий С.В. | | 08.06.22 |
| Затв. | | Кльоц Ю.П. | | 08.06.22 |

Система забезпечення функціонування конвергентної обчислювальної мережі підприємства на основі використання «хмарних» технологій
Відомість проекту

| Літера | Аркуш | Аркушів |
|---|---|---|
| и | 1 | 2 |

ХНУ гр. КІ-18-2

| Формат | Зона | Позиц | Позначення | Найменування | Кільк. | Прим. |
|---|---|---|---|---|---|---|
| A1 | | 6 | КвРКІ. 180224.18.02.01Е8 | Система забезпечення функціонування | 1 | |
| | | | | конвергентної обчислювальної мережі | | |
| | | | | підприємства на основі використання | | |
| | | | | «хмарних» технологій | | |
| | | | | Алгоритм розподілу ІОР у | | |
| | | | | конвергентній обчислювальній | | |
| | | | | мережі підприємства | | |
| | | | | | | |
| A1 | | 7 | КвРКІ. 180224.18.02.01Е8 | Система забезпечення функціонування | 1 | |
| | | | | конвергентної обчислювальної мережі | | |
| | | | | підприємства на основі використання | | |
| | | | | «хмарних» технологій | | |
| | | | | Структура конвергентної обчи- | | |
| | | | | слювальної мережі підприємства | | |
| | | | | | | |
| A1 | | 8 | КвРКІ. 180224.18.02.01Е8 | Система забезпечення функціонування | 1 | |
| | | | | конвергентної обчислювальної мережі | | |
| | | | | підприємства на основі використання | | |
| | | | | «хмарних» технологій | | |
| | | | | Модулі та функції монітора- | | |
| | | | | розподільника інформаційно- | | |
| | | | | обчислювальних робіт | | |

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій
Кафедра Кібербезпеки
Освітній рівень БАКАЛАВР
Галузь знань 12 Інформаційні технології
Спеціальність 123 Комп'ютерна інженерія
Освітня програма Освітньо-професійна програма підготовки бакалаврів

**ЗАТВЕРДЖУЮ**
Завідувач кафедри кібербезпеки
к.т.н. доцент Кльоц Ю.П.
"01" 03 2022 року

**ЗАВДАННЯ**
**НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Бойко Владиславу Олександровичу
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Система забезпечення функціонування конвергентної обчислювальної мережі підприємства на основі використання «хмарних» технологій
Науковий керівник Орленко Вікторія Сергіївна, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджена наказом № ректора університету додаток № від . .2022
2. Строк подання студентом проекту (роботи) на кафедру 01.06.2022.
3. Вихідні дані до проекту (роботи) Розробити систему забезпечення функціонування конвергентної обчислювальної мережі підприємства на основі використання «хмарних» технологій.
4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) Дослідження інформаційно-обчислювального середовища підприємств. Інформаційні процеси в «хмарних» технологіях. Система забезпечення розподілу об'єктів в інформаційно - обчислювальному середовищі підприємства Структура модулів системного забезпечення розподілу об'єктів між серверами інформаційно - обчислювальної мережі підприємства та «хмарними» серверами
5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) 1. Конвергентна мережа підприємства (Е8) 2. Структура системи захисту даних в «хмарному» середовищі (Е8) 3. Алгоритм розподілу ІОР у конвергентній обчислювальній мережі підприємства (Е8) 4. Структура конвергентної обчислювальної мережі підприємства (Е8) 5. Модулі та функції монітора-розподільника інформаційно-обчислювальних робіт(Е8)

6. Консультанти розділів дипломного проекту (роботи)

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---|---|---|---|
| | | завдання видав | завдання прийняв |
| Нормоконтроль | Мостовий С.В., ст. викладач кафедри КбКСМ | - | |
| Антиплагіат | Мостовий С.В., ст. викладач кафедри КбКСМ | - | |

7. Дата видачі завдання: «_1_»___лютого_____2022р.

## КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів дипломної роботи | Строк виконання етапів роботи | Примітка |
|---|---|---|---|
| 1 | Грунтовне ознайомлення з предметною галуззю | Лютий - 1 декада | Виконано |
| 2 | Визначення структури кваліфікаційної роботи | Лютий - 2 декада | Виконано |
| 3 | Робота над першим розділом роботи | Березень - 1 декада | Виконано |
| 4 | Робота над другим розділом роботи | Березень - 2 декада | Виконано |
| 5 | Робота над третім розділом роботи | Березень - 3 декада | Виконано |
| 6 | Оформлення графічного матеріалу | Травень - 1 декада | Виконано |
| 7 | Оформлення пояснювальної записки | Травень - 2 декада | Виконано |
| 8 | Попередній захист кваліфікаційної роботи | Травень - 2 декада | Виконано |
| 9 | Доопрацювання кваліфікаційної роботи | Травень - 3 декада | Виконано |
| 10 | Подання роботи для перевірки на плагіат | Травень - 3 декада | Виконано |
| 11 | Захист кваліфікаційної роботи | Червень - 1 декада | Виконано |

Студент _____ Бойко В.О.
Підпис Ініціали, прізвище

Керівник проекту (роботи) _____ Орленко В.С.
Підпис Ініціали, прізвище

# АНОТАЦІЯ

Тема дипломної проекту: «Система забезпечення функціонування конвергентної обчислювальної мережі підприємства на основі використання «хмарних» технологій».

Автор роботи: студент групи КІ – 18 – 2 Бойко В.О.

Керівник роботи: ктн. доц. Орленко В.С.

Пояснювальна записка: 65 с., 18 рисунків, 3 таблиці, 2 дод., 18 джерел.

Графічна частина: 5 плакатів.

Перелік ключових слів: конвергентна обчислювальна мережа, канали зв'язку, віртуальні мережі, "хмарні" технології, мережевий трафік.

В першому розділі проведено аналіз та дослідження розв'язуваних завдань та стану інформаційно-обчислювального середовища підприємств, зроблено висновок про структуру інформаційно-обчислювальної мережі підприємства та переходу на конвергентні обчислювальні мережі, з включенням «хмарних» структур. Показано особливості конвергентної обчислювальної мережі підприємства із застосування «хмарних» технологій.

У кваліфікаційній роботі вирішенні наступні задачі: запропонована структура системи захисту даних в «хмарному» середовищі; розроблений алгоритму оптимального розподілу інформаційних об'єктів підприємства конвергентної обчислювальної мережі; розроблено структуру модулів управління розподілом інформаційних об'єктів між «хмарними» серверами та серверами підприємства інформаційно-обчислювальної системи.

Підпис студента

08.06.2022
Дата

ЗМІСТ

# ВСТУП

Сучасний розвиток інформаційно-обчислювальних мереж, до організації обчислювальних процесів, пред'являє нові вимоги, що в них відбуваються. Відповідно, гостро стоїть проблема підвищення ефективності прийнятих рішень щодо функціонування та проектування програмного, математичного, та алгоритмічного забезпечення інформаційно-обчислювальних систем (ІОС) підприємств. Основними активно розвиваючими ІОС є засоби та способи ефективної та швидкої обробки та зберігання даних, інтеграції із зовнішніми системами [2].

На етапі розвитку сучасних інформаційних технологій підприємства використовують конвергентні обчислювальні мережі, є сукупність ресурсів інформаційно-обчислювальних систем разом із віртуальними «хмарними» обчислювальними ресурсами мережі.

Аналіз сучасних тенденцій розвитку засобів автоматизації управління підприємствами розвинених країн показує поняття «інформаційно-обчислювальна система» у більш широкому представленні «Інформаційно-обчислювальне середовище» підприємства, побудованому на основі конвергентних обчислювальних мереж [4].

Інформаційно-обчислювальне середовище - сукупність засобів передачі, апаратних та програмних ресурсів обчислювальної техніки, обробки та накопичення даних, забезпечують функціонування на підприємстві інформаційно - обчислювальній мережі. Конвергентна інфраструктура є рішенням різного рівня набору компонентів, включає виділені сервери, віртуальні мережі, організації, "хмарні" ресурси, центри зберігання даних, центри обробки даних.

Для підприємства, перевага конвергентної мережі в тому, що вона є єдиною віртуальною мережею, при її експлуатації, з модулями розподілу навантаження між окремими підмережами та модулями управління інформаційно обчислювальними ресурсами, представляють собою віртуальні обчислювальні вузли. Для користувачів інформаційно-обчислювальна систем конвергентні мережі представляють єдину

мережу з відповідним набором обчислювальних ресурсів незалежно від географічної віддаленості віртуальних підмереж, від складу та будови конвергентних обчислювальних мереж.

Інформаційно-обчислювальна система підприємства функціонує під управлінням програмного забезпечення, таким чином в перспективних розробках конвергентних обчислювальних мереж, програмного забезпечення є найважливішої складової, роль його стає домінуючою неухильно зростає. Програмного забезпечення дозволяє проводити розподіл інформаційно-обчислювальних робіт, процеси віртуалізації ресурсів, розмежовувати рівні доступу між ними, утворювати віртуальні підмережі, забезпечувати реалізацію процесів реагування на апаратні та програмні помилки, що виникають у системі під час експлуатації.

Питання оптимізації інформаційних процесів, з включеною до її складу «хмарною» інфраструктурою, у конвергентній обчислювальній мережі підприємства, залишаються новими. Висока вартість, складність інформаційно-обчислювальних систем, вимагають необхідність пошуку шляхів підвищення ефективності системного програмного забезпечення за рахунок оптимізації інформаційних процесів.

Проведений аналіз та дослідження оказав, що на сьогодні не розроблено алгоритмів та системного програмного забезпечення оптимізації інформаційних процесів підприємства у конвергентній обчислювальній мережі, а використання існуючих підходів до оптимізації інформаційних процесів на підприємстві в обчислювальних мережах неприйнятне, не враховує особливостей конвергентних обчислювальних мереж із використанням «хмарних» технологій.

Метою кваліфікаційної роботи є оптимізація обчислювальних процесів у конвергентній мережі підприємства. Відповідно до поставленої мети у кваліфікаційній роботі необхідно вирішити наступні задачі: аналіз функціонування конвергентної обчислювальної мережі підприємства; розробка структури системи захисту даних в «хмарному» середовищі; розробка алгоритму оптимального розподілу інформаційних об'єктів підприємства конвергентної обчислювальної мережі; розробка структури модулів управління розподілом інформаційних об'єктів

між «хмарними» серверами та серверами підприємства інформаційно-обчислювальної системи.

Практична значимість: алгоритми та системне забезпечення можуть бути використані для проведення оптимізації розподілу інформаційно - обчислювального навантаження у конвергентних обчислювальних мережах підприємства із підключеною «хмарною» інфраструктурою.

# 1 АНАЛІЗ ТА ДОСЛІДЖЕННЯ ІНФОРМАЦІЙНО-ОБЧИСЛЮВАЛЬНОГО СЕРЕДОВИЩА ПІДПРИЄМСТВ

## 1.1 Інформаційно-обчислювальне середовище підприємства

Сучасні підприємства застосовують складні інформаційних систем, які відповідать вимогам постійно ускладнюючим технологіям. Ефективна діяльність підприємства неможлива без інформаційної системи, яка повністю відповідає галузевим потребам.

Конвергенція – одна з основних тенденцій, на сучасному етапу, в інформаційних технологіях, дозволяє використовувати існуючі та нові технології, об'єднуючи інформаційно-обчислювальні системи підприємства з "хмарними" ресурсами. Конвергентні мережі, конвергентна інфраструктура, конвергентні протоколи роблять реалізацію моделі адаптивної інформаційної системи можливим у вигляді масштабуємого пулу ресурсів, керованої як сервіс і розділяємої додатками. В ресурсний пул можуть включатись фізичні та віртуальні ресурси: серверні, системи зберігання даних мережеві. Управління сервісом передбачає автоматизацію процесів замовлення ресурсів, максимальний ступінь уніфікації, забезпечення доступності та продуктивності, виділення ресурсів, планування потужностей. Конвергентна інфраструктура дозволяє проводити зберігання та обробку даних за рахунок високого рівня віртуалізації ресурсів у віртуальному середовищі. «Хмарні» технології є важливим елементом конвергентних обчислювальних мереж підприємства.

Апаратний склад конвергентної мережі підприємства може складатися з готових компонентів запропонованих одним виробником, з різного набору компонентів різних виробників для побудови конвергентної мережі.

Існують рішення для побудови інформаційного конвергентного середовища на основі апаратної платформи: VSPEX представляє рішення для розгортання та побудови конвергентних структур, організації взаємодії в залежності від класів задач на підприємстві. Для вирішення різного класу задач підприємства, базові

зміни можна розширити створенням підмереж; VBlocks дозволяє конфігурувати обчислювальну конвергентну віртуальну мережу на основі відповідних запитів підприємства та для управління конвергентним середовищем спроектувати роботу модульних компонентів.

Для побудови конвергентних обчислювальних мереж підприємства задіють наступні рішення (табл. 1.1):

Таблиця 1.1 - Рішення для побудови конвергентних обчислювальних мереж

| Назва рішення | Зображення | Склад рішення |
|---|---|---|
| DELL ActiveSystems |  | Гіпервізор Microsoft Hyper-V, керуючий модуль Dell OpenManage, ОС Windows Server Datacenter Edition, консоль управління мережевим оснащенням vCenterServer, системи зберігання даних EqualLogic PS4100XV, комутатори PowerConnect 6024, стійкові сервери PowerEdge R610 |
| HP Converged System |  | Гіпервізори Hyper-V, серверна стійка HP 42U, vSphere, комутатори HP 2900 series, стійкові сервери HP DL 360/380, системи зберігання даних HP 3PAR 7200, програмне забезпечення HP Insight Control Server Module plug-in for vCente |
| IBM PureSystems |  | Гіпервізори PowerVM, KVM, Microsoft Hyper V або VMware ESX, система управління серверами IBM Flex System Manager, IBM PureFlex System, дискова підсистема IBM Storwize V7000, серверні станції IBM flex system bladeCenter. |

Для забезпечення конвергентного середовища підприємства до складу програмних пакетів входять: керуюче програмне забезпечення для організація взаємодії модулів системи; гіпервізор – для організації віртуальних машин, гостьових ОС, виконання кількох процесів на одному обчислювальному сервері. Технологія розгортання конвергентної мережі на підприємстві дозволяє скорочувати

число серверних станцій, забезпечити інтеграцію програмного забезпечення в інформаційно-обчислювальне середовище за рахунок технологій повної віртуалізації, надасть змогу зменшити число фізичних серверів. Інтеграція «хмарної» структури в інформаційно-обчислювальне середовище можна реалізувати у вигляді організації тунелю між інформаційно-обчислювальним середовищем підприємства комунікаційним сервером та віртуальним комунікаційним комп'ютером «хмари». При такому об'єднанні робочі місця внутрішньої мережі ІОС будуть взаємодіяти з «хмарними» серверами з привілеями та рівнем доступу, як станції що розташовані в локальній мережі ІОС. «Хмарна» структура може бути інтегрована, відповідно до вимог внутрішньої політики організації з передачі, обробки, зберігання даних, шляхом підключення гібридної, публічної або приватної "хмари". Інтеграція може бути виконана, у випадку розміщення дистрибутиввів додатків «хмарній» архітектурі на основі підключення сервісних послуг, доступ організується через API за допомогою клієнтських частин програм.

Інтеграція проводиться на рівні дискових підсистем, пристроїв зберігання даних, дозволяє підприємству отримати сегмент мережі для виконання операції збереження даних у довгостроковій пам'яті на високій швидкості Інтеграція «хмарних» ресурсів з ІОС підприємства дозволяє отримати конвергентну інформаційно - обчислювальну мережу [3, 4] (рис. 1.1).

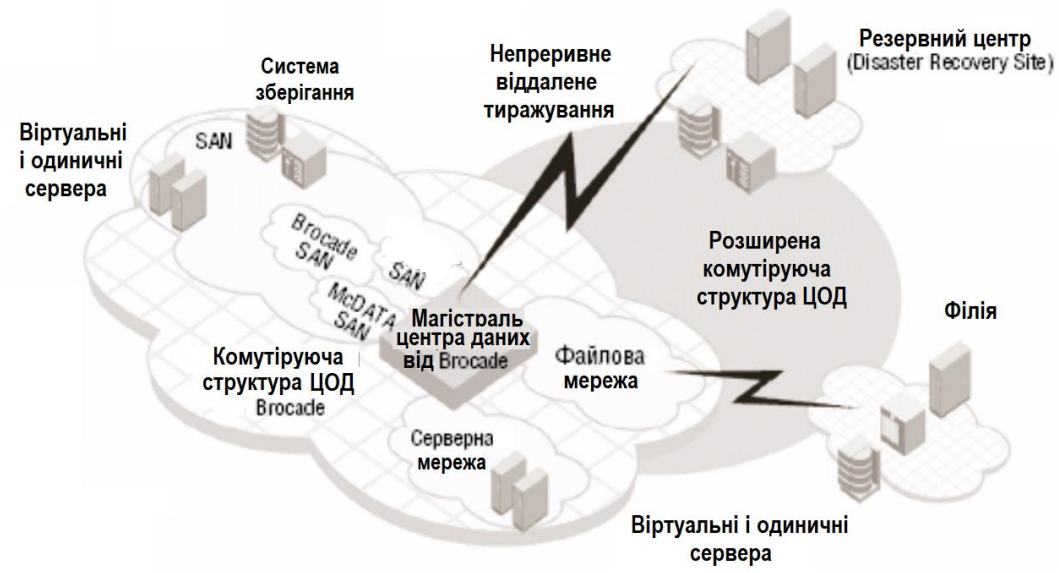



Рисунок 1.1 - Конвергентна мережа підприємства

Організація конвергентної мережі підприємства дозволяє вибирати, залежно від структури ІОС та задач підприємства, різний рівень взаємодії. Рівень взаємодії дозволяє побудувати єдину віртуальну мережу, на якому використовується побудова моделі конвергентної мережі, та надає єдині можливості з управління структурою підприємства, масштабувати ІОС, без додаткового налаштування сегмента мережі, що в даній ситуації, суттєво знижує витрати на експлуатацію мережі.

При побудові інформаційно-обчислювального середовища необхідно враховувати специфічні вимоги підприємства до використання «хмарних» ресурсів: забезпечення механізму аутентифікації та авторизації при використанні інформаційної системи «хмари»; конфіденційність всіх оброблюваних та збережених даних; використання технологій закритої «хмари» (private cloud); захист конфіденційних даних від несанкціонованого доступу; захист каналів передачі даних між банківською інформаційною системою та «хмарним» провайдером; здійснення, відповідно до поточного законодавства, обробки у «хмарі» конфіденційних даних; забезпечення взаємодії, з метою управління інцидентами, співробітників підприємства з технологічною платформою «хмари»; можливості організації зовнішнього та внутрішнього аудиту у сфері інформаційної безпеки «хмарної» інфраструктури. [4,7].

Склад програмного забезпечення мережі на підприємства визначається задачами, які вирішуються. Інформаційне забезпечення мережі підприємства вибирається виходячи із задач накопичення, зберігання, перетворення, представлення даних, що зберігаються у базах даних підприємства. Методи зберігання та оптимізації даних вибираються виходячи з пріоритету необхідних для виконання операцій та списку задач мережі[3, 9].

На етапі проектування ІОС з основних вимог є: можливості масштабування систем без втрати продуктивності, рівень захищеності інформації, ступеня захищеності та каналів передачі конфіденційних даних.

Інфраструктура інформаційно – обчислювальної системи може бути поділена: глобальний рівень - реалізуються базові принципи користувальницької та внутрішньосистемної взаємодії з мережею; низькорівневий доступ до управління

структурою мережі; менеджмент-рівень. На рівні менеджменту вирішуються задачі підтримки, розташованого на глобальному рівні, працездатності базового забезпечення.

Основні рівні роботи системи: управління навантаженням; управління подіями; зберігання та відновлення інформації; управління об'єктами – носіями даних; управління проблемами, що виникають в ІОС; управління захистом.

На інформаційно - обчислювальну мережу підприємства покладаються задачі розпаралелювання обчислювальних процесів інформаційної мережі. Клієнтські машини, на яких проводять операції із інформаційною системою, повинні працювати системами обробки та обслуговування даних паралельно. Частина процесів в мережі, в залежності від зміни поведінки зовнішньої системи середовища, є стохастично виникаючими процесами. Зовнішні системи запускають, при цьому, механізми саморегуляції системи, дозволяють вносити коригування в даних бази даних, в ході проведення поточної операції. в автоматизованих інформаційних системах застосовуються складні механізми управління мережевими об'єктами підприємства.

Автоматизовані інформаційні системи підприємств проектуються виходячи з: великого об’єму інформації, при передачі по мережі за одиницю часу; високої завантаженості та застосування математичних алгоритмів; багатопотокового режиму запису та читання даних у реальному часі;необхідності обробки великих об’ємів даних за найменший час обробки; можливості роботи інформаційних систем паралельно із централізованим сховищем даних; можливості перенесення та масштабування систем без повної модифікації програмного забезпечення; можливість програмного забезпечення працювати з різними принципами організації, зберігання та передачі даних, з операційними системами різного сімейства.

Архітектура обчислювальної мережі підприємства проектується виходячи із об’єму необхідної обчислювальної потужності, задач підприємства, розроблених принципів користувальницької та внутрішньосистемної взаємодії, відмовостійкості системи та необхідного ступеня захищеності.

При зміні структури інформаційної мережі підприємства необхідно враховувати складність проектування нового та складність існуючого на підприємстві системного програмного забезпечення.

Особливості програмного забезпечення в ІОС підприємства: різний ступінь уніфікації компонент та модулів програмного продукту; наявність трансакційних додатків, висока складність міжмодульної взаємодії інтерфейсів, для яких необхідні підвищені вимоги безпеки операцій, продуктивності, надійності, наявність запитів на обробку та вибірку даних великого об’єму, наявність систем підтримки прийняття рішень; висока частка вперше створюваних програмних систем та модулів взаємодії; наявність неоднорідного обчислювального та територіально розподіленого середовища; необхідність інтеграції міжплатформних систем; наявність інтерфейсів взаємодії із зовнішніми системами зберігання та обробки даних.

Для проведення операцій між інформаційно – обчислювальною системою підприємства використовується протокол S.W.I.F.T. Мережа на основі S.W.I.F.T стандарту дозволяє будувати інформаційні мережі передачі даних між філіями підприємств, забезпечуючи при цьому, захист даних використовуючи комбінування логічних та фізичних способів захисту інформації.

## 1.2 «Хмарні» технології у конвергентній обчислювальній мережі підприємств

На теперішній час «хмара» на вимогу клієнта надає оренду обчислювальних потужностей. Найбільшими компаніями, що надають послуги: Microsoft, Amazon, Red hat company, HP.  Основні стратегії розвитку «хмарних» структур - з метою створення конвергентних «хмарних» структур об'єднання мереж. Конвергентна мережа - сукупність мережі інформаційно– обчислювальної системи підприємства та «хмарних» обчислювальних серверів. Зв'язок серверів відбувається на віртуальному рівні у «хмарному» центрі обробки даних, дозволяє здійснювати балансування і маршрутизацію потоку пакетів даних, що надходять на зовнішньому

рівні декількох центрі обробки даних і на внутрішньому, в одному набору серверів [5, 7].

Основні операції відносяться до організації швидкого доступу до даних, зберігання та обробки великих об’ємів даних, виконуються в інформаційно-обчислювальній системи підприємств. «Хмарні» технології надають підприємствам обчислювальні системи підвищеної надійності з можливостями резервного копіювання даних пов'язаних із втратами даних в інформаційно-обчислювальній системі, швидкого розгортання.

Конвергентна організація інформаційно-обчислювальної мережі підприємства відрізняється на всьому життєвому циклі зручністю підтримки. У конвергентній мережі, в силу високого ступеня віртуалізації функцій, не призводить до зміни структури зміна топологій однієї з підмереж на рівні керування. Підприємства використовують інформаційні технології системи 5-го покоління, розподілені мережі з наявністю дискових підсистем, використання клієнт-серверних технологій, менеджерів транзакцій, використання багатозадачних операційних систем [6].

При реалізації інформаційно-обчислювальних систем, надати підприємству необхідність міграції програмного забезпечення між ОС при підключенні, розташованих на відмінних від базового сегмента, сегментів віртуальної мережі, можливість переходу на різні платформенні структури.

Застосування «хмарних» технологій підприємством дозволяє задіяти повний рівень віртуалізації програмного забезпечення та операційних систем, надає великі можливості для надання міграції ресурсів.

Для використання «хмарних» серверів, виходячи з концепцій «хмарних» структур, як обчислювальних сегментів для підприємств, використання загальнодоступних центрів обробки та зберігання даних та публічної "хмари" не допускається. Рекомендується використання закритої (приватної) «хмари» [3, 5].

Поширення клієнт-серверних програм, на теперішній час, дозволяє використовувати, як потужні обчислювальні станції, «хмарні» сервери для передачі даних на клієнтські пристрої, обробки запитів користувачів [2] (рис 1.2).

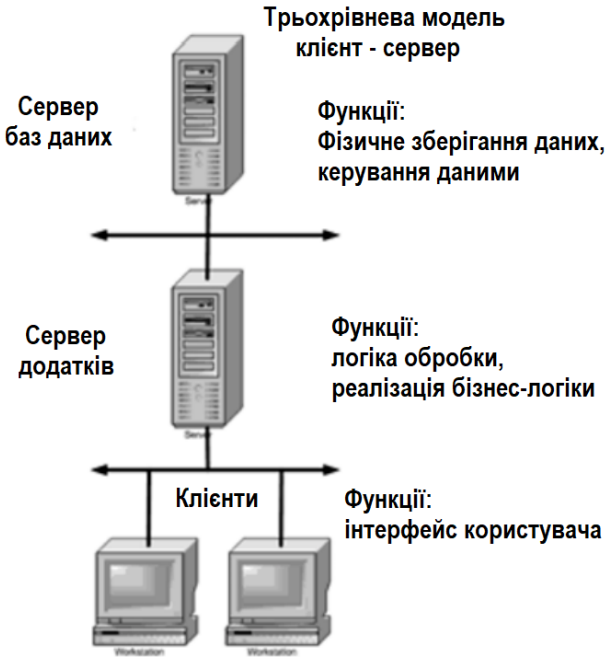



Рисунок 1.2 - Клієнт-серверна архітектура
у інформаційно - обчислювальній мережі

В інформаційно – обчислювальній мережі підприємств програмне забезпечення використовується для автоматизації всіх видів процесів інформаційних технологій (рис. 1.3). Після задіяння «хмарної» інфраструктури в складі інформаційно-обчислювальної система підприємства підтримка та установка апаратного забезпечення передається до технічної підтримки фахівцям "хмарного" провайдера [9].

З метою організації конвергентної структури, при використанні «хмарних» серверів у складі інформаційно-обчислювальної системи, підприємство отримує наступні переваги: сучасну апаратну платформу; високий ступінь еластичності інформаційно-обчислювальної системи; високий рівень надійності процесу зберігання та обробки конфіденційних даних; високу швидкість обробки даних.

Під час розробки програмного забезпечення застосовується технологія моделювання потоків даних (Data flow diagrams DFD). Діаграми потоків даних використовуються для відображення процесів роботи з даними: отримання вхідних даних, перетворення і повернення для передачі іншим процесів. При побудові DFD

використовуються: зовнішні сутності; процеси; системи та підсистеми; потоки даних; джерела зберігання даних.


Рисунок 1.3 - Рівні використання програмного забезпечення ІОМ підприємств

## 1.3 Дослідження інформаційних процесів у конвергентній обчислювальній мережі підприємства

При побудові конвергентної обчислювальної мережі (КОМ) на основі ІОМ підприємства підвищені вимоги пред'являються до організації інформаційно-обчислювальної мережі, необхідно врахування ряд факторів: великі об’єми оброблюваних даних; обмеження на час отримання відповіді від сервера; розподілена обробка даних, що надходять з різних віртуальних підмереж [7].

При проектуванні, експлуатації та розробці КОМ, основною задачею є підвищення ефективності функціонування, які відбуваються в мережі, обчислювальних процесів. Для вирішення розглянутої задачі, пропонується підхід, на представленні інформаційної взаємодії в конвергентній обчислювальній мережі, як сукупності взаємодій технічного персоналу, співробітників підприємства із

системою. Будь-яку взаємодію можна відобразити послідовністю етапів обробки та передачі інформації [9]. Каналами передачі даних здійснюється передача інформації, які пов'язують окремі вузли конвергентної обчислювальної мережі. «Хмарними» та обчислювальними серверами ІОМ здійснюється обробка інформації. Обчислювальні процеси в конвергентної обчислювальної мережі відбуваються між різними об'єктами мережі: комунікаційними пристроями, обчислювальними серверами "хмари", обчислювальними серверами ІОМ, робочими місця користувачів. На основі характеристик складових етапів можуть бути отримані кількісні характеристики обчислювального процесу. В будь-якій конвергентно-обчислювальній мережі кількість ресурсів, кількість користувачів, кількість видів інформаційно обчислювальних робіт обмежені, можуть бути перенумеровані інформаційні процеси. Множина інформаційних процесів $M_i = \{i\}$ - кінцева.

Будемо вважати, що кожен етап $A_{ij}$ пов'язаний лише з одним критичним таким, який є суттєвим при реалізації інформаційних процесів. Некритичні ресурси можна врахувати побічно, шляхом завдання інформаційних процесів граничного числа, які одночасно виконуються на критичному ресурсі.

З конкретним користувачем пов'язується інформаційний процес і характеризується множиною наборів етапів, в якому визначаються кінцевий і початковий етапи, кінцевих етапів може бути декілька. Сукупність інформаційних процесів характеризує не тільки конвергентну обчислювальну мережу, а також її користувачів. ІОС підприємства призначена для виконання інформаційно-обчислювальних робіт, має свої задачі та цілі. Якість інформаційних технологій проектування конвергентної обчислювальної мережі, з урахуванням особливостей інформаційно-обчислювальної системи підприємства, має оцінюватися з використанням показників ефективності, які визначають ступінь пристосованості ІОС підприємства до вирішення покладених на неї задач.

Показник ефективності повинен враховувати властивості та особливості системи, умови її функціонування, тому буде залежати від параметрів потоку вхідних заявок на виконання інформаційно-обчислювальних робіт (ІВР),

характеристик виконуваних інформаційно-обчислювальних робіт, параметрів та структури конвергентної обчислювальної мережі, а також параметрів впливу зовнішнього середовища (виходу з ладу програмних та технічних засобів системи). Показник ефективності визначається процесом функціонування обчислювальної системи, множина процесів функціонування, які відрізняються режимами та умовами роботи, відображаються на множині значень показника ефективності ІОС підприємства.

Для складних систем еможливо виділити єдиний показник ефективності, який дозволить охарактеризувати функціонування системи, з точки зору користувачів. Тому, для оцінки функціонування системи використовують множину показників ефективності, кожен із яких характеризує частну ціль досягнення системою. Відповідні показники ефективності і частні цілі у системному плані мають бути узгоджені, досягнення частної цілі має відповідати виконанню основної задачі системи. Тобто будь-який частний показник ефективності, який є деякою характеристикою функціонування конвергентної обчислювальної мережі, може бути, за результатами аналізу взаємодіючих інформаційних процесів, розрахований.

Значний інтерес представляє визначення показників ефективності, що характеризують частні інформаційні процеси функціонування конвергентної обчислювальної мережі (за даний період середнє завантаження ресурсів системи, середній час реалізації заданого інформаційного процесу). Аналіз частних показників ефективності дозволяє прийняти соєчасно ряд важливих рішень: усунення «вузьких» місць у системі, організації обробки потоку інформації вищої категорії терміновості.

Динамічні властивості конвергентної обчислювальної системи оцінюються з використанням коефіцієнта ефективності повноти обслуговування, відношенням числа заявок, обслужених за відповідний інтервал часу, до числа заявок, що надійшли в систему за даний інтервал.

Показника ефективності повноти обслуговування $k$ характеризує можливості обчислювальної системи у періоди «пікових» навантажень. Даний показник може бути використаний для оцінки та характеристики перехідних режимів роботи

обчислювальної системи. Використовуючи показники ефективності, можна оцінювати та будувати характеристики інших властивостей обчислювальної системи. Такі властивості обчислювальної можуть бути охарактеризовані відношенням показника ефективності *F*, обчисленого з урахуванням впливу фактора (виходу елементів системи з ладу через зовнішні впливи), до показника ефективності $F_0$ при ідеальних умовах функціонування обчислювальної системи: $k_f = F/F_0$. Величина $k_f$ - коефіцієнтом зниження ефективності (внаслідок зовнішніх впливів). У деяких випадках зручніше використовувати різниці відповідних показників для отримання оцінки $\Delta F = |F_0 - F|$ яка показує, зміну ефективності обчислювальної системи під впливом даного фактора. Кількісна оцінка властивостей конвергентної обчислювальної системи зводиться до розрахунку показників ефективності, які залежать від інформаційного процесу функціонування системи, які протікають у досліджуваній системі, за результатами аналізу сукупності інформаційних процесів. Для того щоб оцінити властивість конвергентної обчислювальної системи при заданих умовах функціонування та конфігурації (при відомих заявок на виконання інформаційно-обчислювальних робіт та параметрах потоків, параметрах її елементів, відомих зовнішніх впливах, заданої структури системи), необхідно в системі розглянути інформаційні процеси, з урахуванням взаємного впливу інформаційних процесів, розрахувати їх характеристики та обчислити необхідні показники ефективності (коефіцієнти зміни ефективності обчислювальної системи, зниження ефектності). Характеристики інформаційних процесів можуть бути визначені двома підходами: за допомогою моделювання та з проведення випробувань реальних системи [10]. Проведення натурних експериментів пов'язане зі значними витратами і не завжди може бути проведено, для проведення таких експериментів необхідно мати аналог або працюючу обчислювальну систему. Проведення натурних випробувань проводяться на заключних стадіях створення обчислювальної системи, коли більшість технічних рішень реалізовані в системі. Показник ефективності інформаційних технологій

функціонування та проектування конвергентної обчислювальної мережі може бути представлений у вигляді:

$$F = F\left(N, M, S, V\right), \tag{1.1}$$

де $N$ – множина параметрів потоку заявок до системи, на виконання інформаційно-обчислювальних робіт, $M$ − множина параметрів, які окремо характеризують ІОР, пов'язані з реалізацією заявок класів, та визначають витрати ресурсів обчислювальної системи при виконанні інформаційно-обчислювальних робіт; $S$ − множина системних параметрів, визначають структуру та склад інформаційно-обчислювальної системи, алгоритми управління інформаційними процесами у обчислювальній системі, характеристики окремих елементів системи; $V$ − множина параметрів системи, які характеризують зовнішній вплив середовища, завдання потоків виходу з ладу елементів обчислювальної системи під впливом зовнішніх факторів.

Проведений аналіз та дослідження функціонування конвергентної обчислювальної мережі підприємства показав: інформаційні процеси системи можна інтерпретувати як реалізацію алгоритмів у межах корпоративної мети функціонування обчислювальної системи [8].

Незважаючи на суттєві відмінності роботи алгоритмів, вони мають низку загальних суттєвих ознак, що дозволяє, до моделювання інформаційних процесів, сформувати єдиний підхід у конвергентної обчислювальної мережі.

Алгоритми обробки інформаційного потоку, що надходить із різних каналів системи, мають наступні загальні особливості:

1. Алгоритми різних об'єктів обчислювальної системи працюють паралельно, взаємодіючи між собою.

2. Процес, можна розбити більш дрібні підпроцеси, реалізації алгоритму, кожен з яких є також алгоритмом.

3. Час виконання кожного оператора обчислювальної системи є випадковою величиною. Джерелом випадковості–випадкові оброблювані дані, «людський фактор», додаткові невраховані впливи навколишнього середовища.

4. При прийнятті рішення, що призводить до зміни порядку виконання операторів, для зовнішнього спостерігача вибір напряму здійснюється випадковим чином. Джерелом випадковості: випадкові дії; ступінь навченості.

5. При роботі алгоритму присутні оператори формування запиту на обробку даних до інших об'єктів взаємодії, таким чином кожен суб'єкт є джерелом заявок на обслуговування до обчислювальної системи колективного користування.

6. Алгоритм обробляє запити з інших об'єктів взаємодії інформаційної обчислювальної системи, таким чином кожен об'єкт є обслуговуючим каналом у інформаційній системі масового обслуживания. Так у конвергентної обчислювальної мережі, обробка даних має час початку та час закінчення інтерпретації алгоритму системою і може розглядатися як час обробки прийнятої заявки.

## 1.4 Постановка задачі

У конвергентній обчислювальній мережі підприємства одночасно функціонують серверні станції різного рівня, віртуальні обчислювальні сервери «хмари», фахівці «хмарного» провайдера їх обслуговують, сервери підприємства, обслуговуються фахівцями підприємства. Особливості експлуатації «хмарних» серверів та серверів інформаційно-обчислювальної системи підприємства призводить до оптимізації розподілу обчислювальних задач між серверами ІОС підприємства та «хмарними» серверами: мінімізації часу виконання інформаційно-обчислювальних робіт; мінімізації економічних витрат на проведення інформаційно-обчислювальних робіт.

Опираючись на запропонований підхід інформаційних алгоритмічних процесів, задача оптимізації розподілу обчислювальних задач може бути сформульовані в термінах математичного програмування. Запропонований підхід, заснований на формалізованому описі та виділенні сукупностей інформаційних процесів, дозволяє вирішувати задачу раціонального розподілу інформаційних обчислювальних робіт між обчислювальними серверами конвергентної

обчислювальної мережі. Декомпозиція задачі та оптимізації системи за рівнями представлення інформаційних процесів та розв'язуваних задач полегшує дослідження та отримання практичних рекомендацій побудови та функціонування конвергентної мережі підприємства.

На підставі проведеного аналізу та дослідження розв'язуваних завдань та стану інформаційно-обчислювального середовища підприємств зроблено висновок про структуру інформаційно-обчислювальної мережі підприємства та переходу на конвергентні обчислювальні мережі, з включенням «хмарних» структур. Показано особливості у конвергентній обчислювальній мережі підприємства застосування «хмарних» технологій. З метою формалізації та опису процесів функціонування конвергентної обчислювальної мережі підприємства пропонується використовувати підхід інформаційних процесів, дозволяє встановити взаємозв'язок характеристик інформаційних процесів. Сформульована задача розподілу інформаційних об'єктів між «хмарними» обчислювальними серверами та серверами інформаційно обчислювальної мержі підприємства за критеріями вартості витрат та часу на проведення інформаційних обчислювальних робіт.

# 2 ІНФОРМАЦІЙНІ ПРОЦЕСИ В «ХМАРНИХ» ТЕХНОЛОГІЯХ

## 2.1 Загрози та вразливості інформаційно-обчислювальних процесів в «хмарних» технологіях

«Хмарні» технології розподілених інформаційних обчислень надають послуги на вимогу, максимально збільшуючи використання наявних ресурсів та скорочуючи, при цьому капіталовкладення в інфраструктуру підприємства. «Хмарні» технології забезпечують незалежність апаратних/фізичних платформ від існуючих на підприємстві інформаційно-обчислювальних мереж, мобільність додатків, мережевих додатків і інфраструктурних сервісів [5]. З розширенням використання «хмарних» технологій, забезпеченням доступності інформаційно-обчислювальних ресурсів, інформаційно-обчислювальні мережі підприємств стають більш гнучкими, конкурентно спроможними і економічно ефективними. Однак, приймаючи на використання у своїй діяльності «хмарні» технології, безпека інформаційних ресурсів підприємства виявилась найбільш критичним і важливим питанням. Для виявлення проблем безпеки інформаційним процесам підприємства, необхідно провести аналіз вразливостей, загроз та ризиків, що впливають на безпеку використання «хмарних» технологій.

Проведення дослідження та аналізу вразливостей, загроз у вигляді всебічного і систематичного підходу необхідно для забезпечення доступності, цілісності, конфіденційності для розгортання інформаційно-обчислювальних мереж на підприємстві з використанням «хмарних» технологій. Аналіз вразливостей і загроз набирає необхідну базову статистику, у вигляді зовнішніх залежностей, сценаріїв використання, деталей реалізації зовнішньої і внутрішньої безпеки, припущень про реалізацію [6,7]. Розроблено ряд підходів дослідження загроз для проведення аналізу та оцінки вразливостей і загроз: Microsoft’s Threat Analysis and Modeling [6] - заснована на бізнес-цілях підприємства, які необхідно досягнути з використанням інформаційно-обчислювальних процесів «хмарних» технологій. Даний підхід використовуються для класифікації та генерування вразливостей та загроз шляхом

обрахування шкідливого впливу на компоненти інформаційно-обчислювальної мережі; моделювання Microsoft загроз - модель ефективного процесу, включаєв собі п'ять логічних кроків від класифікації вразливостей і загроз до їх усунення; Practical Threat Analysis (практичний аналіз загроз) [8] - визначає для конкретної архітектури інформаційно-обчислювальної системи ефективний план зниження загроз і ризиків для отримання відповідної вимогам підприємства системи; методологія і структура дослідження загроз для локальних мереж [7]. Даний підхід, заснований на проведенні аналізу локальної мережі, дає представлення про функціонування інформаційної системи. по Надає можливість захистити мережеві функції та інформацію від зовнішніх загроз, на основі використання UML-діаграм для визначення всіх ресурсів; дослідження загроз шляхом використання інформаційних обчислень «хмарних» технологій. Користувачі стикаються з множинною ідентифікацією, з різними сферами безпеки. Дослідження представляє собою моделювання вразливостей та загроз, включає проблему широко розповсюдженого та використовуваного комп'ютерного середовища.

У зв'язку з ростом використання «хмарних» технологій як розподілених інформаційних обчислень, розлянуті підходи не включають в себе усі проблеми використання «хмарних» технологій, для вирішення зазначених проблем безпеки інформаційно-обчислювальних процесів підприємства необхідно запропонувати нові підходи до проведення та аналізу вразливостей і загроз.

Для розгортання безпечного інформаційно-обчислювального середовища на підприємстві, в даному дослідженні запропоновані підходи виявлення вразливостей і загроз в інформаційних процесах «хмарних» технологій і визначення та вирішення питань безпеки. Пропонована підхід до проведення дослідження загроз складається з декількох етапів, які надають, після розгляду, повну картину вразливостей та загроз інформаційних обчислень при використанні «хмарних» технологій.

Для вирішення питань безпеки «хмарних» технологій, визначимо загрози і вразливість: загроза - несанкціонований доступ (шкода), може виникнути на підприємстві в результаті вразливості і знищити системну інформацію, активи організації, діяльність підприємства; Вразливість – слабке місце в інформаційно-

обчислювальній системі підприємства, внутрішнього контролю або реалізації процедур системної безпеки, яке може бути викликане або використане відповідними ресурсами загроз [8].

Cloud Security Alliance [9] випустив посібник, в якості постачальника послуг із забезпечення інформаційної безпеки для критичних областей в інформаційно-обчислювальних системах «хмарних» технологій, для керування загрозами і ризиками в інформаційних процесах. Найбільш значні ризики, які пов'язані з «хмарними» технологіями та їх характером інформаційних обчислень, наступні: витік даних - незаконний перегляд даних конкурентами, несанкціонований доступ до інформації, для підприємства є однією з найгірших ситуацій. Шифрування потоку даних може зменшити ризик даної загрози; зловживання «хмарними» службами - провайдери інформаційно-обчислювальних «хмарних» технологій не проводять жорстких реєстраційних процедур, користувач з кредитною карткою для отримання хмарних послуг може зареєструватися [10]. Інтеграція слабких механізмів виявлення шахрайства зловмисникові при реєстрації дозволяє ефективно використовувати відповідні конфіденційні дані з використанням агресивних «хмарних» сервісів, таких як laaS і PaaS;витік (втрата) даних - оцінюється як жахлива і найсерйозніша загроза для користувачів та бізнесу підприємства. Видалення інформації при аваріях чи постачальником послуг, може призвести до втрати даних на підприємстві; викрадення послуги або облікового запису - дозволяє конкурентам (зловмисникам) мати доступ до критично важливих областей послуг інформаційних процесів «хмарних» технологій, красти облікові дані підприємства. Підприємстві необхідно заборонити обмін відповідними обліковими даними між користувачами і різними службами, використовувати надійні методи аутентифікації; небезпечний інтерфейс - клієнти інформаційно-обчислювальних процесів «хмарних» технологій використовують програмні інтерфейси чи інтерфейс API для управління і взаємодії «хмарними» службами. Технології контролю доступу, моніторингу, аутентифікації при використанні API захищають від шкідливих атак інформаційно-обчислювальні ресурси підприємства; відмова в обслуговуванні – DDOS атака (розподілена відмова в обслуговуванні) є основною загрозою

доступності даних, підвищення надійності організації доступу до даних на загальнодоступних «хмарних» сервісах [10]. DDOS атака не дозволяє користувачам «хмарних» сервісів отримати доступ до своїх додатків, даних і не дає можливості, при цьому, дістатися до місця призначення. Постачальники «хмарних» сервісів повинні бути впевнені в захисті даних та доступності, а клієнти «хмарних» технологій - в рівні захисту доступності інформації всередині провайдера; шкідливий інсайдер - система пошкоджена адміністратором, діловим партнером, авторизованим співробітником який має доступ до інформаційних потоків, ресурсів мережі. шкідливий інсайдер порушує доступність, цілісність, конфіденційність ділової інформації; небезпечна міграція віртуальних машин - переміщуючи віртуальні машини під час об'єднаних і гібридних «хмарних» інформаційно-обчислювальних процесів, зловмисники можуть отримати несанкціонований доступ до інформаційних потоків і пероемістити дані на ненадійний хост. Віртуалізація основний компонент laaS для проведення атак зловмисниками. Надійні «хмарні» технології в обєднанні з технологіями шифрування захищають інформаційні потоки від небезпечної міграції віртуальних машин в інформаційно-обчислювальній мережі підприємства; недостатня належна обачність - доступ до пулу «хмарних» ресурсів, покращення безпеки, зниження витрат - найважливіші фактори, які необхідні для підприємства, для прискорення впровадження «хмарних» технологій. Без розуміння середовища Cloud Service Provider (постачальника хмарних послуг), невідповідність очікувань - критичне питання з контролю інформаційної безпеки «хмарних» технологій. Однак, підприємства повинні усвідомлювати ризики і пропозиції постачальників послуг «хмарних» технологій.

Загроза вразливостей існує у всіх інформаційно-обчислювальних мережах, оскільки використовуванні базові компоненти, які розгортають платформу інфраструктуру, додатки, не забезпечують повної ізоляції між мережними підсистемами «хмарних» технологій.

При проектуванні конвергентної обчислювальної мережі підприємства, необхідно враховувати основні характеристики «хмарних» технологій: сучасні «хмарні» сервіси, засоби безпеки контролю, вразливості. Основні вразливості при

використанні «хмарних» технологій: перехват сесії - для отримання несанкціонованого доступу до даних, використання слабких місць веб-служб, відправка зловмисниками команди до веб-додатку для надання зловмиснику можливості здійснити дії, як розсилка спаму в мережу через Інтернет, видалення даних призначених для користувача; вихід за межі віртуальної машини - дозволяє хакерові на віртуальній машині запускати код, який дозволяє операційній системі взаємодіяти безпосередньо з гіпервізором і зламувати доступ до віртуальних машин, хостовій операційній системи. Для запобігання даної вразливості інформаційна система на рівні віртуальних машин повинна виявляти шкідливу активність; застаріла криптографія - відсутність шифрування потоку даних, розробка ненадійного процесу шифрування надає можливість хакеру доступу до закритих даних. Для захисту інформаційно-обчислювальної системи від даної вразливості, необхідно бути впевненим, що конфіденційні дані надійно закриті, використовувати надійний алгоритм закриття даних, надійне зберігання ключів; несанкціонований доступ до інтерфейсу керування - інтерфейс доступу до «хмарних» інформаційних процесів для управління службами за запитом користувача. несанкціонований доступ до закритих даних дозволяє зловмисникам отримати контроль над додатками і користувачами; інтернет-протокол - відсутність відповідних методів аутентифікації, які не входять в базовий інтернет-протокол, дозволяє впроваджувати в мережу хакерам шкідливий трафік. Протокол IP, протоколи TCP і UDP, вразливі для DoS атак (відмова в обслуговуванні), включаючи, при цьому, перехоплення сеансу зв’язку; відновлення даних - «хмарні» технології надаютьможливість перерозподіляти (розподіляти) ресурси між користувачами. Дана еластична характеристика «хмарних» технологій може призвести до витоку даних, крадіжки даних. Більшість підприємств для відновлення даних використовують сторонніх постачальників, вони повинні враховувати ризик при роботі з інформаційними потоками з зовнішніми підприємствами і забезпечувати відповідну перевірку безпеки даних постачальника послуг; виставлення рахунків - лічильники «хмарних» інформаційно-обчислювальних процесів, послуги вимірювання, обліковий запис користувача і обробка, зберігання,

використовуються для вирішення задачі оптимізації надання відповідних послуг. Вразливості включають виставлення рахунків, витік рахунків, обробку облікових даних обліку; замок постачальника - блокування постачальника, коли користувач «хмарних» технологій залежить від одного постачальника обчислювальних послуг і не може мати справу, в даному випадку з іншим постачальником. В даній ситуації, відсутність стандартів є основною причиною, що користувачі не можуть, без проблем, переходити від одного до іншого провайдера.

## 2.2 Підсистема контролю доступом інформаційних процесів в «хмарних» технологіях

Контроль доступу, представляє собою процедуру (політику), яка обмежує, забороняє, дозволяє доступ до інформаційних потоків системи. Крім того, контроль доступу може реєструвати, відстежувати всі спроби доступу до інформаційно-обчислювальної мережі підприємства. Контроль доступу може виявляти клієнтів, які намагаються отримати несанкціонований доступ до інформаційних потоків системи підприємства. Даний механізм, займає важливе місце в захисті в потоків даних підприємства. На сьогодні існують різні підходи реалізації контролю доступу, включаючи: керування доступом на основі ролей, вибіркове керування доступом, мандатне керування доступом. Дані підходи більш відомі як моделі на основі ідентифікації контролю доступу. Управління доступом, в даних підходах, об'єкти (ресурси) і суб'єкти (користувачі) ідентифікуються відповідними унікальними іменами. Ідентифікація може здійснюватися за допомогою ролей, призначених користувачам чи безпосередньо. Методи контролю доступу до інформаційних потоків підприємства ефективні в незмінних розподілених обчислювальних системах, де присутні відомий набір сервісів та відомий набір користувачів. На теперішній час великі відкриті розподілені обчислювальні системи розвиваються швидкими темпами. До таких обчислювальних систем відносяться інформаційно-обчислювальні мережі і «хмарні» технології. Дані обчислювальні системи подібні

до віртуальних мереж з автономними доменами. Взаєзв'язок між ресурсами і користувачами обчислювальної системи динамічні і носять більш спеціалізований характер в «хмарних» технологіях. У даних обчислювальних системах постачальники ресурсів і користувачі не належать до однієї області безпеки. Користувачі ідентифікуються характеристиками або по їх атрибутам, заздалегідь не заданими відповідними ідентифікаційними даними. Традиційні підходи контролю доступу, не ефективні, засновані на ідентифікації, доступ до обчислювальної системи повинен здійснюватися на основі прийнятих рішень, основаних на відповідних атрибутах.

В «хмарних» технологіях автономні домени мають власний набір політик безпеки. Механізм контролю доступу для підтримки різних політик, областей повинен бути гнучким. З розвитком розподілених інформаційно-обчислювальних систем, керування контролем доступу на основі атрибутів стає все більш виходить на перше місце.

Контроль доступу дозволяє керувати ресурсами, користувачами, файлами. Контролює права користувача на доступ до об'єктів (ресурсів), файлів. В інформаційно-обчислювальних системах контролю доступу застосовуються авторизація, аутентифікація, ідентифікація перш ніж предоставити доступ до потоків даних системи.

На ранніх етапах розвитку інформаційних технологій розробники обчислювальних систем усвідомлювали важливість запобігання несанкціонованого доступу користувачів до інформаційних потоків в інформаційних мережах. На даному етапі розвитку інформаційних технологій розроблені відповідні підходи контролю доступу, основним показником була особистість користувача, дозволяла користувачам використовувати ресурси обчислювальної системи. Із зростанням кількості користувачів, розміром та числа обчислювальних мереж, контроль доступу на основі ідентифікації (КДОІ) виявився недостатнім для захисту потоку даних обчислювальних мереж від такого числа користувачів. В КДОІ були введені вдосконалені концепції контролю доступу до даних обчислювальної системи, які включали групу/ користувача/громадськість. КДОІ виявився проблематичним для

розподілених обчислювальних систем. Управління доступом до ресурсів обчислювальних систем стало вразливим для помилок і трудомістким. Був запропонований новий підхід - керування доступом до обчислювальної системи на основі ролей. Керування доступу до обчислювальної системи підприємства на основі ролей визначає на основі ролі доступ користувача до даних. Роль визначається на принципі найменших привілеїв та призначається користувачу. Роль користувача визначається з найменшою кількістю функцій (дозволів), необхідних для виконання необхідної роботи користувачем. Дозволи можуть бути видалені, додані, якщо змінюються в процесі функціонування системи, привілеї для відповідної ролі. Проблеми стали очевидними, коли керування доступу до обчислювальної системи на основі ролей була поширена на адміністративні домени. Виявилося трудомістким, які привілеї асоціювати з роллю. Був запропонований підхід керування доступом до обчислювальної системи на основі атрибутів. Доступ до системи надається по атрибутам (номер телефону дата народження). Розглянуті підходи ще називають методами керування доступом до обчислювальної системи на основі аутентифікації, необхідний зв'язок між доменами, також розглянуті підходи ускладнюють процес призначення прав адміністратора. Загальні схеми використання, можуть бути реалізовані шляхом порушення принципу найменших привілеїв, або скорочення функціональних можливостей.

Контроль доступу до обчислювальних систем на основі атрибутів розширює контроль доступу користувача на основі ролей, наступними функціями: інтерференція атрибутів; децентралізація атрибутів і функцій; делегування повноважень на визначення атрибутів.

Керування доступом до обчислювальної системи на основі атрибутів забезпечує політику конфіденційності. Дозволяє підприємству, при ефективній співпраці, зберігати свою автономність. Керування доступом на основі атрибутів забезпечує автоматичні переговори, які можна перевіряти при необхідності, про довіру.

«Хмарні» технології перспективні для додатків в області інформаційних технологій, однак для підприємств необхідно вирішити ряд питань, пов'язаних із

розгортанням додатків, зберіганням даних середовищі «хмарних» обчислювальних мереж. Забезпечення безпеки даних є з найбільш значних перешкод на шляху впровадження «хмарних» технологій: конфіденційність, нормативно-правова відповідність, правові питання, довіра. Однією з основних задач є забезпечення цілісності та безпеки даних, які зберігаються в «хмарному» середовищі, з огляду великий об'єм даних та критичний характер «хмарних» обчислювальних систем. Необхідно усунути недовіру підприємств до безпеки, зробити «хмарне» обчислювальне середовище надійним і допомогти підприємствам адаптуватися до «хмарного» середовища у відповідних масштабах.

Основні проблеми безпеки даних в «хмарному» середовищі: доступність даних, конфіденційність даних, безпечна передача і розміщення. Втрата даних, зовнішні атаки, проблеми з багатокористувацької атакою, перебої в обслуговуванні - основні проблеми безпеки в «хмарному» середовищі.

Дані в «хмарному» середовищі не повинні бути змінені, втрачені неавторизованими користувачами. Постачальникам «хмарних» технологій довіряють підтримувати точність, цілісність даних. Також важливим аспектом є конфіденційність даних, оскільки користувач зберігає свої конфіденційні (особисті) дані в «хмарному» середовищі. Для забезпечення конфіденційності даних підприємства (користувача) використовуються підходи контролю доступу. За рахунок підвищення надійності «хмарних» обчислень може бути вирішена проблема конфіденційності закритих даних. Цілісність, конфіденційність, безпека даних що зберігаються в «хмарному» середовищі, є важливими вимогами і повинні враховуватися при розміщенні даних в «хмарі».

Для безпечного зберігання даних в «хмарному» середовищі необхідно впровадженя аудиту даних в «хмарне» обчислювальне середовище. Аудит - процес перевірки даних підприємства, може бути здійснений стороннім аудитором так і підприємством (користувачем). Це надає можливість підтримувати цілісність даних, які зберігаються в «хмарному» обчислювальному середовищі. Роль верифікатора доступу до даних розділена на дві частини: приватний аудит, тільки власник (користувач) даних має доступ перевірки цілісності даних, які зберігаються в

«хмарному» обчислювальному середовищі, що призводить до збільшення числа верифікаційних операцій на власника. Друга частина - можливість публічного аудиту, дозволяє будь-якому користувачеві, зробити запит до сервера і виконати перевірку достовірності даних від імені клієнта. Публічний аудит має необхідні можливості для виконання відповідної роботи з перевірки цілісності даних, які зберігаються в «хмарному» обчислювальному середовищі. Важливо, щоб публічний аудит ефективно перевіряв «хмарне» середовище даних без запиту локальної копії даних, повинен володіти нульовими знаннями про інформацію, яка зберігається на «хмарному» сервері.

В «хмарному» обчислювальному середовищі присутні три мережевих об'єкта - «хмарний» сервер, клієнт, механізм що надає можливість діяти від імені клієнта (АВС). Клієнт зберігає дані на сервері, що надається провайдером «хмарних» послуг. АВС, періодично перевіряє цілісність даних, здійснює перевірку даних клієнта, повідомляє клієнта, при цьому, про будь-які помилки, зміни виявлені в даних клієнта. На рис. 2.1 представлена архітектура «хмарного» обчислювального сховища даних.

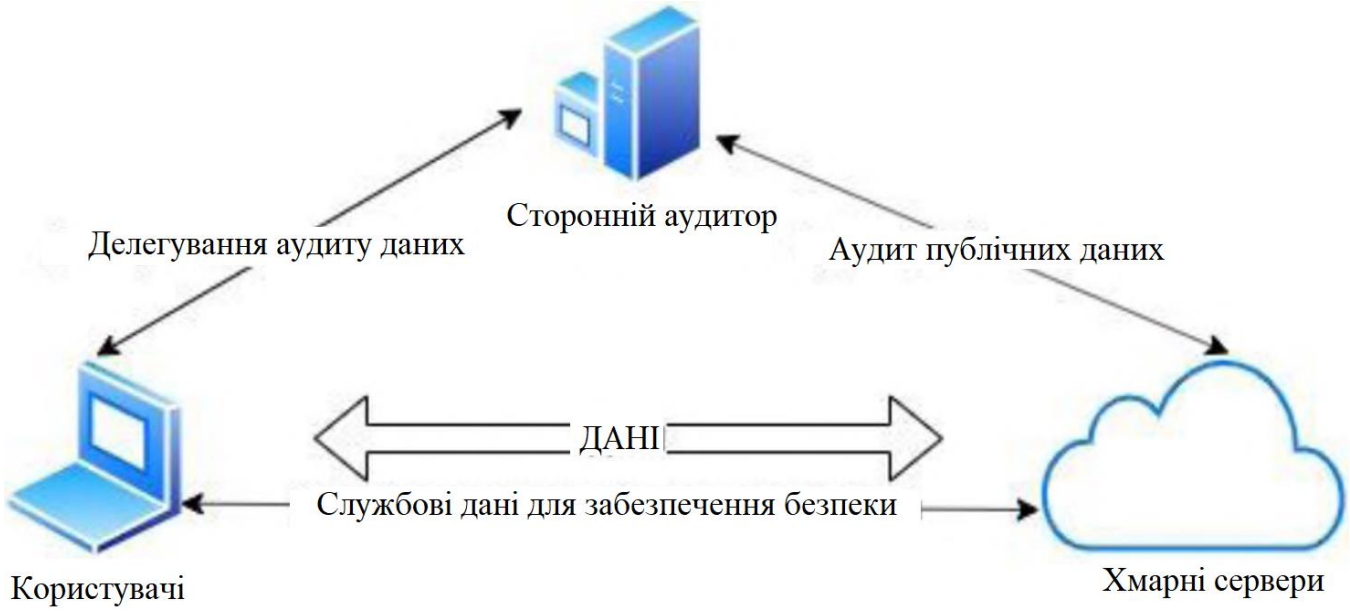



Рисунок 2.1 - Архітектура «хмарного» обчислювального сховища даних

## 2.3 Розробка структури системи захисту інформаційних ресурсів в «хмарних» технологіях

Обрана система для перевірки коректності «хмарних» даних стороннім аудитором, за запитом, періодично без отримання даних, створення додаткового навантаження на «хмарні» сервери, на користувачів «хмарного» обчислювального середовища в режимі онлай. Система забезпечує закриття даних стороннім аудитором в процесі проведення аудиту даних в «хмарному» середовищі. Система підтримує цілісність, конфіденційність та правильність зберігання даних.

Архітектура схеми аудиту складається з трьох компонентів: сторонній аудитор, сховище «хмарного» сервера, користувач даних. Власник даних несе відповідальність за поділ інформаційного файлу на блоки, генерування хеш-значення SHA-512 для файлів, шифрування даних на основі алгоритму AES. «Хмарний» сервер використовується, в даному випадку, для зберігання зашифрованих блоків інформаційних файлів. «Хмарний» сервер немає додаткового навантаження по прведенню обчислення верифікаційних доказів. Верифікаційний доказ - генерація хешей для зашифрованих блоків файлів, генерація цифрового підпису для верифікації користувача. Дану задачу виконує сторонній аудитор. Коли клієнт даних запитує аудит у стороннього аудитора, запитуються закриті дані з «хмарний» сервера. Після отримання даних сторонній аудитор генерує хеш-код для кожного блоку закритих файлів, використовує алгоритм SHA-512. У процесі верифікації генерується аудитором цифровий підпис, який порівнюється з підписом власника даних. Якщо цифрові підписи співпадають, дані не були підроблені сторонніми особами і не пошкоджені. Якщо цифрові підписи не співпадають, цілісність даних підроблена або порушена. Результати перевірки цілісності даних надаються власнику даних. На рис. 2.2 показана архітектура пропонованої схеми аудиту верифікації даних. Користувач даних є важливою частиною пропонованої системи аудиту. Власник даних виконує більшу частину обов'язків, пов'язаних із забезпеченням цілісності даними. У запропонованій системі аудиту власник даних виконує реєстрацію і вхід на «хмарному» сервері, також в сервісі стороннього

аудитора. Користувачу необхідно зареєструватися і бути активним в «хмарному» середовищі. Відповідно, якщо пароль та логін користувача даних існують в базі даних, то будуть зареєстровані в «хмарному» середовищі як дійсні користувачі, інакше отримають повідомлення про помилку.

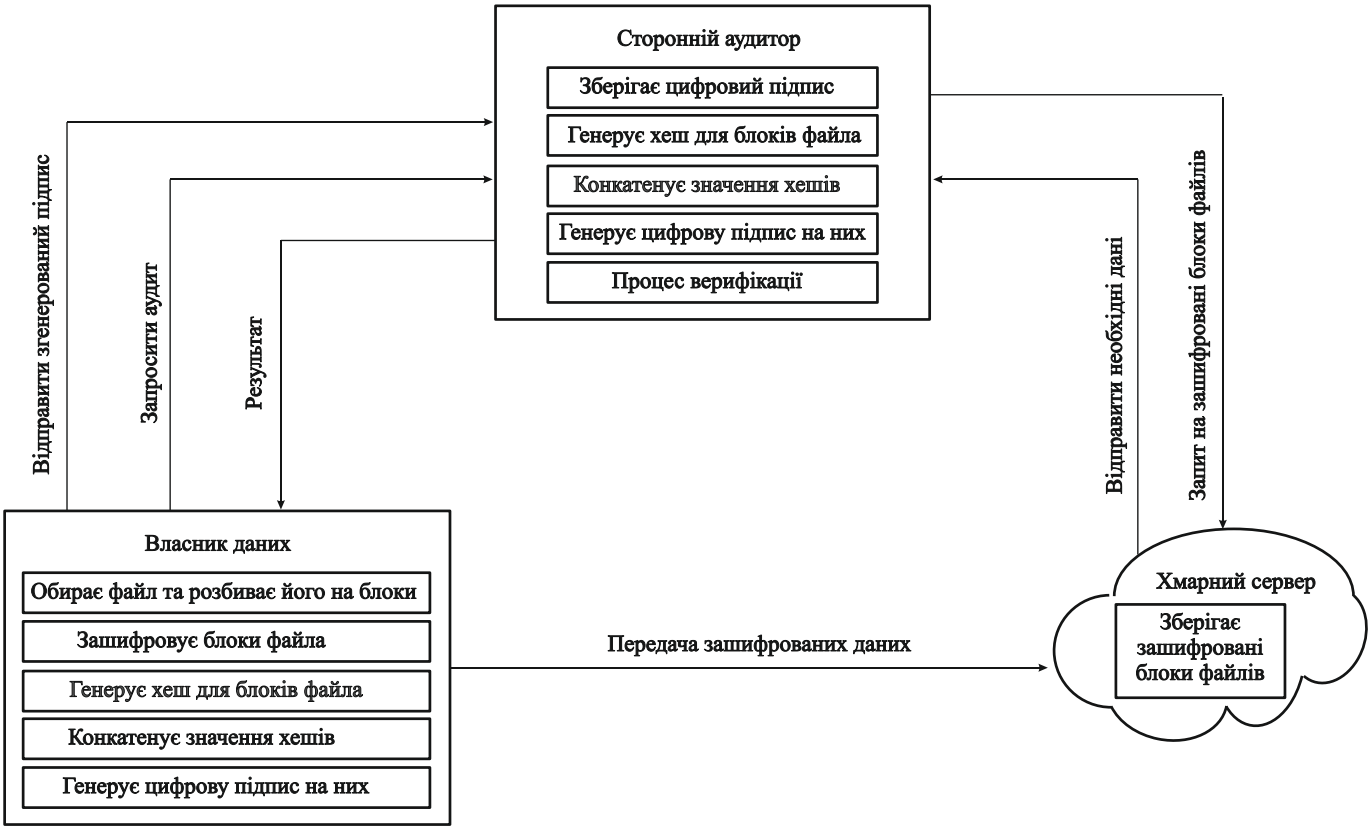



Рисунок 2.2 - Архітектура пропонованої схеми аудиту верифікації даних

Після успішної реєстрації користувач вкаже файл, який необхідно зберегти на «хмарному» сервері. Файл необхідно розбити на блоки. Файл розбивається на блоки, розмір блока файла задається. Для забезпечення конфіденційності даних блоки файла шифруються з використанням алгоритму AES. Зашифровані блоки файлу зберігаються на клієнті, Копія шифрованого файлу передається на «хмарний» сервер. Після шифрування блоків, генерується хеш-значення блоків, використовується алгоритм хешування SHA-512, хеш-код кожного блоку конкатенуються і генерується цифровий підпис (RSA), які використовуються для перевірки модифікації даних. В сервіс стороннього аудитора відправляється цифровий підпис, який використовується для перевірки цілісності даних, які

зберігаються в «хмарному» середовищі.

Зашифровані дані власник зберігає в «хмарному» середовищі. У запрпонованій схемі для перевірки даних використовується сторонній аудитор. Сторонній аудитор проводить аудит даних на вимогу клієнта та періодично. Після отримання запиту на проведення аудиту, аудитор починає процес аудиту. Сторонній аудитор зберігає цифровий підпис, створений власником даних. В процесі верифікації, сторонній аудитор порівнює два цифрових підписи. Якщо співпадають, то цілісність даних підтримується, в іншому випадку цілісність даних порушена, дані змінені чи підроблені. Сторонній аудитор надає власнику даних отримані результати.

Після проведеного дослідження контролю доступу, методів аудиту, можна запропонувати загальну структура системи захисту даних в «хмарному» середовищі (рис. 2.3).

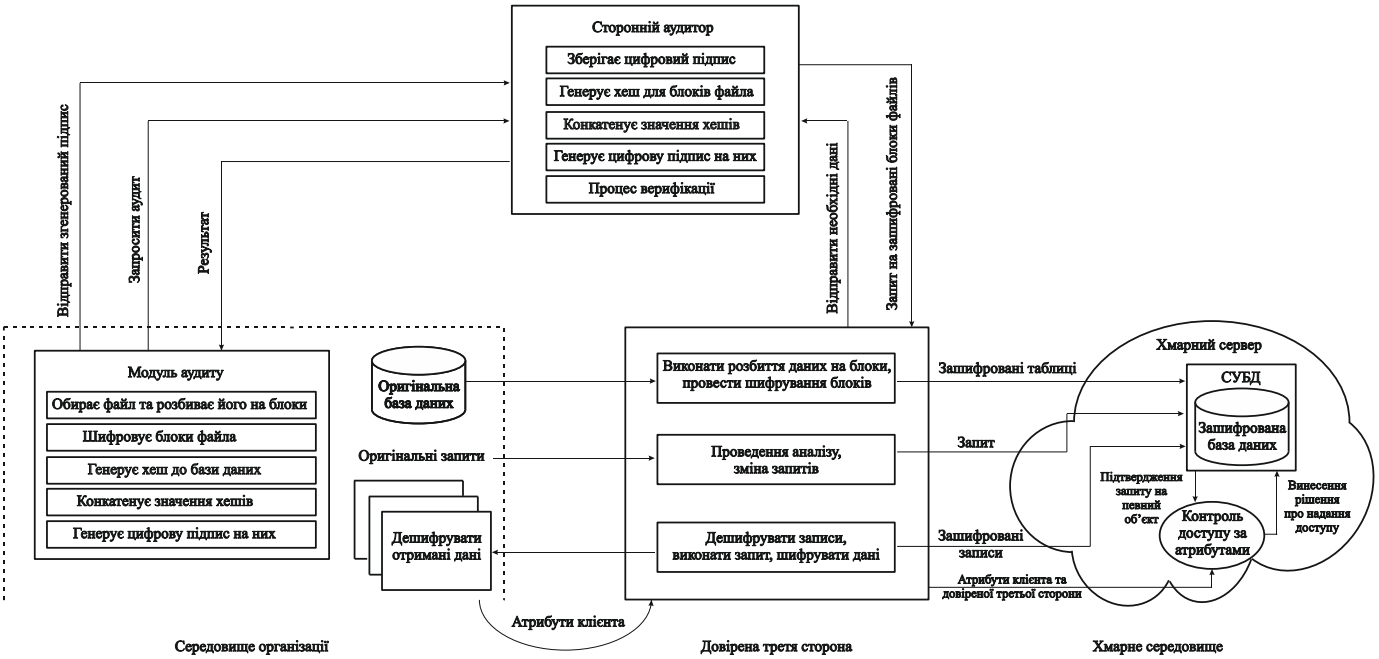



Рисунок 2.3 - Структура системи захисту даних в «хмарному» середовищі

## 2.4 Представлення інформаційних процесів конвергентних обчислювальних мереж підприємства системої масового обслуговування

Інформаційне взаємодія робочих серверів і станцій, як системних так і «хмарних» технологій, можна представити як сукупність потоків відповідей і потоків запитів. Це надає передумову для представлення інформаційних процесів конвергентних обчислювальних мереж підприємства системої масового обслуговування.

Системи масового обслуговування, для аналізу функціонування обчислювального середовища, досить широко застосовується. Системи масового обслуговування застосовується для вирішення задач розрахунку параметрів мережі, які характеризують якість функціонування інформаційно-обчислювальну мережу підприємства, при відомих параметрах потоків заявок на інформаційно-обчислювальну систему, заданих параметрах її компонентів та відомій конфігурації системи. Дана задача розглядається системою масового обслуговування як задача аналізу. До таких типових задач відносяться: ймовірність відмов у обслуговуванні; задачі розрахунку вероятностно-часових параметрів обробки та передачі різних типів заявок; задачі для системи заданої конфігурації; оцінок часу вирішення відповідних задач. Типові задачі синтезу системи масового обслуговування пов'язані: з визначенням пропускної здатності та топології каналів зв'язку; визначенням оптимальної конфігурації інформаційно-очислювальної мережі підприємства; розподілом заявок і резервних потужностей у обчислювальної системі, що надходять між обчислювальними потужностями підприємства. Даний підхід до вирішення задачі дозволяє абстрагуватися від множини параметрів, пов'язаних із відповідними особливостями реалізації конкретних інформаційно-обчислювальних мереж підприємства. Застосовування інформаційно-обчислювальних мережевих моделей, пов'язано із взаємопов'язаними в часі інфомаційних процесів функціонування множини різнорідних інфомаційних мережевих підсистем, з яких складаються складні обчислювальні системи підпиємства до яких відносяться мережі масового обслуговування. Мережі масового

обслуговування є адекватно відобажають моделі: інформаційо-обчислювально мереж; виробничих мереж; локальних мереж; транспортних мереж; мереж зв'язку та передачі даних. Мережі масового обслуговування інтенсивно розширюють області використання. Мережі масового обслуговування: складаються з кінцевого числа вузлів мережі; визначаються характером циркуляції вимог до вузлів; розподілами тривалостей обслуговування у вузлах; характеризуються вхідним потоком відповідних вимог; потребують обслуговування у вузлах. Розімкнені мережі масового обслуговування з експоненційно розподіленим часом обслуговування і вхідним пуассонівським потоком заявок, в яких заявка, що закінчила обслуговуватися, з певною ймовірністю слідувала в інший вузол мережі. Отримані відповідні результати по замкненим мережам масового обслуговування [17], для випадку різних класів вимог отримані результати були узагальнені. Для опису функцій та складу мережі масового обслуговування використовуються функціональні схеми (рис. 2.4).

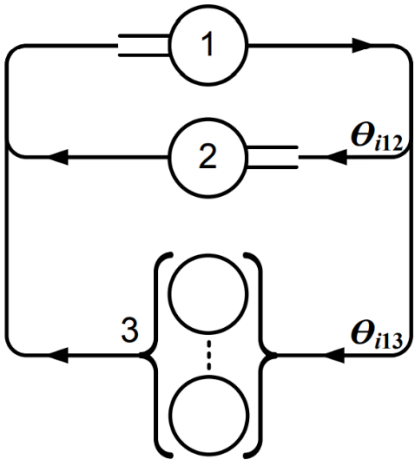



Рисунок 2.4 - Функціональна схема мережі масового обслуговування

Вузол 1 на функціональна схема мережі масового обслуговування (рис. 2.1) відповідає процесору, вузол 2 відповідає (моделює) накопичувач на жорстких дисках системи, вузол 3 - набір пристроїв, через загальну шину із якими здійснюється обмін. Загальна структура представляє циклічний обчислювальний процес, за алгоритмом «процесор – зовнішній накопичувач – процесор». Наведена загальна структура (рис.2.4) до опису мережі масового обслуговування дозволяє

наочно показати відповідні компоненти та їх взаємозв'язок мережі масового обслуговування, проте для аналізу параметрів систем масового обслуговування та подальшого розрахунку має обмеження відповідно до застосовності. У наведеній загальній структурі використовується умовні позначення, які не дозволяють сформувати єдиний підхід до використання мережі масового обслуговування, сформованих, при цьому з різнорідних вузлів, а також поєднати мережу масового обслуговування з інших об'єктами.

У системах масового обслуговування описуються процеси з допомогою графа [11, 15]:

$$G = \{A, Z, R\} \qquad (2.1)$$

де $A$ – множина вершин графа, (стани системи), Z - множина дуг графа (процес перемикання системи з одного стану системи в інший при обслуговуванні чи надходженні), R - матриця суміжності графа.

Для багатокомпонентної, складної системи мережі масового обслуговування виду (2.1) будується шляхом об'єднання структур, що представляють компоненти, які входять до обчислювальної системи. На рис. 2.5 представлений граф мережі масового обслуговування, який, в даному випадку відображає реалізацію двокомпонентного інформаційно-обчислювального процесу.

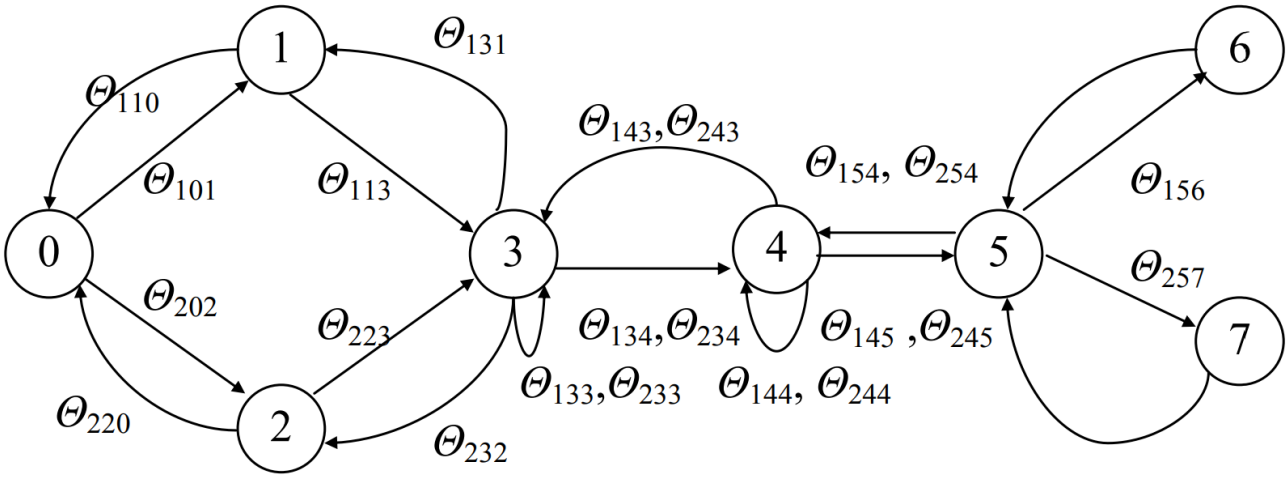



Рисунок 2.5 - Типовий граф мережі масового обслуговування

Перший інформаційний процес, представляє автономний обчислювальний процес 1, закінчується та ініціюється у вузлі 1, другий інформаційний процес – на

автономному підграфі у вузлі 2. Потік заявок з автономного процесу на виконання інформаційно-обчислювальним процесом можуть передаватися, не чекаючи, при цьому відповіді на попередню заявку. Дана ситуація призводить до необхідності введення фіктивного вузла мережі - вузол 0 (джерела заявок), з якого заявки відповідного типу 1 і 2 надходять з інтенсивностями $\lambda_{01}, \lambda_{02}$ 01, і 02, а виконані заявки повертаються в джерело в результаті завершення процесу. Величини $\Theta_{ijk}, i=\overline{1,2},\ j=\overline{0,7},\ k=\overline{0,7}$ задають частоти (ймовірності) переходів між вузлами обчислювальної мережі ($\Theta_{i33}$ та $\Theta_{i44}$ відображають інформаційні процеси повторної передачі повідомлень). Проведений аналіз інформаційної мережі підходами до мереж масового обслуговування дозволяє оцінити затримки у реалізації етапів інформаційно-обчислювального процесу, отримати числові характеристики інформаційно-обчислювальних процесів з урахуванням їхньої взаємодії. Основною властивістю підходів до опису систем масового обслуговування є припущення про відсутність післядії системи потоку обслуговувань і потоку заявок. Це дозволяє сформувати інформаційні процеси зі структурою, подібною до наведеної структури наведеної на рис. 2.5, що значно спрощує підходи, для розрахунку узагальнених характеристик систем масового обслуговування. Суттєвим недоліком представлення інформаційно-обчислювальних процесів із застосуванням систем масового обслуговування, що для визначення потоку обслуговувань у системі та показника потоку заявок необхідно мати аналог системи чи готову систему. У простих варіантах на систему масового обслуговування набирається статистика, за часовими характеристиками потоку заявок. Отримані результати можна використовувати в системах масового обслуговування. Проведення випробувань на натурній системі пов'язане зі значними витратами і в більшості випадків не можливе.

У складних інформаційних системах, до яких відносяться конвергентні обчислювальні мережі підприємства, які включають множину взаємодіючих об'єктів мережі, проведення експериментів проблематично. Представлення інформаційних процесів у даній ситуації вимагає взаємопов'язаного опису значного числа

взаємодіючих інформаційних процесів обробки та передачі інформації ресурсами інформаційно-обчислювальної мережі, а також урахування неоднорідності потоків заявок та динамічного характеру. При проведенні натурних експериментів на інформаційних системах підприємств, які знаходяться в промисловій експлуатації, важко пов'язати характеристики системи загалом та узагальнені показники із показниками окремих компонентів інформаційної системи. Таким чином, виникає проблема з вибором відповідних параметрів компонентів інформаційної системи, що варіюються та впливають на характеристики інформаційної системи в цілому.

Таким чином використання систем масового обслуговування для проектування обчислювального забезпечення інформаційно-обчисювальної мережі підприємств має суттєві обмеження. Для проектування конвергентної обчислювальної мережі підприємства, можуть бути використанні підходи задіяні в мережах масового обслуговування.

## 2.5 Висновки

Поведений аналіз існуючих фреймворків для моделювання загроз в сфері надання «хмарних» послуг, та запропонований кращий з фреймворків за деякими критеріями. На основі моделювання загроз в сфері надання «хмарних» послуг, є можливість ефективно оцінювати та виявляти ризики безпеки в складних розподілени обчислювальних системах, що надапє архітекторам інформаційно-обчислювальних систем пом'якшувати потенційні проблеми на ранніх етапах проектування та розробки систем.

Визначено які з існуючих моделей, протоколів, алгоритмів, має сенс використовувати в сучасних інформаційно-обчислювальних підсистемах підприємства при задіяні «хмарних» ресурсів.

Запропонована структура системи захисту даних в «хмарному» середовищі, надає можливість орієнтуватися в сучасних «хмарних» технологіях побудови конвергентної обчислювальної мережі підприємства.

# 3 СИСТЕМА ЗАБЕЗПЕЧЕННЯ РОЗПОДІЛУ ОБ'ЄКТІВ В ІНФОРМАЦІЙНО - ОБЧИСЛЮВАЛЬНОМУ СЕРЕДОВИЩІ ПІДПРИЄМСТВА

## 3.1 Алгоритм оптимального розподілу інформаційних об'єктів між серверами інформаційно - обчислювальної мережі підприємства та «хмарними» серверами

Розглянуті підходи в мережах масового обслуговування представлення інформаційних процесів у конвергентній інформаційно-обчислювальній мережі підприємства та підхід до оптимального розподілу інформаційних елементів між серверами інформаційно - обчислювальної мережі підприємства та «хмарними» серверами, надає можливість розробити алгоритм функціонування системного програмного забезпечення з розподілу у конвергентній обчислювальній мережі підприємства інформаційно-обчислювальних робіт. Запропонований алгоритм дозволяє: проводити розподіл обчислювальних задач, які перебувають в розподільнику, визначити кількість необхідних машинних ресурсів для реалізації інформаційно-обчислювального процесу для вирішення обчислювальної задачі (рис. 3.1).

Алгоритм оптимального розподілу задач, які представляють сукупність інформаційних об'єктів, дозволяє виконати розподіл обчислювальної задачі $x_k$, $k=\{1,2,...,K\}$, яка поступила від автоматизованого робочого місця $f_g$, $g=\{1,2,...,G\}$ на обчислювальний сервер множини «хмарних» обчислювальних серверів $Cl=\{cl_1,cl_2,...,cl_m\}$ та множини серверів інформаційно-обчислювальної мережі підприємства $L=\{l_1,l_2,...,l_n\}$. Таким чином, $SRV=\{L,Cl\}$ утворює конвергентну обчислювальну мережу підприємства, $j=\{l_1,l_2,...,l_n,cl_1,cl_2,...,cl_m\}$ – загальне число серверів конвергентної обчислювальної мережі, $t$ –інтервал часу опитування серверних станцій. $P$ – комунікаційний сервер інформаційної-

обчислювальної мережі підприємства, $z_{x_k}$ – набір потоків пакетів даних, що відносяться до задачі (інформаційно-обчислювальна робота). Алгоритм складається з наступних кроків:

1. Отримати інформаційно-обчислювальну роботу $x_k$ від автоматизованого робочого місця $f_g$ у виді потоку пакетів даних $z_{x_k}$ по інформаційно-обчислювальній роботі $x_k$.

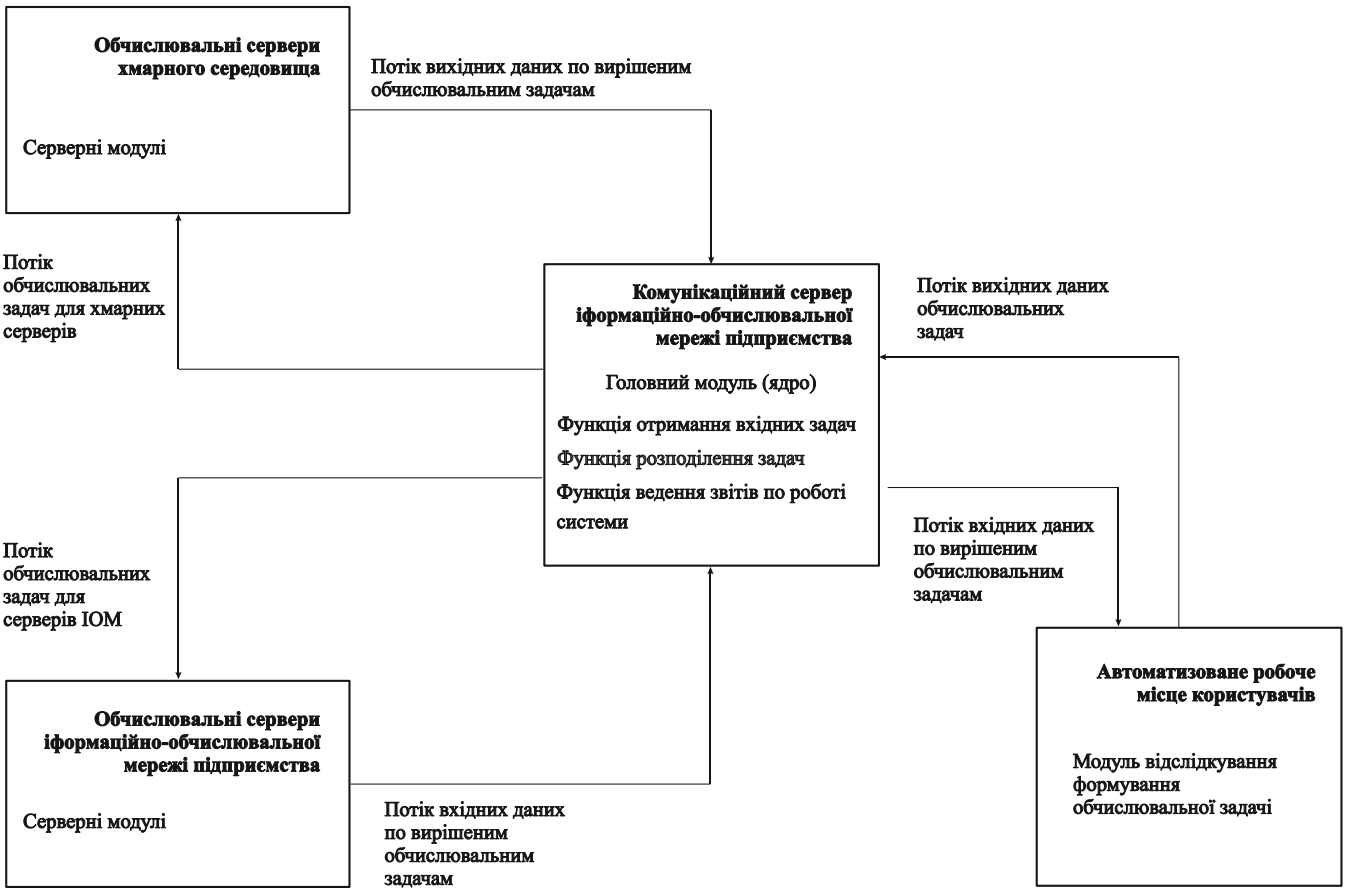



Рисунок 3.1 - Схема функціонування монітора-розподільника у конвергентній обчислювальній мережі підприємства

2. Визначення об’єму обчислювальних ресурсів, необхідних для виконання інформаційно-обчислювальної роботи.

3. Визначення сервера з множини серверів $SRV = \{L, Cl\}$, для яких об’єм оперативно-запам’ятаючого пристрою, обчислювальна потужність процесора, об’єм вільного простору на постійно-запам’ятаючого пристрою швидкість передачі даних по мережі, задовольняють вимогам, необхідним для вирішення інформаційно-

обчислювальної роботи. Для кожного сервера $SRV_j$ перевіряється умова. Якщо сервер відповідає заданим вимогам додаємо сервер у множину доступних, для вирішення інформаційно-обчислювальної роботи серверів $A\{\ \}$, якщо множина серверів $A\{\ \}$ порожня, то перехід до п. 4, інакше до п. 5.

4. Очікування інтервалу часу *t*. Перехід до п.3.

5. Вибір сервера $a_{ms}$ з множини серверів $A\{\ \}$, для якого прогнозована часова оцінка часу виконання інформаційно-обчислювальної роботи мінімальна.

6. Отримання IP - адреси сервера $a_{ms}$.

7. Зміна адреси IP - адреси одержувача в $z_{x_k}$ – набору пакетів даних на IP-адрес сервера $a_{ms}$.

8. Зміна IP - адреси відправника в $z_{x_k}$ – набору пакетів даних на IP- адрес комунікаційного сервера *P*.

9. Відправка інформаційно-обчислювальної роботи $z_{x_k}$ на сервер $a_{ms}$.

10. Додавання інформації про відправлену задачу в журнал ведення логів: комп'ютер одержувача, комп'ютер відправника, найменування задачі.

11. Очікування надходження відповідних даних про виконання задачі $z_{x_k}$ – набору пакетів даних від сервера $a_{ms}$.

12. Якщо дані від сервера $a_{ms}$ надійшли, то перехід до п. 13, інакше перехід до п. 11.

13. Зміна IP - адреси відправника в $z_{x_k}$ – набору пакетів даних на IP- адрес комунікаційного сервера *P*.

14. Зміна адреси IP - адреси в $z_{x_k}$ – набору пакетів даних на IP- адрес одержувача $f_g$.

15. Надіслати отримані дані на одержувача $f_g$.

16. Перехід до п. 1.

Алгоритм (рис.3.2), дозволяє визначити трудомісткість обчислювальних задач, заснований на регресійному аналізі, виходячи з наперед визначених класів інформаційно-обчислювальних робіт, відібраних на етапі збору статистичних даних. Для визначення необхідних для реалізації обчислювальної роботи вільних ресурсів сервера та обчислювальної складності задачі виконуються етапи:

1. Формування правила щодо залежності показників. У разі збільшення об’ємів доступних обчислювальних потужностей обчислювальної системи зменшується, при цьому, час виконання завдання.

2. Визначення, при проведенні інформаційно-обчислювальних робіт, залежних змінних, значення яких можуть бути змінені під час виконання наведеного алгоритму. До таких змінних відносяться: максимальна пропускна здатність мережного каналу, обчислювальна потужність центрального процесора, загальний об’єм оперативно-запамятовуючого пристрою. До незалежних змінних відносяться - доступним на момент надходження задачі інформація по вільним ресурсам: вільний об’єм оперативно-запамятовуючого пристрою, практична швидкість передачі даних через мережу, об’єм доступних операцій в одиницю часу на центральному процесорі.

3. Етап збору статистичних даних формує таблиці залежностей часу виконання інформаційно-обчислювальної роботи від кількості доступних ресурсів оперативно-запамятовуючого пристрою, швидкості передачі даних через мережу, центрального процесора.

4. За результатами формування списку залежностей часу виконання інформаційно-обчислювальної роботи виконується побудова функції.

5. Проведення розрахунку достовірності та точності отриманих результатів.

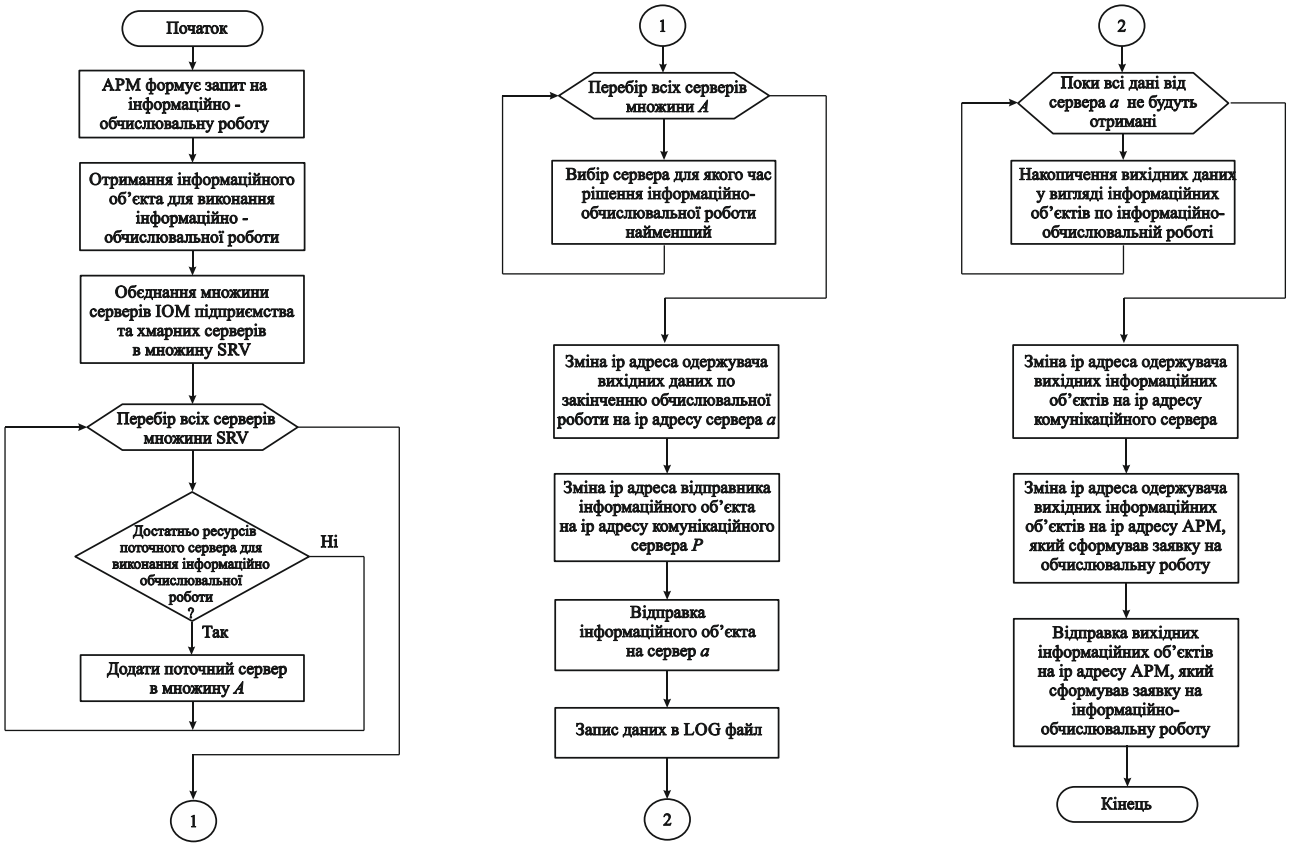



Рисунок 3.2 - Алгоритм розподілу інформаційно-обчислювальних робіт у конвергентній обчислювальній мережі підприємства

## 3.2 Структура модулів системного програмного забезпечення розподілу об'єктів між серверами інформаційно - обчислювальної мережі підприємства та «хмарними» серверами

На основі запропонованих алгоритмів розподілу інформаційних об'єктів у конвергентних обчислювальних мережах підприємства розроблено відповідне системне програмне забезпечення монітор-розподільник іформаційно-обчислювальних робіт, дозволяє реалізувати розподіл інформаційно-обчислювальних задач у конвергентній мережі підприємства.

На етапі проектування монітора-розподільника інформаційних об'єктів до системного програмного забезпечення пред'явленні наступні функціональні вимоги: забезпечити ручне управління додаванням у область видимості монітора-

розподільника, серверів на всіх етапах роботи конвергентній мережі підприємства; можливість розподілу потоку даних у мережах з будь-яким числом обчислювальних серверів; на кожному сервері, що використовується в обчислювальній системі підприємства, можливість надавати звіти щодо кількості виконаних обчислювальних задач; можливість реалізації, з метою введення даних, функцій ручного контролю вартості інформаційних обчислень на кожному сервері системи.

Для розподілу у конвергентній обчислювальній мережі підприємства інформаційно-обчислювальних робіт визначені нефункціональні вимоги: можливість роботи Windows служб у вигляді допоміжних компонентів; підтримка платформ Windows; для отримання даних про продуктивність сервера підтримка WIN API MPI.

Монітор-розподільник - системне програмне забезпечення, побудоване на основі наступних модулів: ядро (головний модуль); серверний модуль; клієнтський (автоматизоване робоче місце) модуль. Головний модуль монітора-розподільника навантаження розміщується на комунікаційному сервері інформаційно-обчислювальної мережі (рис. 3.3).

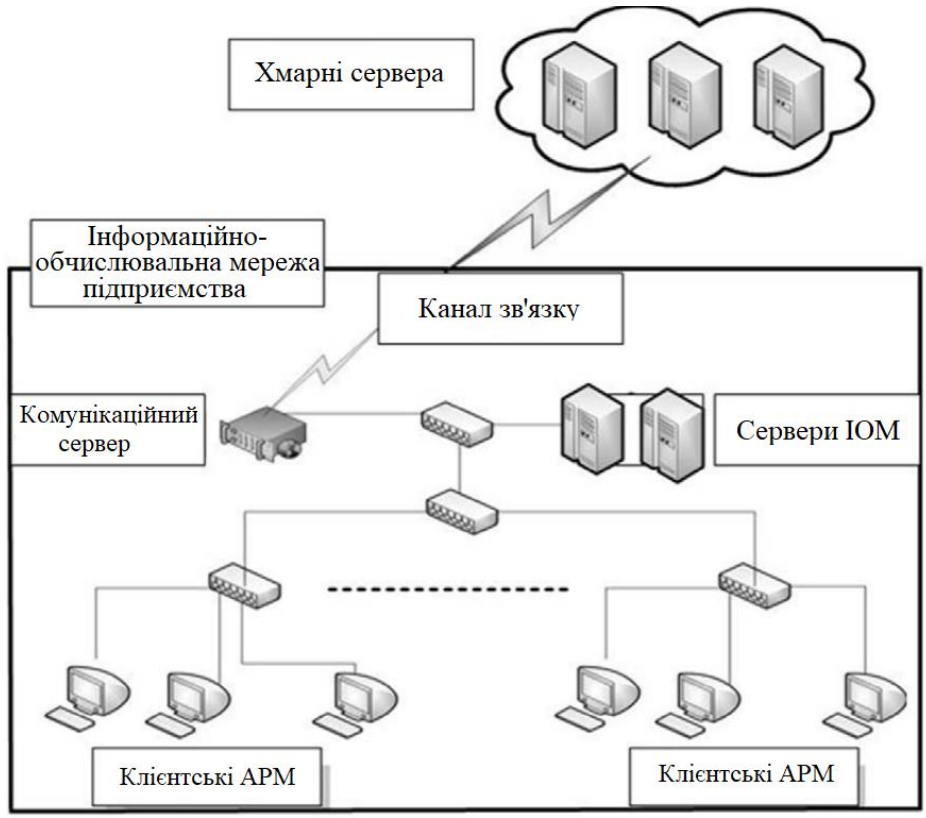



Рисунок 3.3 - Структура конвергентної обчислювальної мережі підприємства

Допоміжні модулі виконують функцію відстеження надходження інформаційно-обчислювальних задач та встановлюються на робочих станціях інформаційно-обчислювальної мережі підприємства. Допоміжні модулі на серверах «хмарного» середовища та обчислювальних серверах інформаційної мережі підприємства встановлюються з метою відстеження часу надходження задачі, повернення вихідного потоку даних за вирішеною задачею, отримання інформації про вільні обчислювальні ресурси конвергентної обчислювальної мережі.

Монітор-розподільник проектований та реалізований модульному принципу побудови архітектури (рис. 3.4). Модульний принцип дозволяє організувати найшвидші терміни проектування, розробки та обслуговування системного програмного забезпечення.

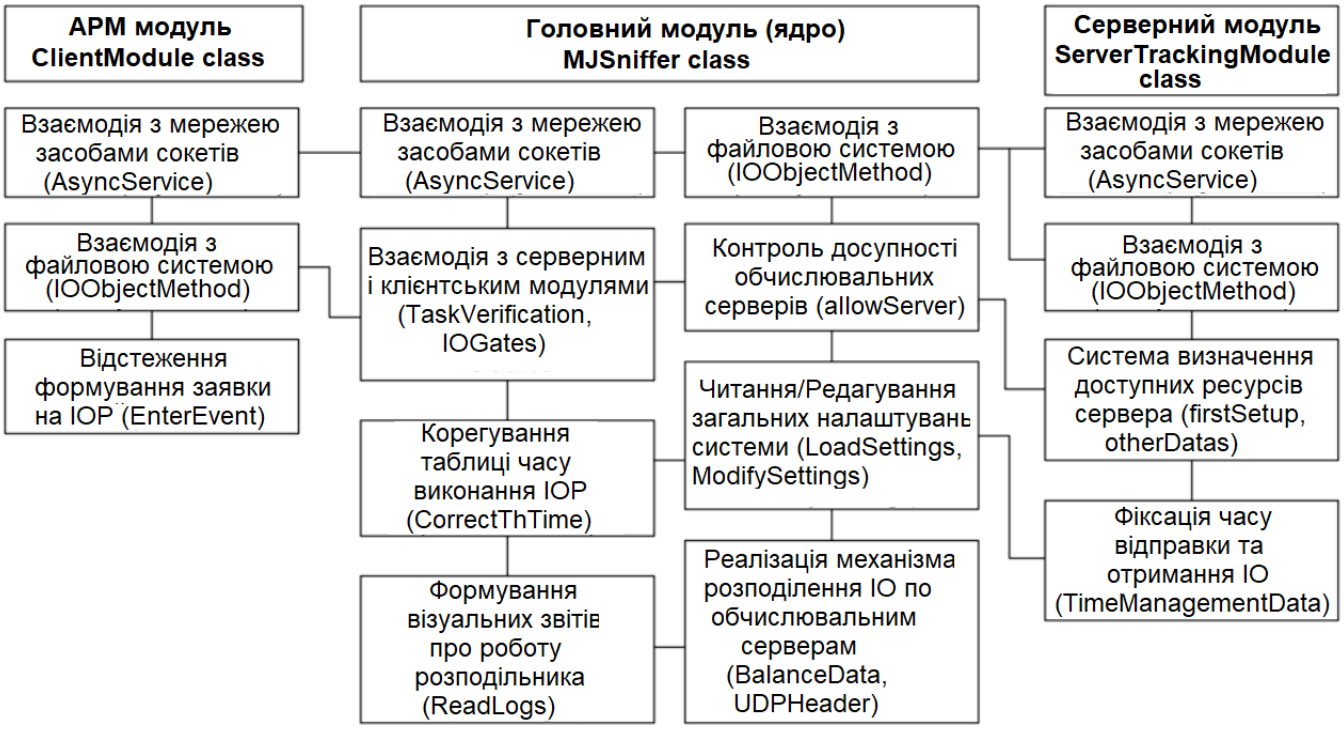



Рисунок 3.4 - Модулі та функції монітора-розподільника інформаційно-обчислювальних робіт

Загальна схема роботи, відповідно до алгоритму розподілу потоку даних, головного модуля (рис. 3.5) відображає основні етапи та особливості роботи головного модуля розподільника інформаційни об’єктів.

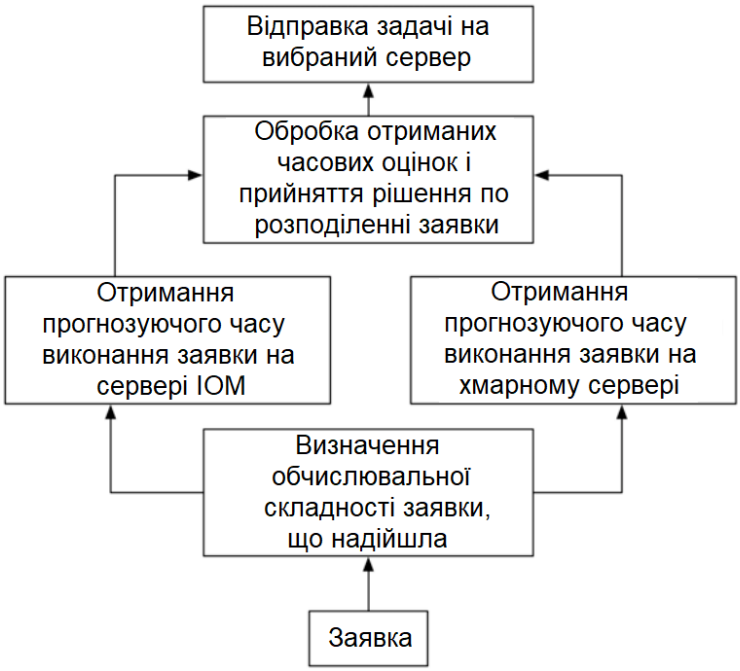



Рисунок 3.5 - Принцип роботи ядра монітора-розподільника при надходженні інформаційно-обчислювальної роботи

Процес взаємодії клієнтських модулів і головного модуля, які розміщуються на автоматизованих робочих місцях співробітників інформаційно-обчислювальної мережі підприємства, представляє двонаправлений процес обробки потоку даних, головний модуль є інтерфейсом, виконує необхідні перетворення, виконує подальше пересилання потоку даних, приймає дані від клієнтського модуля, клієнтський модуль є ініціатором потоку інформаційно-обчислювальних задач. Потік обчислювальних даних є двонаправленим, який проходить через головний модуль (рис. 3.6). Двоспрямованість потоків обчислювальних даних - необхідна умова для роботи відповідних функцій головного модуля: отримання оцінок часу передачі потоку даних по мережі підприємства між автоматизованими робочими станціями і серверними станціями; для визначення коефіцієнта відхилення, побудова порівняльних таблиць розрахункового часу вирішення інформаційно-обчислювальних задач від практично досягнутого; збирання статистики обчислювальних задач на серверах за часом обробки; перерозподілу інформаційно-обчислювальних задач у разі виходу з ладу обчислювальних серверів.

Ядро системи взаємодіє з клієнтськими модулями обчислювальної системи, розташованими на серверах інформаційної мережі при виникненні наступних подій: необхідно встановити значення швидкості передачі потоку даних через мережу підприємства; виникнення інформаційно-обчислювальної роботи на автоматизованому робочому місці; необхідно визначити об’єми вільних обчислювальних ресурсів на обчислювальному сервері мережі.

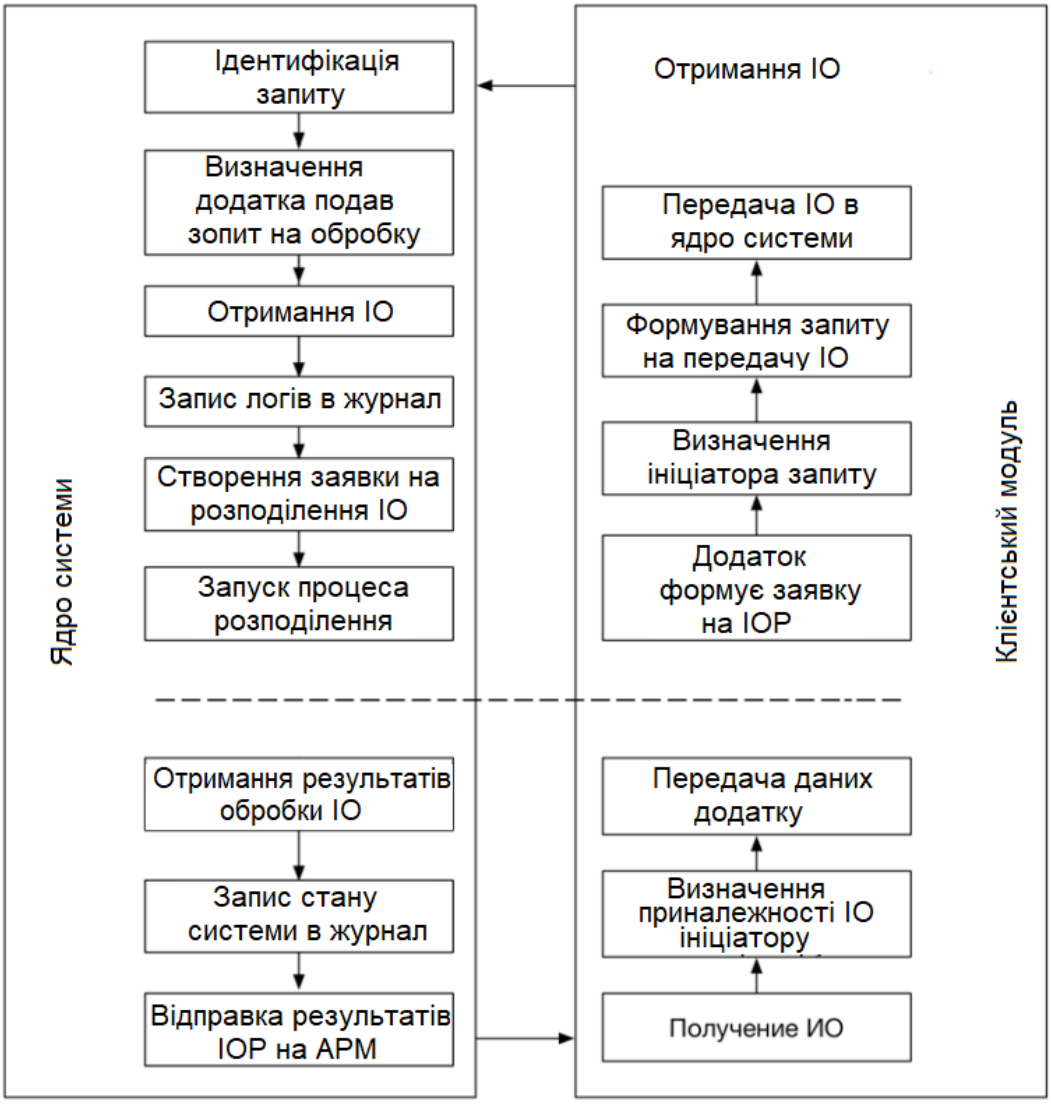



Рисунок 3.6 - Схема взаємодії клієнтського модуля та ядра монітора-розподільника при надходженні інформаційно-обчислювальної роботи

У разі інформаційно-обчислювальної роботи на автоматизованому робочому місці, задача передається на комунікаційний сервер мережі підприємства, відбувається процес ідентифікації обчислювальної задачі та передачі її на обраний обчислювальний сервер мережі підприємства.

На етапі ідентифікації обчислювальної задачі відбувається визначення класу складності інформаційно-обчислювальної роботи, після чого проводиться запис вироблених дій в журнал логів і передача обчислювальної задачі безпосередньому обчислювальному серверу в конвергентній обчислювальній мережі підприємства (виконавцю).

Визначення належності обчислювальної задачі до одного з існуючих класів та процес ідентифікації є функцією, яка складається з наступних основних блоків дій: визначення складності основного класу, до якого належить обчислювальної задачі, яка надійшла; визначення програми – ініціатора обчислювальної задачі; визначення теоретичного рівня часу виконання обчислювальної задачі; визначення рівня складності обчислювальної задачі усередині знайденого класу; вибір множини серверів мережі, які задовольняють вимогам інформаційно-обчислювальним роботам за розміром оперативно-запамятовуючого пристрою; швидкістю передачі даних через мережу; вільними ресурсами центрального процесора; коригування часу виконання обчислювальної задачі з урахуванням порівняльної таблиці практичних часових оцінок; знаходження компромісного рішення щодо вартості інформаційно-обчислювальних робіт, часу виконання інформаційно-обчислювальних робіт, залежно від обраної стратегії розподілу; складання таблиці часових оцінок виконання інформаційно-обчислювальної завдачі на кожному обчислювальному сервері мережі підприємства; складання таблиці вартості виконання інформаційно-обчислювальної завдачі на кожному обчислювальному сервері мережі підприємства.

Визначення доступних обчислювальних ресурсів, при настанні події, відбувається опитування сервера обчислювальної мережі, з метою визначення:
кількості вільних тактів центрального процесора, вільного об’єму пам’яті оперативно-запам’ятаючого пристрою.

При настанні відповідної події визначення швидкості передачі потоку даних по обчислювальній мережі між обчислювальним сервером та головним модулем системи відбувається процес передачі файла з метою визначення практичної швидкості передачі потоку даних через обчислювальну мережу. Використання

програмного забезпечення для розподілу навантаження передбачає два етапи: етап експлуатації системи; етап введення в експлуатацію.

Етап введення в експлуатацію обчислювальної системи включає модульну установку програмного забезпечення на комунікаційний сервер мережі підприємства, обчислювальні сервери, автоматизованні робочі місця інформаційно-обчислювальної мережі, обчислювальні сервери «хмарного» середовища. Після завершення виконання даної операції, відбувається процес налаштування обчислювальної системи для складання розрахункових теоретичних таблиць часу виконання інформаційно-обчислювальних робіт обчислювальної мережі підприємства на підставі часу, отриманого при виконанні обчислювальних робіти, що формуються в інформаційно-обчислювальній мережі підприємства. Завдання інформаційно-обчислювальної мережі підприємства для складання розрахункової теоретичної таблиці вибираються за правилами: хоча б одна обчислювальна робота, що використовується в інформаційно-обчислювальної мережі підприємства, з кожного програмного продукту, має бути включено до теоретичної таблиці розрахунку часу виконання інформаційно-обчислювальної роботи; завдання не повинні дублюватися; у разі значного розбіжності часу, що генеруються одним програмним забезпеченням, витраченого на вирішення однотипних обчислювальних задач, у таблицю розміщуються всі класи знайдені для одного програмного забезпечення, задачі поділяються на відповідні класи; обчислювальні задачі не використовуються, час на вирішення яких не вимагає серверних обчислювальних ресурсів.

Коригування часу вирішення обчислювальної задачі $q_i$ виконується, у випадку, якщо $t_{qi}$ може бути знайдено в розрахунковій таблиці $t_{pr}$. Коригування часу вирішення обчислювальної задачі виконується з метою отримання, за часом виконання, найбільш точних даних іннформаційно-обчислювальних робіт. Можуть бути використанні різні підходи, для налаштування управління монітором-розподільником: виконання всіх можливих обчислювальних робіт завдань за мінімально можливі витрати на експлуатацію конвергентної обчислювальної мережі підприємства; виконання всіх обчислювальних задач мережі за мінімально

можливий час; комбіновані підходи за вартості обчислень, та часом вирішення, що проводяться; виконання всіх обчислювальних робіт при заданому порозі витрачання коштів на зберігання даних та використання зовнішніх джерел обробки.

Часова оцінка розрахунку виконання інформаційно- обчислювальних робіт проводиться за допомогою виконання задач на кожному обчислювальному сервері мережі, який може бути задіяний для вирішення інформаційної задачі у конвергентної обчислювальної мережі підприємства.

При складанні розрахункових таблиці використовуваних ресурсів і таблиці виконання обчислювальних задач алгоритм оперує даними:

- множини задач $X = \{x_1, x_2, ..., x_k\}$;

- множини класів задач $Q = \{q_1, q_2, ..., q_\beta\}$;

- множини програмного забезпечення $S = \{s_1, s_2, ..., s_n\}$;

- множини обчислювальних серверів інформаційно-обчислювальної мережі підприємства $L = \{l_1, l_2, ..., l_n\}$;

-множини віртуальних серверів «хмарного» середовища $Cl = \{cl_1, cl_2, ..., cl_m\}$.

В результаті виконання процедури налаштування роботи обчислювальної системи, крім теоретичних таблиць часових оцінок, створюється теоретична таблиця використання ресурсів інформаційно-обчислювального сервера мережі підприємства по кожному обчислювальному елементу. На підставі використання таблиці ресурсів на етапі розподілу інформаційно-обчислювальних робіт вибирається допустима підмножина серверів обчислювальної мережі, які можуть виконати обчислювальну роботу з урахуванням заданих критеріїв. В табл. 3.1 наведено співвідношення класів складності інформаційно-обчислювальних робіт та використовуваних ресурсів.

Після вирішення інформаційно-обчислювальної роботи ресурси обчислювального сервера мережі вивільняються, внаслідок при надходженні нової інформаційно-обчислювальної роботи відбувається опитування всіх обчислювальних серверів мережі з метою формування підмножини доступних

обчислювальних серверів для вирішення інформаційно-обчислювальної роботи, яка надійшла.

Таблиця 3.1 - Співвідношення класів складності інформаційно-обчислювальних робіт та використовуваних ресурсів

| Номер задачі | ПЗ | Клас задачі | ОЗП (мб) | ЦП (FLOPS) | Розмір переданих даних (мб) |
|---|---|---|---|---|---|
| $x_1$ | $s_1$ | 1 | $m_{1,1}$ | $c_{1,2}$ | $d_{1,3}$ |
| $x_2$ | $s_1$ | 2 | $m_{2,1}$ | $c_{2,2}$ | $d_{2,3}$ |
| … | … | … | … | … | … |
| … | $s_1$ | $q_1$ | … | … | … |
| … | $s_2$ | 1 | … | … | … |
| … | $s_2$ | 2 | … | … | … |
| … | … | … | … | … | … |
| … | $s_2$ | $q_2$ | … | … | … |
| … | $s_a$ | 1 | … | … | … |
| … | $s_a$ | 2 | … | … | … |
| … | … | … | … | … | … |
| $x_k$ | $s_a$ | $q_\beta$ | $m_{\mu,1}$ | $c_{\pi,2}$ | $d_{\xi,3}$ |

Даний етап є першим етапом роботи системного монітора-розподільника в режимі експлуатації конвергентної обчислювальної мережі підприємства.

Монітор-розподільник інформаційно-обчислювальних робіт розроблявся з використанням ОС Microsoft Windows 10, середовище MS Visual Studio 2019 з використанням мов програмування C++ і C#. Вибір мови системи програмування C++ був зумовлений наступними критеріями:

- швидкість роботи системного програмного забезпечення, написаних за допомогою мови програмування C++;

- можливість працювати безпосередньо з файловою системою, потоками даних, що передаються через обчислювальну мережу з використанням сокетів;

- можливість працювати безпосередньо з ділянками пам'яті оперативно-запам'ятовучої пам'яті комп'ютера;

- наявність достатньо великої кількості відлагоджених бібліотек, дозволяють розширити можливості мови програмування без зниження загальної швидкості роботи обчислювальної системи;

- можливість низькорівневої роботи з файловою системою, потоками даними, що передаються з використанням стека протоколів TCP/IP.

Використання мови програмування C# у проекті обумовлено необхідністю, для надання адміністратору, застосування візуального середовища доступу до графічного меню управління та моніторинг роботи обчислювальної системи розподілу інформаційно-обчислювальних робіт. У проект, написаний з використанням мови програмування C#, можуть бути впроваджені бібліотеки та модулі, написані з використанням мови програмування C++, що в даній ситуації дозволяє поєднати швидкість роботи програмного забезпечення з достатньо великими можливостями візуалізації процесів обчислювальної мережі, що відбуваються в ядрі монітора-розподільника, із застосуванням сучасного середовища розробки програмних продуктів MS Visual Studio 2019. Середовище програмного продукту MS Visual Studio 2019 дозволяє розробляти програмні системи використовуючи мови програмування C#, C та C++ надаючи сучасні засоби проектування та побудови візуального інтерфейсу WPF (Windows presentation foundation). Перераховане середовище розробки надає можливість проектувати та розробляти сучасне системне програмне забезпечення, допрацьовувати, компілювати використовуючи різні версії вбудованих бібліотек, виробляти його рефакторинг, що є основним при виборі середовища розробки програмних продуктів, в даному випадку для проектування та розробки монітора-розподільника інформаційно-обчислювальних робіт.

Готове системне рішення монітора-розподільника складається з серверного модуля, клієнтського модуля та ядра (головного модуля). Ядро (головний модуль) реалізує алгоритм розподілу обчислювальних даних між обчислювальними серверами та алгоритм контролю автоматизованого робочого місця та серверного

модуля. Обчислювальний серверний модуль встановлюється на кожен сервер інформаційно-обчислювальної мережі підприємства та на обчислювальні сервери «хмарного» середовища. У налаштуваннях головного модуля монітора-розподільника заносяться дані про IP адреси обчислювальної мережі, призначені обчислювальним серверам, на яких, в свою чергу, встановлюються серверні модулі. На кожному автоматизованому робочому місці в інформаційно-обчислювальній мережі встановлюється клієнтський модуль.

Монітор-розподільник дозволяє, тільки для додатків, побудованих на основі архітектури клієнт-сервер, проводити розподіл обчислювального навантаження (рис. 3.7), де можливе розмежування клієнтської частини - виконує роль інтерфейса, дозволяє користувачеві обчислювальної системи взаємодіяти з системою засобами графічного інтерфейса, серверної частини - виконує роль обробника потоків запитів. Монітор-розподільник інформаційно-обчислювальних робіт TiCloudBalancer реалізований на основі проекту з використанням середовища розробки Visual Studio 2019 з відкритим вихідним кодом Balancer, при розробці якого використовувалася мова програмування Csharp.

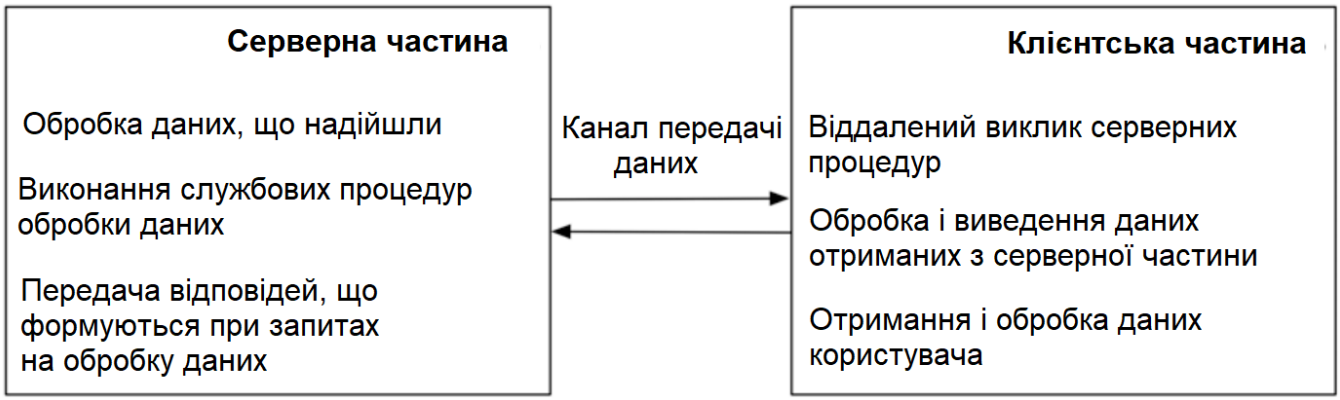



Рисунок 3.7 - Структура парограмного забезпечення розподільника обчислювального навантаження

Обчислювальна система побудована з використанням технології об'єктно-орієнтованого підходу до написання системних модулів. Кожен модуль обчислювальної системи є закінченим програмним продуктом (рис. 3.8).

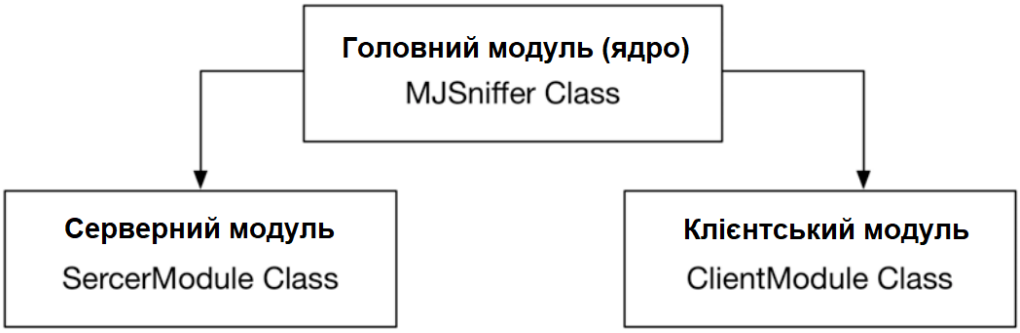



Рисунок 3.8 - Структура системного програмного забезпечення монітора-розподільника навантаження TiCloudBalancer

Серверний модуль монітора-розподільника навантаження – системне програмне забезпечення, працює як служба Windows, реалізований класом ServerTrakingModule і складається з наступних основних публічних методів:

- метод OnStart – завантаження налаштувань системи монітора-розподільника. Завантаження налаштувань з конфігураційного файлу, запуск розподільника з необхідним ключами, якщо ключі встановлені в системному реєстрі Windows;

- метод firstSetup – отримання інформаційних даних про ресурси обчислювального сервера на момент налаштування інформаційно-обчислювальної системи;

- метод onStop – вивантаження розподільника системної служби та зупинення роботи обчислювальної системи;

- метод findInfo – оболонка монітора-розподільника над WMI функціями, дозволяє отримати та візуліазувати інформацію про ресурси обчислювальної системи у потрібному форматі;

- метод otherDatas – отримання інформаційних даних про поточні доступні ресурси обчислювального сервера.

Отримання необхідних даних на поточний час відбувається шляхом виклику приватного методу з параметрами private static void findInfo(string WinClass, string info), де WinClass – посилання на інформаційний об'єкт, у якому містяться необхідні дані про системні ресурси обчислювальної системи, параметр info – найменування інформаційних ресурсів, які потрібно отримати. Функція може візуалізувати дані в

режимі налаштування в консоль, для перегляду користувачем або віддавати дані комунікаційному системному програмному забезпеченню, що викликав метод класу ServerTrakingModule. Системний модуль дозволяє надавати відповідну інформацію за запитом зовнішнього системного програмного забезпечення за допомогою передачі та прийому даних з використанням відповідного відкритого порту. Модулі системного програмного забезпечення дозволяють передавати та приймати потік даних за допомогою використання методів, що дозволяють надсилати та прослуховувати дані. Для передачі даних по обчислювальній мережі використовуються асинхронні клієнт-серверні запити.

Під час розподілення інформаційних об’єктів (TiCloudBalancer) модулі обчислювальної мережі обмінюються даними використовуючи головний модуль. Головний модуль обчислювальної системи після запуску на виконання та проходження процесу налаштування системи створює відповідний файл налаштувань main_config.xml, в якому міститься необхідна інформація налаштування обчислювально системи підприємства, які задаються системним адміністратором мережі. Необхідна інформація з конфігураційних файлів обчислювальної мережі завантажується в систему при її старті, при необхідності налаштування можуть бути змінені з використанням графічного інтерфейсу на головній сторінці системи в процесі роботи.

Робота з конвергентною обчислювальною мережею підприємства на стороні головного модуля реалізується за допомогою реалізації методу AsyncService, який запускає відповідні служби прослуховування по певному відкритому порту з метою обміну інформацією між обчислювальними серверами мережі та ядром, між головним та клієнтським модулями.

Для опитування системних програмних модулів, розташованих на обчислювальних серверах мережі, головний модуль обчислювальної системи використовує метод SendDataToClient, який, в свою чергу, генерує команду команду getSystemResource з метою отримання інформації у відповідь в форматі JSON про вільні ресурси обчислювальної системи. Для отримання інформації про характеристики системи використовується метод getAllSystemResource. дозволяє.

Швидкість передачі інформації інформаційно-обчислювальною мережею між сервером, який обробляє запити і комунікаційним системним забезпеченням, здійснюється з використанням методу getUploadNetSpeed. Під час отримання на серверній частині потоку даних, відправлення яких ініційовано функцією getUploadNetSpeed, фіксується час отримання інформації, також фіксується час відправлення даних, після їх відправлення. В результаті виконаних операцій проводиться розрахунок швидкості отримання та відправлення потоку даних між двома обчислювальними серверами на підставі отриманих результатів значень часу прийому та відправлення потоку даних (рис. 3.9).

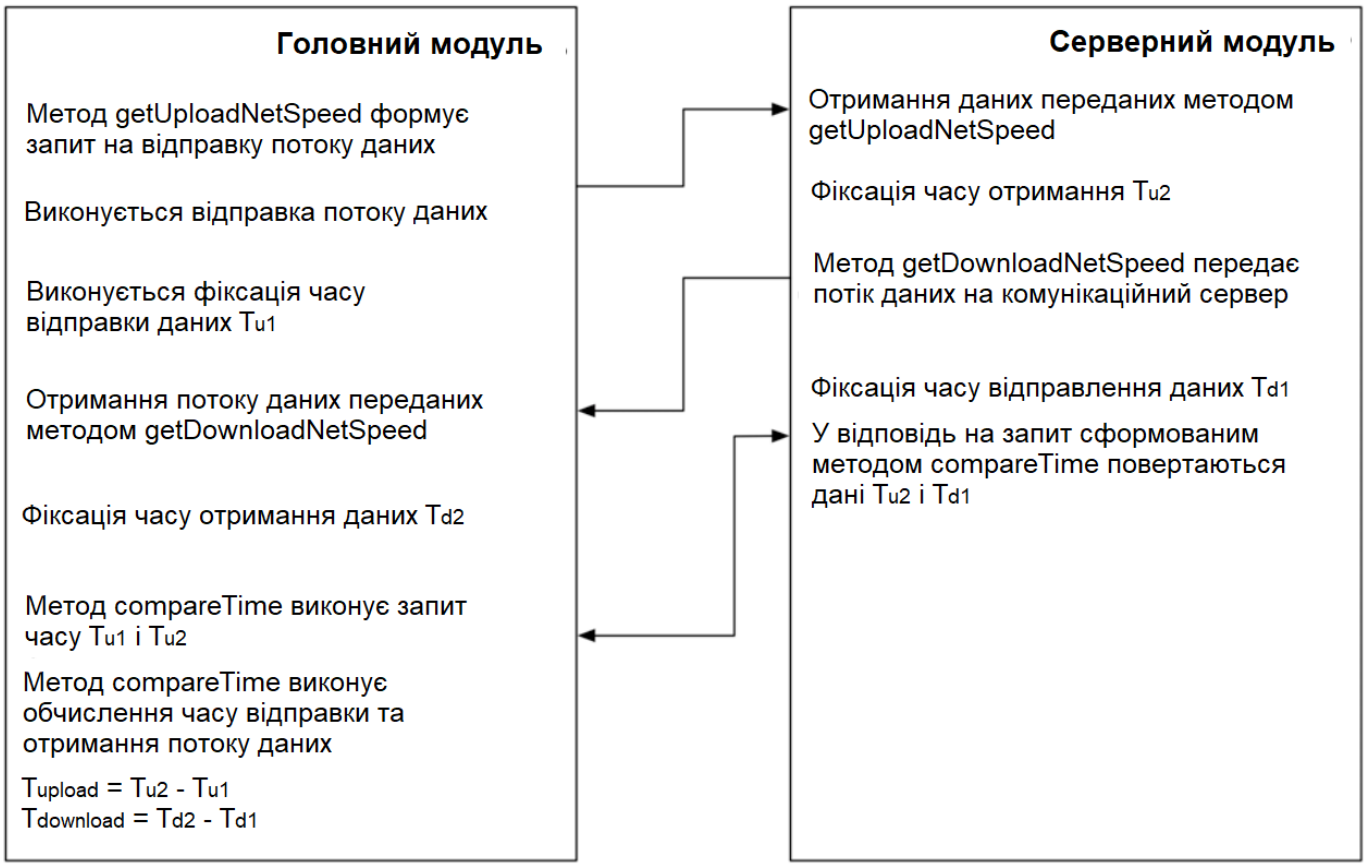



Рисунок 3.9 - Розрахунок швидкості приймання/передачі потоку даних між комунікаційним системним забезпеченням та обчислювальними серверами

При надходженні інформаційно-обчислювальної роботи викликається функція taskBalancer, яка надає можливість вибору вибір оптимального обчислювального сервера для обробки потоку даних з точки зору використовуваної політики розподілу інформаційних об’єктів, яка переглядає розрахункові таблиці з

розрахунковими зведеними даними за часом виконання інформаційно-обчислювальної роботи. Відповідні дані за часом виконання зберігаються в оперативно-запам’ятовуючому пристрої для швидкого доступу та обробки.

Процес вибору оптимального обчислювального сервера мережі складається з двох етапів. На першому етапі відбувається вибір підмножини обчислювальних серверних станцій мережі, які здатні обробити заявку, що надійшла. Перелік обчислювальних серверів, придатних для вирішення задачі, що надійшла, заносяться в apppropriateServers (асоціативний масив), елементи якого відсортовані по числу вільних ресурсів каналу передачі даних, оперативно-запамятаючого пристрою, центрального процесора, за зменшенням. На другому етапі відбувається вибір оптимальних обчислювальних серверів з урахуванням економічних вимог щодо вирішення інформаційно-обчислювальної роботи: якщо необхідна мінімізація витрат по обчислювальним серверам мережі, в даному випадку вибирається обчислювальний сервер з з найменшим значенням ключа cash (найменшою вартістю проведення обчислень: якщо необхідна мінімізація часу виконання інформаційно-обчислювальної роботи без урахування, при цьому вартості проведення обчислень, то ключ масиву apppropriateServers cash не враховується, вибирається, в даному випадку перший елемент з масиву apppropriateServers.

Після визначення, для проведення розрахунку задачі, ID обчислювального сервера, відбувається визначення IP адреси в обчислювальній мережі підприємства по ID зі списку придатних серверів, які зберігаються в allowServer, модифікація IP адреси одержувача в пакеті потоку даних, відправка даних з використанням функції sendDataToServer.

Для визначення, що витрачається на виконання обчислювальної роботи і пересилання даних, практичного часу, проводиться запис в лог- файл з процедури saveLog, яка проводить запис рядка з параметрами та запис про подію в асоціативний масив allTask: IP адреса сервера, що надіслав запит; час надходження заявки; IP адреса обчислювального сервера, що обробив задачу; теоретичний час обробки обчислювальної роботи; практичний час, за який була опрацьована інформаційно-обчислювальна робота; ID обчислювальної задачі із таблиці завдань.

Клієнтський модуль системне забезпечення, яке є службою, при отриманні потоку даних від інформаційно-обчислювальної системи мережі, передає їх комунікаційному обчислювальному серверу для подальшого розподілу. Список клієнтських частин системи, дані по яких мають бути передані на обчислювальний комунікаційний сервер, зберігаються у форматі xml в конфігураційному файлі app_config.xml.

На етапі реалізації підключення до серверної обчислювальної системи використовується IP адреса та номер порту, IP адреса та номер порту підключення прописуються адміністратором мережі підприємства у конфігураційному файлі налаштувань модулів обчислювальної системи мережі. Для модуля клієнта з налаштуваннями обчислювальної мережі підприємства призначений формат файлу net_config.xml. Для зміни налаштувань клієнтської частини необхідно змінювати вміст конфігураційного файла net_config.xml, після чого необхідно перезапустити клієнтську службу, для того щоб служба отримала нові налаштування та продовжила роботу з інформаційно-обчислювальною системою підприємства. Для надсилання даних із клієнтської обчислювальної системи на головний модуль інформаційної системи використовується функція SendDataToServer, яка відправляє по отриманному IP адресу з конфігураційного файлу, необхідний набір потоку даних на комунікаційний обчислювальний сервер. Клієнтський обчислювальний модуль отримує з файлу app_config.xml відомості про IP адреси, номера портів, за наявності активності на яких включається модуль клієнта, інформуючи головний модуль системи підприємства про формування нової інформаційно- обчислювальної роботи з використанням функції startActivity.

Інформаційно-обчислювальна мережа підприємства є закритою від зовнішніх впливів системою, яка функціонує і налаштована за правилами, закладеними при проектуванні обчислювальної мережі підприємства. Інтеграція модуля монітора-розподільника інформаційно-обчислювальної роботи в інформаційній мережі підприємства проводиться у кілька етапів. На етапі відповідності вимогам щодо встановлення модуля монітора-розподільника обчислювальних робіт за допомогою перевірки програмної та апаратної бази з боку адміністратора інформаційної мережі

підприємства. Для установки інформаційно-обчислювальної системи необхідно щоб інформаційно-обчислювальна мережа функціонувала на основі технології ОС Windows, автоматизоване робоче місце, комунікаційний сервер та обчислювальні сервери, що входять до складу інформаційно-обчислювальної мережі підприємства, повинні працювати під управлінням ОС Windows, Server edition 2018 та вище. Автоматизовані робочі місця інформаційно-обчислювальної мережі підприємства повинні мати єдиний доступ до Інтернет мережі. Системне програмне забезпечення, для якого виконується балансування обчислювального навантаження, повинно мати обчислювальну серверну частину, встановлену на кожен сервер «хмарного» середовища і на кожен обчислювальний сервер інформаційно-обчислювальної мережі підприємства, мати клієнтську частину, встановлену на автоматизованих робочих місцях користувачів інформаційної мережі та архітектуру клієнт-сервер.

На етапі інтеграції системного програмного забезпечення розподілу об'єктів у конвергентну обчислювальну мережу підприємства відбувається встановлення дистрибутивів модулів монітора-розподільника на мережні обчислювальні вузли. Для установки дистрибутивів монітора-розподільника використовується майстер установки Install Shield Wizard, дозволяє встановити системне забезпечення як службу під керуванням операційної системи, дозволяє взаємодіяти з інформаційною мережею та системним обчислювальним середовищем. При інсталяції служби, служба встановлюється від доменного адміністратора, якщо в інформаційній мережі налаштована доменна політика або системного адміністратора обчислювальної мережі. Порт для передачі потоку даних між системним програмним забезпечення монітора-розподільника за замовчуванням призначається за номером 4757 і може відповідним чином бути змінений після установки обчислювальної системи в налаштуваннях служби.

Через складність інформаційно-обчислювальної мережі підприємств при включенні нового системного програмного забезпечення до складу обчислювальної системи підприємства проводиться процес безперервної інтеграції, який дозволяє, в даній ситуації, за мінімально можливий час інсталяції встановити всі необхідні

обчислювальні вузли для повноцінної роботи оновленної інформаційної системи підприємства.

Послідовність інсталяції обчислювальних модулів (рис. 3.10) в інформаційно-обчислювальну мережу підприємства дозволяє покроково вносити зміни до методів обробки потоків даних у структуру обчислювальної мережі підприємства, зменшуючи, при цьому, час бездіяльності обчислювальної системи.

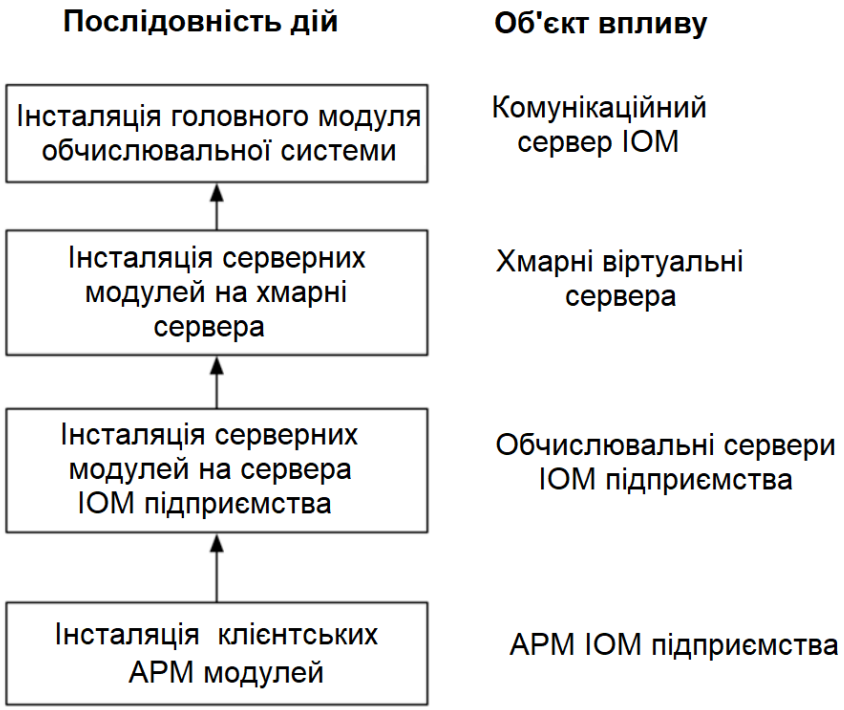



Рисунок 3.10 - Покрокова інтеграція монітора-розподільника у конвергентну обчислювальну мережу підприємства

Після завершення процесу інсталяції системи в інформаційно-обчислювальну мережу підприємства відбувається процес налаштування системи. Час, що витрачається на налаштування інформаційної системи, може бути розрахований наступним чином: $T_{заг} = Сума\,всіх\left(t_q\right)$, де $q = \{q_1, q_2, ..., q_n\}$ - множина всіх інформаційно-обчислювальних робіт, вирішуваних в інформаційно-обчислювальною мережею підприємства.

Виходячи з оцінки часу необхідного для налаштування обчислювальної системи підприємства, системний адміністратор мережі може запланувати час

інтеграції та час завершення етапу інтеграції з метою зменшення часу бездіяльності окремих обчислювальних компонентів інформаційно-обчислювальної системи.

Після завершення етапу налаштування обчислювальної системи розпочинається процес експлуатації інформаційної системи підприємства. На даному етапі експлуатації системи всі помилки та необхідні ризики всередині системного програмного забезпечення монітора-розподільника мають бути зведені до мінімуму з метою зменшення простою інформаційної системи або видачі обчислювальною системою некоректних даних.

У процесі експлуатації модуля монітора-розподільника системний адміністратор мережі має можливість переглядати отриману статистику з проведених операцій, монітор-розподільник інформаційно-обчислювальних робіт TiCloudBalancer написаний з використанням технології WPF C# дозволяє відображати: вартість проведених обчислень, виходячи із коефіцієнтів вартості інформаційно-обчислювальних ресурсів за одиницю часу, заданих адміністратором обчислювальної мережі; інформацію про завантаження обчислювальних серверів мережі підприємства; статистику щодо проведених розподілом інформаційно-обчислювальних робіт.

## 3.3 Висновки

На основі проведеного дослідження процесу інформаційних об'єктів у конвергентній обчислювальній мережі підприємства запропонованні алгоритми функціонування системного програмного забезпечення монітора-розподільника потоку даних.

На основі запропонованих алгоритмів та вибраних мов програмування розроблена структура монітора-розподільника інформаційно-обчислювальних робіт. Відповідно до структури системного програмного забезпечення розроблений системний програмний продукт монітор-розподільник інформаційно-

обчислювальних робіт, що підтверджує працездатність запропонованих алгоритмів розподілу даних.

Запропонований підхід до інтеграції монітора-розподільника потоку даних в інформаційно-обчислювальну мережу підприємства з конвергентною обчислювальною структурою мережі.

# ВИСНОВКИ

У даній кваліфікаційній роботі проаналізовані базові поняття «хмарних» технологій, основні властивості та архітектура, моделі розгортання «хмарних» технологій. Кваліфікаційна робота присвячена дослідженню інформаційних процесів у конвергентній обчислювальній мережі підприємства. Конвергентна обчислювальна мережа підприємства - сукупність «хмарних» віртуальних серверів та серверів інформаційно-обчислювальної мережі підприємства.

Показана роль конвергентної обчислювальної мережі у інформаційних-обчислювальних мережах підприємств, важливість завдання оптимального розподілу інформаційних об'єктів за критеріями вартості виконання та швидкості обробки інформаційно-обчислювальної роботи. Представлено особливості апаратної та програмної архітектури конвергентної обчислювальної мережі з включеною «хмарною» інфраструктурою.

Сформульовано задача розподілу інформаційних об’єктів між обчислювальними серверами підприємства та «хмарними» обчислювальними віртуальними серверами за критеріями вартості та часу використання обчислювального сервера для відповідної конкретної інформаційно-обчислювальної роботи.

Запропонована структура системи захисту даних в «хмарному» середовищі; розроблений алгоритму оптимального розподілу інформаційно-обчислювальних робіт підприємства конвергентної обчислювальної мережі; розроблено структуру модулів управління розподілом інформаційних об'єктів між «хмарними» серверами та серверами підприємства інформаційно-обчислювальної системи.

Результати кваліфікаційної роботи можуть бути застосовані до побудови конвергентної обчислювальної мережі підприємства, системи захисту баз даних, застосовної до хмарних технологій, підприємств будь-яких розмірів, від маленьких стартапів до великих концернів.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ


1. Алгоритмізація та програмування: Практикум / Л.І. Кублій. –Київ: КПІ ім. І. Сікорського, 2019. – 209 с.

2. Берко А. Ю. Системи баз даних та знань. Книга 1. Організація баз даних: навч. посібник [для студ. вищ. навч.закл.] / Берко А. Ю., Верес О. М., Пасічник В.В. – Львів : Магнолія, 2017. – 456 с.

3. Грод І.М. Аналіз ефективності деяких алгоритмів. Теорія і практика: навчальний посібник / І.М. Грод, С.В.Мартинюк, –Тернопіль:ТНПУ, 2017. – 64с.

4. Довгий С.О. Сучасні телекомунікації: мережі, технології, економіка, управління, регулювання / С.О. Довгий, О.Я. Савченко, П.П. Воробієнко – К.: Український Видатничій Центр, 2012. – 520 с.

5. Дурняк Б. В. Управління запитами в системах документообігу / Б. В. Дурняк, М. С. Пасєка, Т. М. Майба. – Львів : Укр. акад. друкарства, 2016.– 192 с.

6. Вишневецька В.П. Хмарні технології: [навч. посіб. / В. П. Вишневецька — К. : НПУ ім. М. П. Драгоманова, 2017. — 159 с

7. Катренко А. В. Системний аналіз / А. В. Катренко. – Львів: Новий світ, 2019. – 396 с.

8. Лавров Є. А. Математичні методи дослідження операцій : підручник / Є.А. Лавров, Л. П. Перхун, В. В. Шендрик. – Суми : Сумський державний університет, 2017. – 212 с.

9. Литвин В. В. Інтелектуальні системи / В. В. Литвин, В. В. Пасічник, Ю. В. Яцишин. – Львів: Новий Світ, 2015. – 406 с.

10. Петрик М.Р. Моделювання програмного забезпечення : науково-методичний посібник / М.Р. Петрик, О.Ю. Петрик – Тернопіль: ТНТУ ім. І. Пулюя, 2015. – 200 с.

11. Ситнік Б. Т. Основи інформаційних систем і технологій: Навч. посібник. – Харків: УкрДУЗТ, 2019. – 175 с.

12. Сучасні технології програмування: частина І. Практичні роботи / уклад.: В. І.Бендюг, Б. М. Комариста. – Київ: КПІ ім. І. Сікорського, 2019. – 269 с.

13. Табунщик Г. В. Проектування та моделювання програмного забезпечення сучасних інформаційних систем / Г. В. Табунщик, Т.І. Каплієнко, О.А. Петрова − Запоріжжя : Дике Поле, 2016. – 250 с.

14.Технології об'єктно-орієнтованого програмування: частина 1. Комп’ютерний практикум [Електронний ресурс]: навч. посіб. / В. І.Бендюг, Б. М. Комариста. – Київ: КПІ ім. І. Сікорського, 2018. – 225с.

Alankus G. Advanced C++: Master the technique of confidently writing robust C++ code / G. Alankus – Packt Publishing, 2019. – 762с.

15. Albahari Joseph. C# 8.0 in a Nutshell: The Definitive Reference/Joseph Albahari, Eric Johannsen – O'Reilly Media, 2020г. – 1104с.

16. Fuentes V.T. Enforcing database security on cloud using a trusted third party based model [Text] / V.T. Fuentes // 2438, Theses and Dissertations, ScholarWorks@UARK, 2017 y. - 50 p.

17. Griffiths Ian, Programming C# 8.0: Build Cloud, Web, and Desktop Applications/ Ian Griffiths. – O'Reilly, 2020. – 800с.

18. Hamza Y.A. Cloud computing security: Abuse and nefarious use of cloud computing [Text] / Y.A. Hamza - Int. J. Comput. Eng. Res, 2013 - 53 p.

# ДОДАТОК А

(обов’язковий)

Лістинг програмного коду ядра монітора-розподільника даних

```
namespace Balancer
{
public enum Protocol
{
TCP = 6,
UDP = 17,
Unknown = -1
};
public partial class BalancerForm: Form {
private Socket mainSocket; //The socket which captures all incoming packets
private byte[] byteData = new byte[4096];
private bool bContinueCapturing = false; //A flag to check if packets are to be captured or not
byte[] bytes=new byte[1024]; Socket senderSock;
IPAddress ipGlobal = null;
string sendDataGlobal = string.Empty; SocketPermission permission; IPHostEntry ipHost;
IPAddress ipAddr = null; IPEndPoint ipEndPoint;
string logText = string.Empty; Stopwatch stopWatch = New Stopwatch();
private delegate void AddTreeNode(TreeNode node); public BalancerForm()
{
InitializeComponent();
}
private void btnStart_Click(object sender, EventArgs e) {
if (cmbInterfaces.Text == "" && !isLocalTest)
{
"Balancer",
 }
try
{
MessageBox.Show("Select an Interface to capture the packets.",
MessageBoxButtons.OK, MessageBoxIcon.Error); return;
if (!bContinueCapturing) {
//Start capturing the packets... btnStart.Text = "&Stop"; bContinueCapturing = true;
raw socket, with the being IP
//Затискання плакатів для packets has to be a
//address family being of type internetwork, and protocol
mainSocket = новий Socket(AddressFamily.InterNetwork, SocketType.Raw, ProtocolType.IP);
```

```
/*//Bind the socket to the selected IP address mainSocket.Bind(new
IPEndPoint(IPAddress.Parse(cmbInterfaces.Text), 0));*/
IPHostEntry ipHost = Dns.GetHostEntry(""); IPAddress ip = null;
//Bind the socket to the selected IP address if (isLocalTest == true)
{
//IPHostEntry ipHost = Dns.GetHostEntry(""); IPAddress[] ipsAddr = ipHost.AddressList;
for (int i = 0; i < ipsAddr.Length; ++i)
{
if (IsIpAddress(ipsAddr[i].ToString()))
{
ip = ipsAddr[i]; ipGlobal = ip;
}
cmbInterfaces.Text = ip.ToString(); mainSocket.Bind(new
IPEndPoint(IPAddress.Parse(ip.ToString()), 0));
}
else
{
mainSocket.Bind(new IPEndPoint(IPAddress.Parse(cmbInterfaces.Text), 0));
}
//Set the socket options mainSocket.SetSocketOption(SocketOptionLevel.IP,
//Applies only to IP packets
//Set the include the header
//option to true
 SocketOptionName.HeaderIncluded, true);
 outgoing packets
 byte[] byTrue = new byte[4] { 1, 0, 0, 0};
byte[] byOut = new byte[4] { 1, 0, 0, 0}; //Capture
Winsock 2
 //Socket.IOControl is analogous to the WSAIoctl method of
mainSocket.IOControl(IOControlCode.ReceiveAll,
//Equivalent to SIO_RCVALL constant
//of Winsock 2
byTrue, byOut);
SocketFlags.None,
}
else
//Start receiving packets asynchronously mainSocket.BeginReceive(byteData, 0, byteData.Length,
New AsyncCallback (OnReceive), null);
{
btnStart.Text = "&Start"; bContinueCapturing = false;
//To stop capturing the packets close the socket mainSocket.Close();
//label1.Text = TTotalLength.ToString();
}
```

```
catch (Exception ex)
{
MessageBox.Show(ex.Message, "Balancer", MessageBoxButtons.OK,
MessageBoxIcon.Error);
}
private void OnReceive(IAsyncResult ar)
{
 try {
 int nReceived = mainSocket.EndReceive(ar);
//Analyze the bytes received...
ParseData(byteData, nReceived); if (bContinueCapturing)
{
 byteData = new byte[4096];
receive the incoming
SocketFlags.None,
}
//Another call to BeginReceive для того, щоб бути початком
//packets
mainSocket.BeginReceive(byteData, 0, byteData.Length, New
AsyncCallback(OnReceive), null);
 catch (ObjectDisposedException)
{
catch (Exception ex) {
MessageBox.Show(ex.Message, "Balancer", MessageBoxButtons.OK,
MessageBoxIcon.Error);
}
private void SnifferForm_FormClosing(object sender, FormClosingEventArgs e)
{
if (bContinueCapturing) {
mainSocket.Close();
}
int panel_visible = 1;
private void button1_Click(object sender, EventArgs e)
{
++panel_visible;
if (panel_visible %2 == 0) panel1.Visible = true;
else
panel1.Visible = false;
}
private void checkBox2_CheckedChanged(object sender, EventArgs e)
{
if (checkBox2.Checked == false) richTextBox2.Enabled = false;
 else
}
 richTextBox2.Enabled = true;
```

```
 private void checkBox1_CheckedChanged(object sender, EventArgs e) {
if (checkBox1.Checked == false) richTextBox1.Enabled = false;
else
}
richTextBox1.Enabled = true;
private void button2_Click(object sender, EventArgs e){
List<string> LocServIPs = new List<string>(); List<string> OblServIPs = new List<string>();
string rt1=richTextBox1.Text; string[] rts1 = rt1.Split('\n');
string rt2=richTextBox2.Text; string[] rts2 = rt2.Split('\n');

SaveFileDialog sDialog = New SaveFileDialog();
sDialog.Filter = "ips files (*.ips)|*.ips|All files (*.*)|*.*"; sDialog.FileName = "ips1";
Regex check = new Regex(@"^\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}(\:\d{1,5})?$");//xxx.xxx.xxx.xxx:xxx x
int Kcheck = 0;
for (int i = 0; i < rts1.Length; ++i) {
if (!check.IsMatch(rts1[i]) && rts1[i] != "") //
++Kcheck;
}
for (int i = 0; i < rts2.Length; ++i) {
if (!check.IsMatch(rt s2[i]) && rts2[i] != "") //верне був -
правильно чи ні
++Kcheck;
}
if (Kcheck > 0) {
 else
 MessageBox.Show("Помилка у введених IP-адресах");
if (sDialog.ShowDialog() == DialogResult.OK) {
 try {
List<string> obj = new List<string>(); obj.Add(rt1);
obj.Add(rt2);
SerializeData.Serializer.SaveListToBinnary(sDialog.FileName, obj);
}
catch {
MessageBox.Show("Помилка збереження");
}
int rb_change = 0;
private void richTextBox1_TextChanged(object sender, EventArgs e){
if (rb_change == 0) {
richTextBox1.Text = null; richTextBox2.Text = null;
}
++rb_change;
}
```

```
private void richTextBox2_TextChanged(object sender, EventArgs e)
{
if (rb_change == 0)
{
richTextBox2.Text = null; richTextBox1.Text = null;
}
++rb_change;
}
private void button3_Click(object sender, EventArgs e)
{
OpenFileDialog oDialog = New OpenFileDialog();
oDialog.Filter = "ips files (*.ips)|*.ips|All files (*.*)|*.*";
if (oDialog.ShowDialog() == DialogResult.OK)
{
try {
List<string> objlist = new List<string>();
SerializeData.Serializer.LoadListFromBinnary(oDialog.FileName, out objlist);
richTextBox1.Text = objlist[0];
richTextBox2.Text = objlist[1]; richTextBox1.Text = objlist[0];
}
catch {
MessageBox.Show("Не вдається завантажити"); }
```

## Конфігураційні файли монітора-розподільника даних

Вміст файлу конфігурації main_config.xml

```
<config>
<priority>minimum_ cashe</priority>
<server_type>Microsoft Windows 2008 Server</server_type>
<local_servers>
<server id=”1” state=”working” CPU=”Intel Core i5 2,3 ГГц” Memory=”16Гб” network=”23 мбіт/с” />
<server id=”2” state=”working” CPU=”Intel Core i5 2,3 ГГц” Memory=”16Гб” network=”18 мбіт/с” />
<server id=”3” state=”working” CPU=”Intel Core i5 2,3 ГГц” Memory=”16Гб” network=”53 мбіт/с” />
</local_servers>
<cloud_servers>
<server id=”4” state=”working” CPU=”Intel Core i5 4 ГГц” Memory=”64Гб” network=”157 мбіт/с” />
<server id=”5” state=”working” CPU=”Intel Core i7 4 ГГц” Memory=”64Гб” network=”115 мбіт/с” />
</cloud_servers>
</config>
<config>
```

Вміст конфігураційного файлу app_config.xml

<allow_application>

<app id="32" port="1032"/>

<app id=”33” port=”2343”/>

</allow_application>

</config>

<config>

Вміст файлу конфігурації net_config.xml

<netmask>255.255.128.000</netmask>

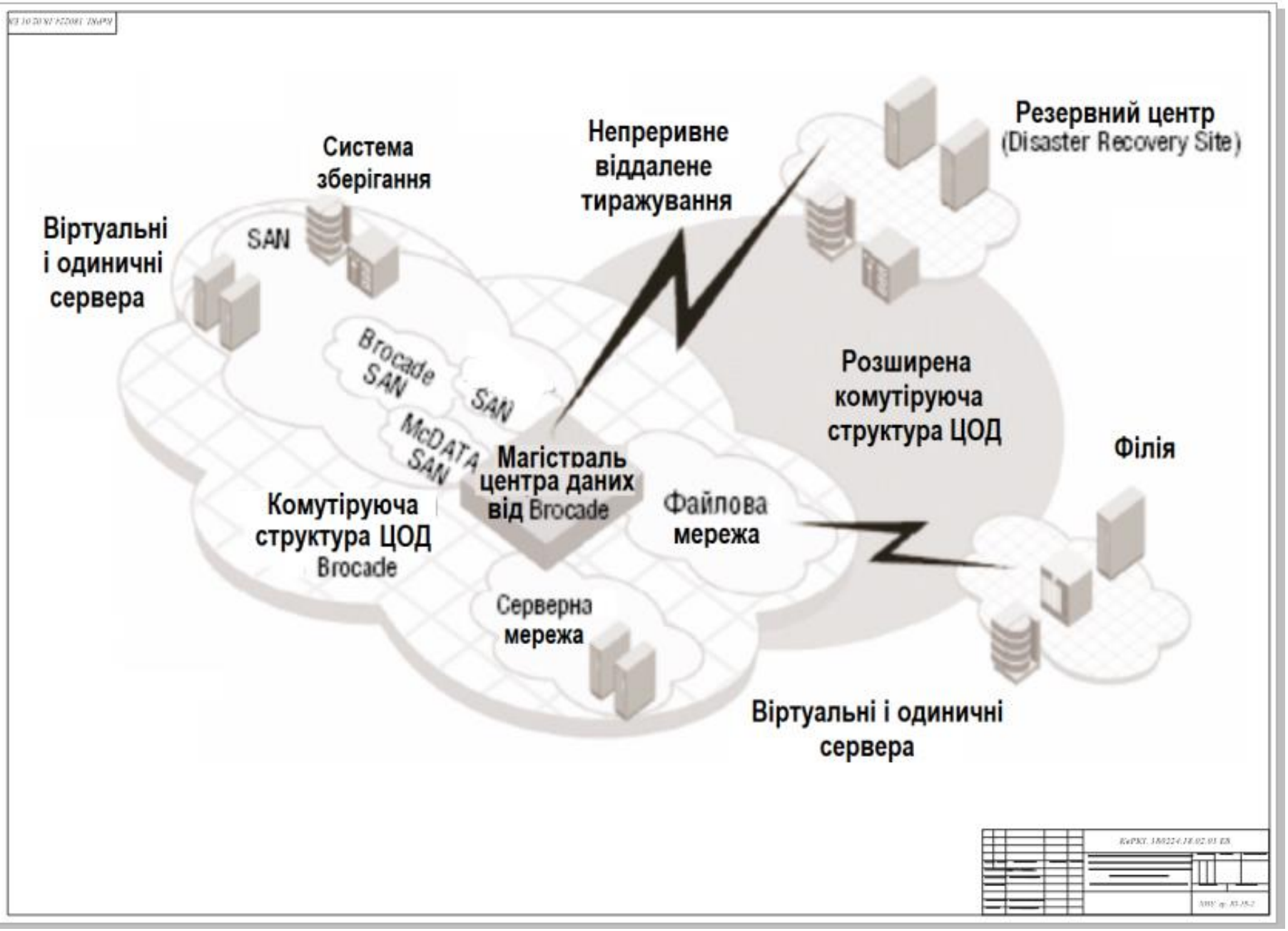
<port>5747</port>

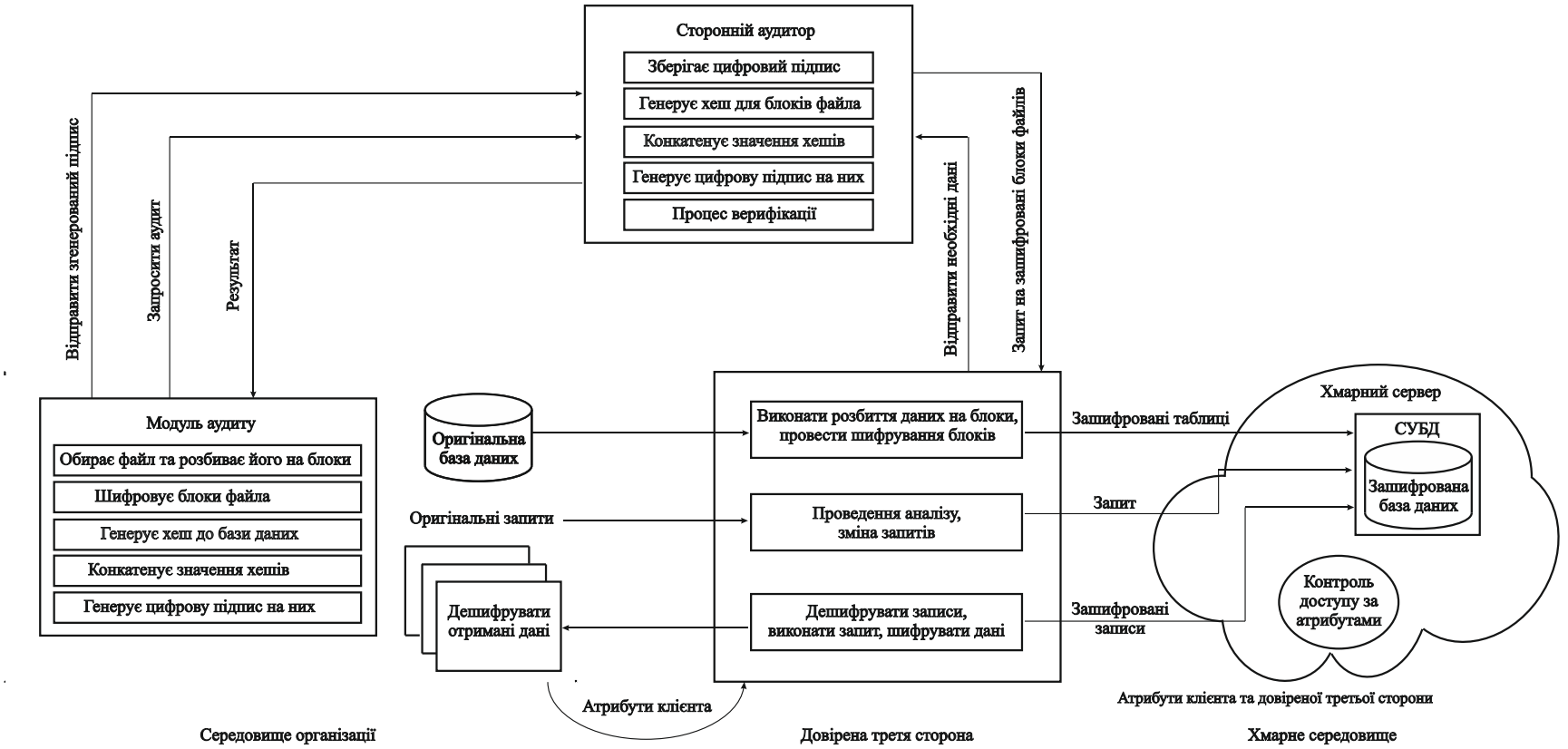
<gateway_ip>192.168.23.23</gateway_ip>

</config>

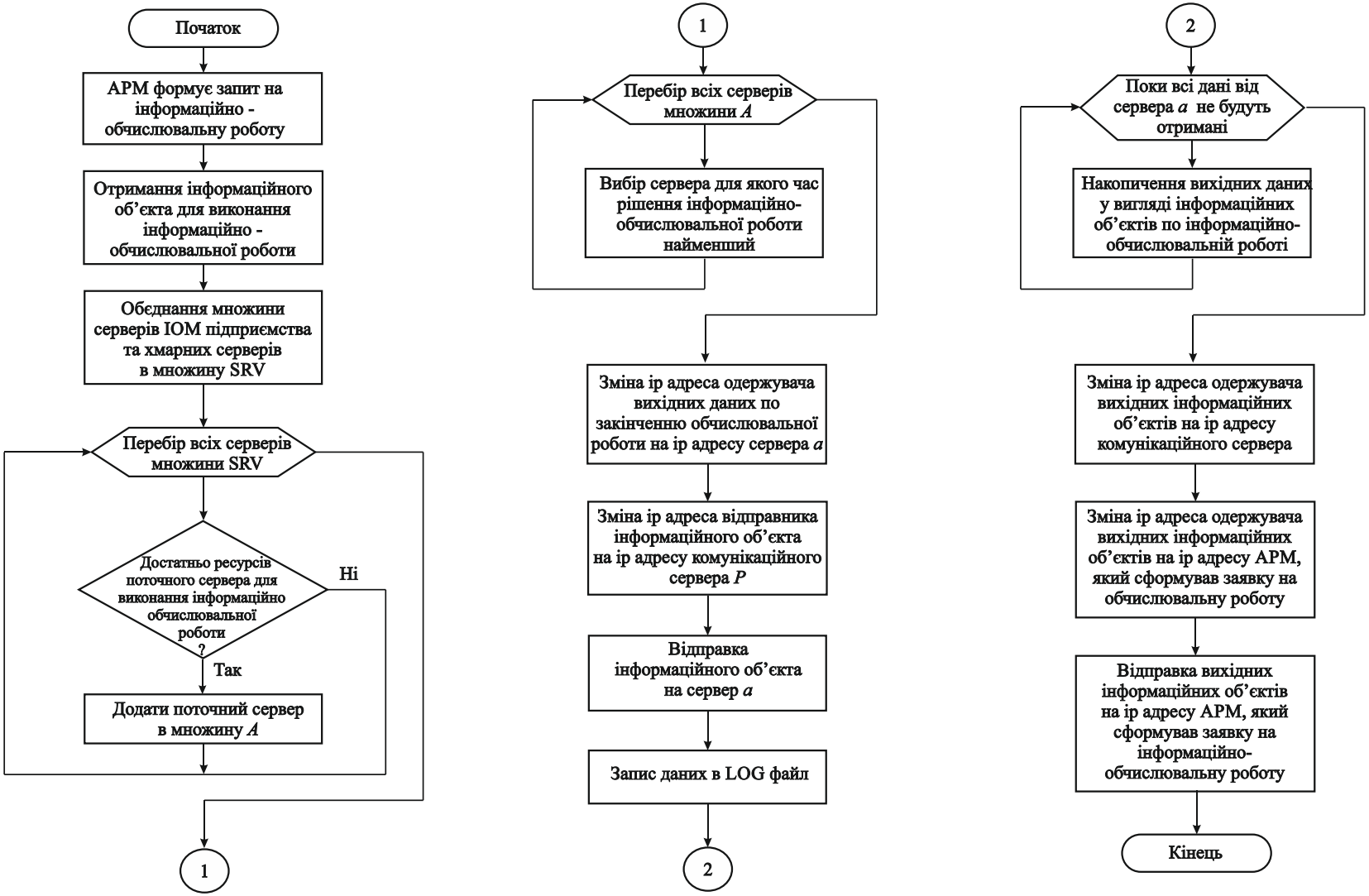
**ДОДАТОК Б**

Копія графічної частини

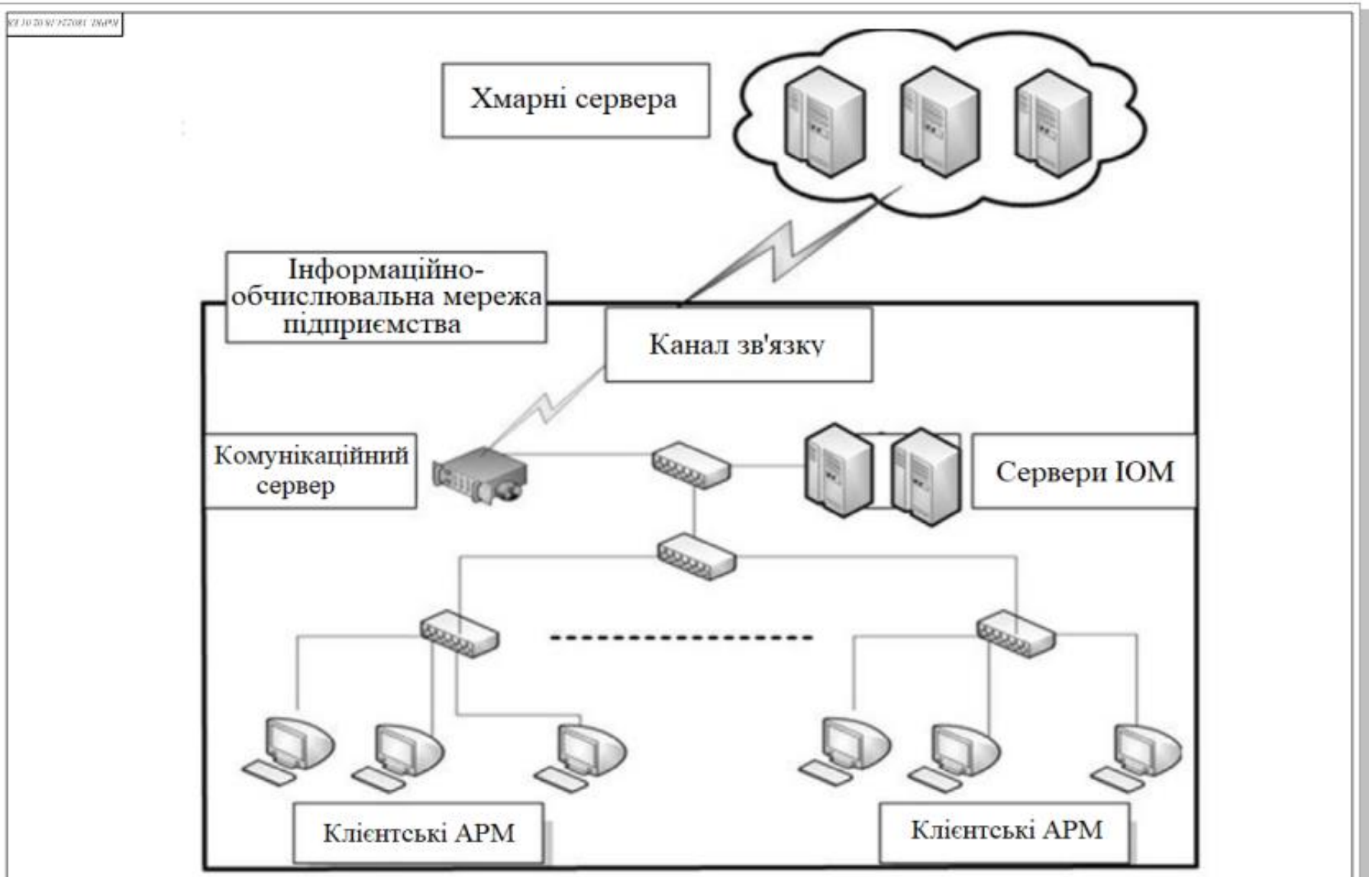

Система
зберігання
Непреривне
віддалене
тиражування
Резервний центр
(Disaster Recovery Site)
Віртуальні
і одиничні
сервера
SAN
Brocade
SAN
SAN
McDATA
SAN
Розширена
комутіруюча
структура ЦОД
Філія
Магістраль
центра даних
від Brocade
Комутіруюча
структура ЦОД
Brocade
Файлова
мережа
Серверна
мережа
Віртуальні і одиничні
сервера

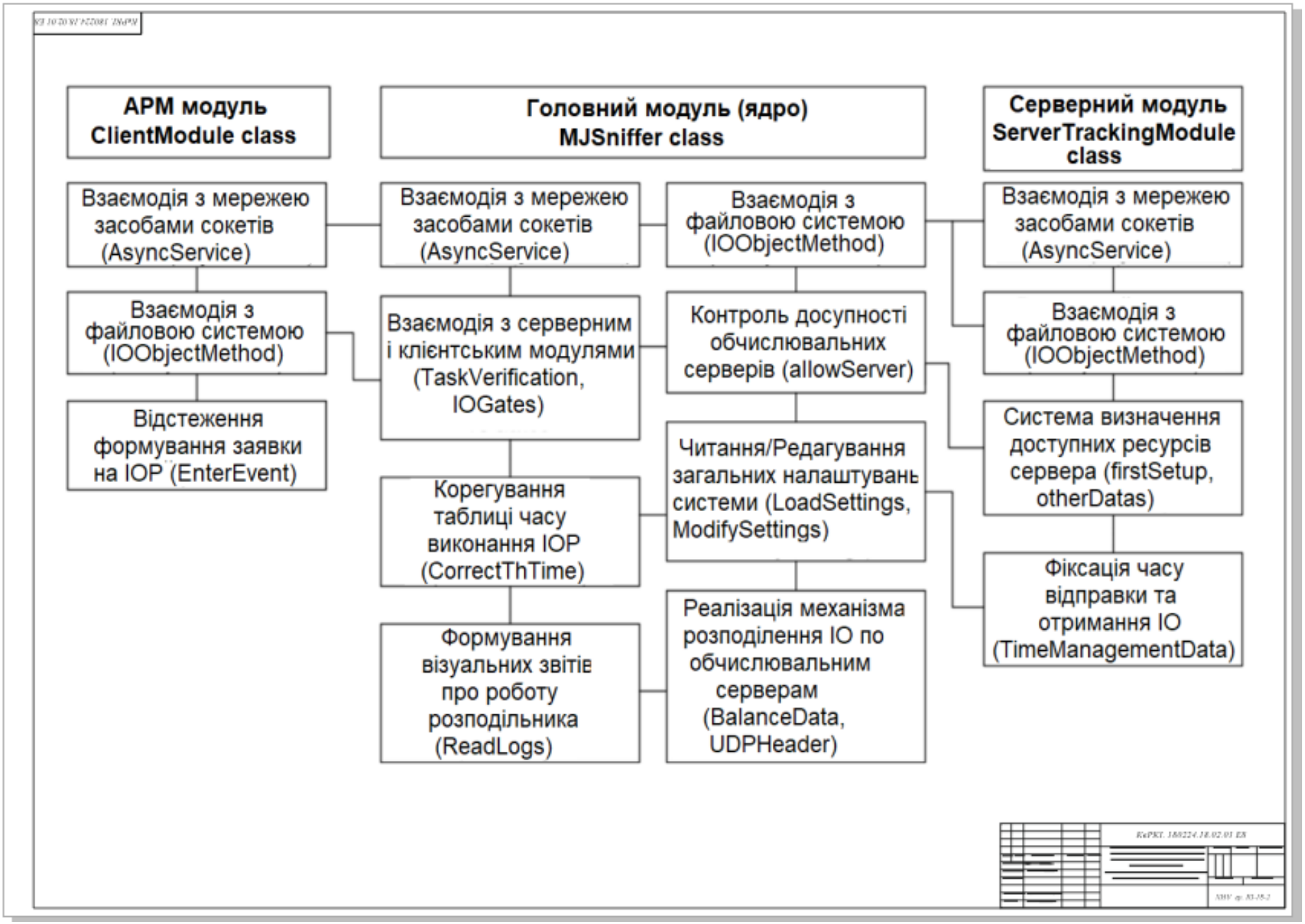



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | КвРКІ. 180224.18.02.01 ЕВ | |
| Зм. | Арк. | № докум. | Підпис | Дата | | | |
| Розробив | | Бойко В.О. | | | Система забезпечення функціонування конкурентної обчислювальної мережі підприємства на основі використання «хмарних» технологій. Структура системи захисту даних в «хмарному» середовищі | Літера | Маса | Масштаб |
| Перевірив | | Орленко В.С. | | | | | | |
| Т. контр. | | | | | | Аркуш | Аркушів | |
| Н. контр. | | Мостовий С. В. | | | | ХНУ, гр. КІ-18-2 | | |
| Затв. | | Ковиз Ю.П. | | | | | | |




| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | КвРКІ. 180224.18.02.01 ЕВ | |
| Зм. | Арк. | № докум. | Підпис | Дата | | | |
| Розробив | | Бойко В.О. | | | Система забезпечення функціонування конкурентної обчислювальної мережі підприємства на основі використання «хмарних» технологій. Алгоритм розподілу ІОР у конкурентній обчислювальній мережі підприємства | Літера | Маса | Масштаб |
| Перевірив | | Орлова В.С. | | | | | | |
| Т. контр. | | | | | | Аркуш | Аркушів | |
| Н. контр. | | Мостовий С. В. | | | | ХНУ ср. КІ-18-2 | | |
| Затв. | | Ковиз Ю.П. | | | | | | |


Хмарні сервера
Інформаційно-обчислювальна мережа підприємства
Канал зв'язку
Комунікаційний сервер
Сервери ІОМ
Клієнтські АРМ
Клієнтські АРМ


АРМ модуль
ClientModule class
Головний модуль (ядро)
MJSniffer class
Серверний модуль
ServerTrackingModule
class
Взаємодія з мережею засобами сокетів (AsyncService)
Взаємодія з файловою системою (IOObjectMethod)
Відстеження формування заявки на IOP (EnterEvent)
Взаємодія з мережею засобами сокетів (AsyncService)
Взаємодія з серверним і клієнтським модулями (TaskVerification, IOGates)
Корегування таблиці часу виконання IOP (CorrectThTime)
Формування візуальних звітів про роботу розподільника (ReadLogs)
Взаємодія з файловою системою (IOObjectMethod)
Контроль досупності обчислювальних серверів (allowServer)
Читання/Редагування загальних налаштувань системи (LoadSettings, ModifySettings)
Реалізація механізма розподілення IO по обчислювальним серверам (BalanceData, UDPHeader)
Взаємодія з мережею засобами сокетів (AsyncService)
Взаємодія з файловою системою (IOObjectMethod)
Система визначення доступних ресурсів сервера (firstSetup, otherDatas)
Фіксація часу відправки та отримання IO (TimeManagementData)

# Anti-Plagiarism v-15.257

**Максимальное совпадение с одним документом 1.0%**

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 10%**

| | Документ | | Суммарное совпадение по Базе Данных | |
|---|---|---|---|---|
| ID: 104369<br>Название: Система забезпечення функціонування конвергентної обчислювальної мережі підприємства на основі використання « хмарних » технологій<br>Добавлено в БД: 2022-06-02<br>Авторы: Бойко Владислав Олександрович<br>Руководители: Орленко Вікторія Сергіївна<br>Консультанты:<br>Опоненты: | Символы | Лексемы | Символы | Лексемы |
| | 98089 | 1408 | 1397 (1%) | 20 (1%) |

Источник плагиата

| ID | Описание | Наличие плагиата в документе | |
|---|---|---|---|
| | | Символы | Лексемы |

UNICHECK

Ім'я користувача:
**Кафедра кібербезпеки**

ID перевірки:
**1011428847**

Дата перевірки:
**02.06.2022 11:53:51 EEST**

Тип перевірки:
**Doc vs Internet + Library**

Дата звіту:
**02.06.2022 12:05:37 EEST**

ID користувача:
**100008300**

---

Назва документа: **Бойко_Кваліфікаційна**

Кількість сторінок: **71** Кількість слів: **13726** Кількість символів: **118261** Розмір файлу: **3.54 MB** ID файлу: **1011309136**

---

# 7.28%
# Схожість

Найбільша схожість: **5.98%** з джерелом з Бібліотеки (ID файлу: **1011309109**)

| | | |
|---|---|---|
| 3.06% Джерела з Інтернету | 181 | Сторінка 73 |
| 6.41% Джерела з Бібліотеки | 138 | Сторінка 74 |

# 0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

# 0%
# Вилучень

Немає вилучених джерел

# Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

| | |
|---|---|
| Замінені символи | 2 |

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

**РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ**
освітнього ступеня «бакалавр»

Студент Бойко Владислав Олександрович
Тема Система забезпечення функціонування конвергентної обчислювальної мережі підприємства на основі використання «хмарних» технологій
Спеціальність: 123 - Комп'ютерна інженерія

**Обсяг кваліфікаційної роботи освітньо-кваліфікаційного рівня «бакалавр»:**
кількість листів креслень 5 ; кількість сторінок записки 65
1. Короткий зміст КР та прийнятих рішень В рамках кваліфікаційної роботи запропонована структура системи захисту даних в «хмарному» середовищі; розроблений алгоритм оптимального розподілу інформаційно-обчислювальних робіт підприємства конвергентної обчислювальної мережі; розроблено структуру модулів управління розподілом інформаційних об'єктів між «хмарними» серверами та серверами підприємства інформаційно-обчислювальної системи.

2. Висновок про відповідність КР дипломному завданню Кваліфікаційна робота у повній мірі відповідає поставленому завданню як в теоретичній, так і в практичній частині представленої роботи

3. Характеристика виконання кожного розділу проекту, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому, теоретичному, розділі кваліфікаційної роботи якісно та в повній мірі розглянуті методи вирішення поставленої задачі, проведено дослідження розв'язуваних завдань та стан інформаційно-обчислювального середовища підприємств, зроблено висновок про структуру інформаційно-обчислювальної мережі підприємства та переходу на конвергентні обчислювальні мережі, з включенням «хмарних» структур. Показано особливості у конвергентній обчислювальній мережі підприємства застосування «хмарних» технологій. Сформульована задача розподілу інформаційних об'єктів між «хмарними» обчислювальними серверами та серверами інформаційно обчислювальної мережі підприємства за критеріями вартості витрат та часу на проведення інформаційних обчислювальних робіт. В загальному усі розділи відповідають завданню та містять сучасні методи вирішення поставлених завдань.

4. Позитивні сторони проекту Кваліфікаційна робота відповідає сучасним вимогам до проектування інформаційних систем та містить ряд інноваційних рішень, зокрема, з точки зору розподілу інформаційних об'єктів між «хмарними» обчислювальними серверами та серверами інформаційно обчислювальної мережі підприємства за критеріями вартості витрат та часу на проведення інформаційних обчислювальних робіт..

5. Негативні сторони проекту В рамках кваліфікаційної роботи не розглядається психологічне питання відносно робітників при переході підприємства на конвергентні обчислювальні мережі, з включенням «хмарних» структур в рамках розробленої системи. Відсутні рекомендації щодо узагальнення запропонованого підходу до можливості використання на інших підприємствах.

6. Оцінка графічного оформлення та пояснювальної записки проекту Графічне оформлення виконане відповідно до теми кваліфікаційної роботи. На першому листку креслення наведено конвергентна мережа підприємства. В наступних листах креслення розглянуті питання - структура системи захисту даних в «хмарному» середовищі, алгоритм розподілу інформаційно-обчислювальних робіт у конвергентній обчислювальній мережі підприємства. Структура конвергентної обчислювальної мережі підприємства, модулі та функції монітора-розподільника інформаційно-обчислювальних робіт. В загальному графічне оформлення виконане на достатньому рівні. Пояснювальна записка відповідає стандартам до її оформлення

7. Відгук про проект в цілому В загальному кваліфікаційна робота заслуговує позитивної оцінки. Весь матеріал кваліфікаційної роботи структурований, чіткий та послідовний. Усі розділи кваліфікаційної роботи послідовні та логічні, що дозволяє чітко розуміти матеріал викладений в рамках даної кваліфікаційної роботи. Графічний матеріал дозволяє наочно побачити доцільність та ефективність рішень, які були прийняті за основу при розробці системи забезпечення функціонування конвергентної обчислювальної мережі підприємства на основі використання «хмарних» технологій.

8. Інші зауваження _____

9. Оцінка кваліфікаційної роботи Розглянувши позитивні та негативні сторони представленого дипломного проекту, можна зробити висновок, що він заслуговує оцінку «добре».

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи) _____

Мінцан Віктор Володимирович

доцент ТНІТ

«08» 06 2022 р. _____ (підпис)

ХНУ. Зам. 54. тир. 6000, 2007.

Завідувачу кафедри кібербезпеки
к.т.н., доц. Кльоцу Ю.П.
Бойка Владислава Олександровича
ПІБ здобувача вищої освіти
студента ФІТ, 4 курсу, групи КІ-18-2

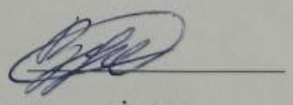
ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений (а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному *репозитарії університету*.

*Також надаю* університету право на передачу моєї роботи для обробки та *збереження* в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та *використання* роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

6.06.2022
дата


підпис

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КІБЕРБЕЗПЕКИ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Система забезпечення функціонування конвергентної обчислювальної мережі підприємства на основі використання «хмарних» технологій

Автор: В.О. Бойко

Спеціальність: 123 – Комп'ютерна інженерія

Освітня програма: освітньо-наукова

Науковий керівник: В.С. Орленко, к.т.н, доцент

Після аналізу звіту подібності зроблено такий висновок:

| № | Висновок | Позначка про відповідність |
|---|---|---|
| 1 | Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту. | відповідає |
| 2 | Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи | |
| 3 | Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат. | |
| 4 | Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту. | |

Підтвердження:

Запозичення виявлені в роботі, є законними і не є плагіатом, оскільки:

1) запозичення розміщені в розділах аналізу існуючих аналогів та технологій, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
3) окремі виявлені збіги є загальновживаними шаблонами, що використовуються при оформлені текстової документації, а саме шаблони рамок
4) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту, використання абревіатур.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості, складає 7,28%, з урахуванням наведених обгрунтувань, відповідає характеру дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи _____ В.С. Орленко

Гарант ОП _____ С.М. Лисенко

Завідувач кафедри Кб _____ Ю.П. Кльоц