

Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“___” _____ 2026 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за першим (бакалаврським) рівнем вищої освіти
на тему
“Програмне забезпечення системи моніторингу хмар та ЦОД за
допомогою технології System Center Operations Manager”

Виконав здобувач вищої освіти
IV курсу, групи КІ-22-2
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Бурдейний А.В.
«___» _____ 2026 р.

Керівник проекту
кандидат технічних наук, доцент
_____ Буравченко К.О.
«___» _____ 2026 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Освітній ступінь бакалавр
Галузь знань 12 "Інформаційні технології"
Спеціальність 123 "Комп'ютерна інженерія"
Освітньо-професійна (освітньо-наукова) програма "Комп'ютерна інженерія"

ЗАТВЕРДЖУЮ
Завідувач кафедри
д.т.н., проф.
_____ Олексій СМІРНОВ
« 27 » лютого 2026 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ПЕРШИМ (БАКАЛАВРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Бурдейному Артему Вадимовичу

(прізвище, ім'я, по батькові)

1. Тема роботи *Програмне забезпечення системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager*

2. Керівник роботи *Буравченко Костянтин Олегович, канд. техн. наук, доцент*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 136-02 від 27.02.2026 року

3. Строк подання студентом роботи до захисту 23.05.2026 р.

4. Мета та завдання випускної кваліфікаційної роботи: *Метою роботи є розробка програмного забезпечення системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager*

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

1. Призначення та область використання.

2. Перегляд аналогічних існуючих систем.

3. Опис і обґрунтування проектних рішень.

4. Етапи програмування системи.

5. Впровадження системи в промислову експлуатацію.

6. Висновки

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

Структурна схема системи 1 аркуш

Функціональна схема системи 1 аркуш

Діаграма процесів 1 аркуш

Блок-схема алгоритму роботи додатку 2 аркуша

7. Дата видачі завдання « 27 » лютого 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.03.2026 р.	
2.	Постановка задачі, оформлення ТЗ	15.03.2026 р.	
3.	Розробка моделі компонента	20.03.2026 р.	
4.	Розробка структур даних	25.03.2026 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.03.2026 р.	
6.	Програмування алгоритмів	10.04.2026 р.	
7.	Оформлення ПЗ	17.04.2026 р.	
8.	Попередній захист роботи	23.05.2026 р.	

Дата видачі завдання
« 27 » лютого 2026 р.

Підпис керівника

Буравченко К.О.
(прізвище та ініціали)

Завдання прийнято до виконання
« 27 » лютого 2026 р.

Підпис здобувача

Бурдейний А.В.
(прізвище та ініціали)

АНОТАЦІЯ

Бурдейний А.В. Програмне забезпечення системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2026.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

Метою розробки є програмне забезпечення системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

Результат роботи – програмна реалізація системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерна інженерія, System Center Operations Manager

ABSTRACT

Burdeinyi A.V. Software for cloud and data center monitoring system using System Center Operations Manager technology. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2026.

In this final qualification work for the first (bachelor's) level of higher education, software has been developed that is intended for a cloud and data center monitoring system using System Center Operations Manager technology.

The purpose of the development is software for a cloud and data center monitoring system using System Center Operations Manager technology.

The result of the work is a software implementation of a cloud and data center monitoring system using System Center Operations Manager technology.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software are provided.

The program can be used on a PC with Windows 10/11 OS.

The program was developed in the Python environment.

Keywords: computer engineering, System Center Operations Manager

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	2
ВСТУП.....	3
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	5
1.1 Призначення системи.....	5
1.2 Область застосування.....	8
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	10
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти.....	10
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	12
2.3 Розгорнута постановка завдання	18
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	19
3.1 Опис функціонування системи	19
3.2 Розробка структурної схеми.....	24
3.3 Розробка функціональної схеми	31
3.4 Розробка діаграми процесів.....	45
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	48
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	48
4.2 Захист розробленого програмного забезпечення.....	69
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	71
6 ОСНОВНІ ВИСНОВКИ.....	78
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	80

					ВКРБ-123.26.0034.00.00.ПЗ			
Вим.	Арк.	№ докум.	Підп.	Дата	<i>Програмне забезпечення системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager</i>	Лім.	Аркуш	Аркушів
Розроб.	Бурдейний А.В.					Б	1	87
Перев.	Буравченко К.О.							
Н.контр.	Коваленко А.С.					ЦНТУ КІ-22-2		
Затв.	Смірнов О.А.							

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

БД	—	база даних
ЛОМ	—	локальна обчислювальна мережа
ЦОД	—	центр обробки даних
ASP	—	Active Server Pages – активні серверні сторінки
DHCP	—	Dynamic Host Configuration Protocol – протокол динамічної конфігурації вузла
HTTP	—	HyperText Transfer Protocol – протокол передачі гіпер тексту
IMAP	—	Internet Message Access Protocol – протокол доступу до електронної пошти Інтернету
ICMP	—	Internet Control Message Protocol – міжмережевий протокол керуючих повідомлень
MMC	—	Microsoft Management Console
POP3	—	Post Office Protocol Version 3 – протокол поштового відділення, версія 3
SCOM	—	System Center Operations Manager
SQL	—	Structured Query Language – мова структурованих запитів
SMTP	—	Simple Mail Transfer Protocol – простий протокол передачі пошти
SNMP	—	Simple Network Management Protocol – простий протокол керування мережею
Syslog	—	стандарт відправки повідомлень про зміни які відбуваються в мережі
UDP	—	User Datagram Protocol – протокол користувальницьких дейтаграм

ВСТУП

Актуальність теми. System Center Operations Manager (SCOM) – це програмний продукт для керування й моніторингу ІТ-сервісів, застосунків, серверів у гетерогенному середовищі Windows, UNIX і Linux. Продукт здатний консолідувати інформацію про функціонування різних компонентів ІТ-інфраструктури, забезпечуючи її узагальнене подання в єдиній консолі.

Operations Manager являє собою компонент System Center, що поєднує в одній консолі відомості про моніторинг служби, пристроїв і операцій на множині комп'ютерів. Оператори можуть швидко оцінити стан ІТ-середовища й ІТ-служб, що виконують різні робітники навантаження в різних системах, одержуючи у вигляді оповіщень і різноманітних подань відомості про працездатність, захист, продуктивність і проблемах з конфігурацією, які впливають на працездатність служб.

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.
- Дослідження системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.
- Програмна реалізація системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

КБПЗ_2026

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Нижче приводиться короткий довідник по використанню загальних завдань

Перевірка наявності проблем у групі керування

– Перевірте подання **Працездатність групи керування** в папці Operations Manager робочої області **Спостереження**.

– Перегляньте зведення **Стан і попередження** на сторінці "Огляд моніторингу".

– Перевірте **Активні попередження** в робочій області **Спостереження**.

– Перевірте **Стан завдань** у робочій області **Спостереження**.

Запуск спостереження за комп'ютером

На сторінці "Огляд адміністрування" клацніть область **Налаштування комп'ютерів і пристроїв для керування**.

Запуск моніторингу мережевого пристрою

На сторінці "Огляд адміністрування" клацніть область **Налаштування комп'ютерів і пристроїв для керування**.

Створення або зміна пула ресурсів

У робочій області **Адміністрування** виберіть **Пули ресурсів**:

- Створення групи.
- Перегляд доступних груп.
- Перегляд членів групи.
- Перегляд стану групи.
- Перегляд схеми групи.
- Зміна групи.

Виберіть **Групи** в робочій області **Створення й налаштування**.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

Створення подання

У робочій області **Спостереження** або **Моя робоча область** у нижній частини панелі навігації натисніть кнопку **Створити подання**.

Налаштування параметрів подання для власного використання

У робочій області **Спостереження** клацніть правою кнопкою миші подання на панелі навігації, а потім виберіть пункт **Налаштування подання**.

Зміна інтервалу пульсів.

У робочій області **Адміністрування** виберіть **Параметри**, потім в області **Агент** натисніть **Пульс**.

Зміна припустимої кількості пропущених пульсів

У робочій області **Адміністрування** виберіть **Параметри**, потім в області **Сервер** натисніть **Пульс**.

Зміна параметра для правила, монітора або попередження.

Зміна правил, моніторів або попереджень за допомогою створення перевизначення. Виберіть правило, монітор або попередження, відкрийте параметри перевизначень, клацнувши правою кнопкою миші або нажавши елемент **Перевизначення** в панелі інструментів, або вибравши **Перевизначення** в панелі завдань.

Зміна частоти видалення записів з робочої бази даних.

У робочій області **Адміністрування** виберіть **Параметри**, клацніть правою кнопкою миші **Очищення бази даних** і виберіть пункт **Властивості**.

Надання користувачеві дозволів на перегляд відомостей Operations Manager або виконання завдань.

У робочій області **Адміністрування** виберіть **Ролі користувачів**, клацніть правою кнопкою миші певну роль і натисніть **Властивості**.

Відображення подання панелі моніторингу на сайті SharePoint.

Необхідно розгорнути веб-частину Operations Manager на сайті SharePoint, налаштувати веб-частину для зв'язку з веб-консоллю й додати веб-частину на сторінку SharePoint.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

Одержання повідомлень про попередження по електронній пошті, за допомогою миттєвого обміну повідомленнями або текстовими повідомленнями.

Клацніть правою кнопкою миші попередження, виберіть пункт **Передплата на повідомлення**, потім натисніть кнопку **Створити**.

Дослідження сірого агента.

У робочій області **Спостереження** в поданні **Operations Manager\Agent Details\Agent Health State** в області **Стан агентів зі спостерігача служби працездатності** клацніть сірий агент, потім на панелі **Завдання** виберіть **Показати дані підключення сірого агента**.

Тимчасова зупинка спостереження за комп'ютером.

У робочій області **Спостереження** виберіть **Комп'ютери Windows**, клацніть правою кнопкою комп'ютер, для якого потрібно зупинити спостереження, потім натисніть **Режим обслуговування**.

Спостереження за працездатністю групи керування

У робочій області **Спостереження** перейдіть до розділу **Operations Manager\Працездатність групи керування**, щоб переглянути важливі відомості про стан працездатності компонентів і агентів у групі керування.

Перенос баз даних Operations Manager

Щоб оновити SQL Server або перейти на сервер з новим обладнанням, виконаєте дії по переміщенню робочої бази даних або бази дані сховища.

Налаштування зберігання даних у базах даних Operations Manager

Для зберігання даних протягом певного періоду часу й підтримки оптимальної продуктивності можна налаштувати зберігання даних у базі даних Operations Manager і базі дані сховища даних.

Як і коли варто очищати кеш

Відомості про те, як і коли варто очищати кеш при усуненні неполадок консолі або агента.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

Налаштування підключень із Operations Manager до SQL Server

Якщо необхідно перемістити бази даних Operations Manager на інший сервер SQL Server, переналаштувати екземпляр SQL Server або реалізувати SQL AlwaysOn, виконаєте дії.

Інтеграція з Azure Monitor

Журнали інтеграції з Azure Monitor надають платформу аналітики, що доповнює сховище даних і забезпечує поліпшену аналітику й продуктивність запитів для більших обсягів інформації, а також дозволяє зіставляти журнали по ресурсах Azure і зберігати дані моніторингу протягом більш довгого часу.

1.2 Область застосування

Областю застосування є хмари й ЦОД.

IT Cloud (IT-хмара) – це сукупність декількох IT-ресурсів. До складу хмари можуть входити сервера, канали зв'язку, мережеве устаткування, дисковий простір і різне ПЗ. Головна мета створення хмарних систем – передача доступу до даних певному колу, що зберігаються в хмарі, користувачів.

По форматі доступу хмари ділять на приватні й публічні. До інформації, що зберігається в публічній хмарі, доступ можна одержати за допомогою мережі Інтернет. До приватної хмари доступ строго обмежений, такі ресурси використовуються переважно для внутрішніх цілей компанії-власника.

Існує також поняття гібридної хмари, що будується на базі публічних і приватних ресурсів. У гібридній хмарі передбачена можливість залучення вільних ресурсів публічної хмари, коли вичерпуються ліміти наявної частки.

Головні переваги хмарних технологій – можливість систематизації інформації, надійний захист даних і збереження швидкого доступу до них.

Центр обробки даних (ЦОД), або дата-центр – це спеціалізований будинок, у якому компанії розміщують своє серверне й мережеве устаткування. Саме дата-центри відповідають за фізичну схоронність і безперебійну роботу

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

устаткування. У цьому розділі розглянемо, як функціонують ЦОДу й завдяки чому їм вдається дотримувати твердих вимог з безпеки, які до них пред'являє ІТ-співтовариство.

Для обробки даних? – У цілому, звичайно, так. І ще всі знають, що дані обробляються комп'ютерами (між іншим, їх українська історична назва: електронно-обчислювальні машини, ЕОМ).

Очевидно, що це обчислювальне устаткування повинне десь перебувати. От для його розміщення й призначені центри обробки даних.

Далі згадаємо про те, що комп'ютерному устаткуванню потрібне електроживлення; що під час його роботи виробить тепло; що устаткування любить збирати пил; і, звичайно, що воно не повинне бути доступно будь-якому, що цікавиться.

Зазначені вимоги породжують основні функції центра обробка даних:

- захист розміщеного устаткування від впливу навколишнього середовища;
- забезпечення устаткування якісним і безперебійним електроживленням;
- відвід виділюваного тепла, вентиляція кондиціонованим повітрям;
- керування фізичним доступом до устаткування, його охорона.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти

Усе більше інформації з'являється в інтернеті про інтеграцію компаній які займаються програмним забезпеченням, для керування інфраструктурою дата центра. Зовсім недавно з'явилася інформація про злиття двох компаній Nlyte Software і FieldView Solutions. Компанія DCIM на ринку вже більше 10 років і на даний момент у її состав входять більше шести десятків постачальників програмного забезпечення для ЦОД. Найближчим часом планується скоротити число постачальників, але дане рішення не вплине на загальну ситуацію компанії. Усе більше й більше компаній залучає можливість одержати понад прибуток, до них ставляться, як великі компанії так і стартапи. Вендори не поспішають вийти на ринок DCIM, і набираючись досвіду на простих системах моніторингу й керування інфраструктурою дата центра.

Компанії Siemens Building Technology і ZNV Technology підписали договір про співробітництво в сфері ЦОД. Їх основне спільне співробітництво стосується спільному дослідженні й сприянні більше інтелектуальних, безпечних, надійних, ефективних, екологічно чистих технологій, продуктів і рішень дата центрів, а саме на створення високоефективних і професійних продуктів, які спрощують експлуатацію й технічне обслуговування інфраструктури будинків.

Останнє відновлення DCIM-рішення Sunbird dcTrack допомагає управляти дата центром швидше й точніше. Софтверна компанія Sunbird Software, що спеціалізується на розробці програмних рішень для керування інфраструктурою дата-центра, оголосила про доступність нової версії DCIM-рішення Sunbird dcTrack. Продукт із порядковим номером 5.1 пропонує операторам ЦОД широкі

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

можливості для моніторингу активів і пов'язаних з ними силовими інфраструктурами. За допомогою даного продукту, оператори ЦОД, зможуть планувати й аналізувати використані ресурси, для прийняття рішень, відносно, обсягу інфраструктури.

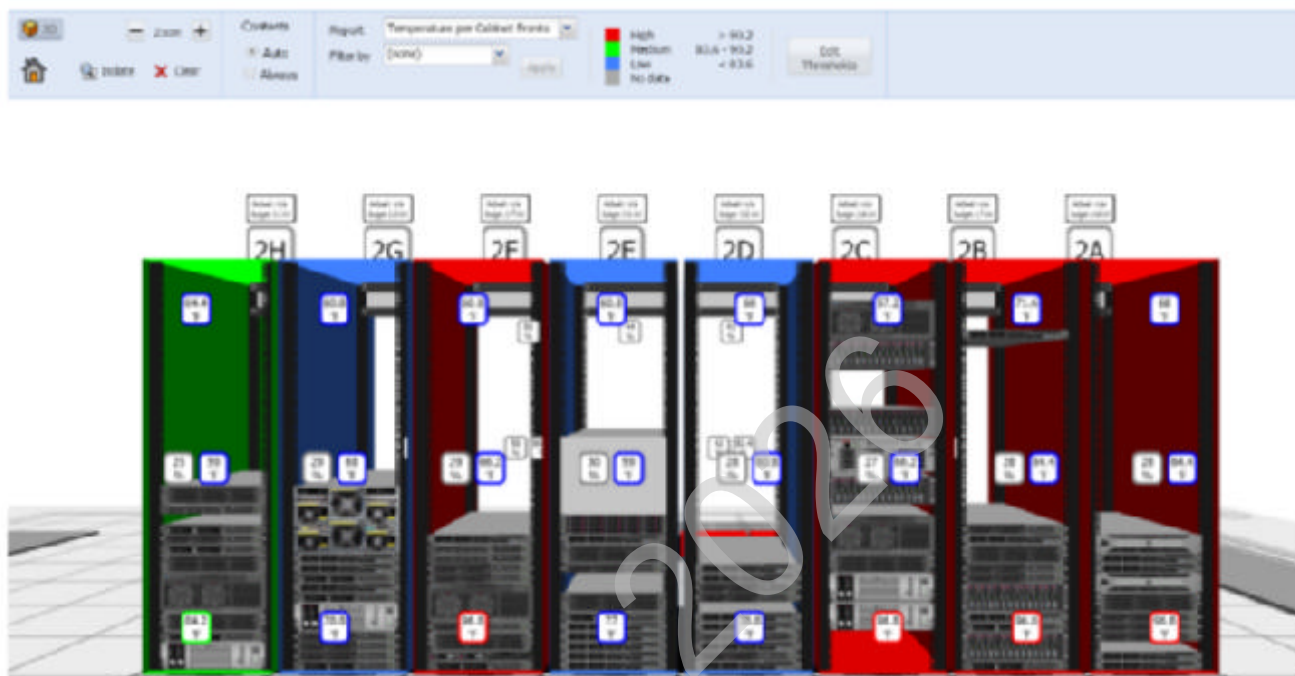


Рисунок 2.1 – Інтерфейс користувача Sunbird dcTrack

Розроблювачі DCIM, додали в новий продукт інструмент тривимірної візуалізації даних 3D Data Visualizer, що дозволить операторам ЦОД відображати результати моніторингу навколишнього середовища, одержувати звіти по енергоспоживанню й інших даних у режимі on-line, а також звіти по обсязі зайнятого простору й ваги розміщеного устаткування для більше точного планування й керування активами. Все це доступно без якого або, додаткового програмного забезпечення, для роботи досить стандартного браузера.

Девелопери подбали про розширення бібліотеки 3D моделей, переробивши код бібліотеки й зробивши її доступної через веб-інтерфейс, не забувши при тому впровадити більше швидкий і детальний механізм пошуку

моделей. У версії 5.1 поліпшений механізм імпорту й експорту даних про порти й структурні кабельні системи. Також з'явилася підтримка конфігурації активів з підключенням через IPv6.

Оновлене програмне рішення ONEView з порядковим номером 3.0 від компанії Hewlett Packard Enterprise (HPE), з'явиться вже в третьому кварталі цього року. Що дозволить власникам відразу декількох географічно розподілених ЦОД переглядати дані про всієї своєї IT-інфраструктуру на одному екрані. Поліпшене пропозиції пропонує операторам ЦОД, інтеграцію з HPE Helion CloudSystem 10 для спрощення контролю над фізичними серверами й віртуальними кластерами, а для спрощення керування з мережевими комутаторами – HPE Intelligent Management Center.

Компанія Panoramic Power збирається випустити новий продукт – Device Analyzer. Інструмент Device Analyzer може використовуватися для моніторингу й оптимізації самих різних елементів інфраструктури ЦОД: від систем кондиціонування повітря до освітлення. Виходячи із зібраних раніше даних, система також уміє виявляти аномалії й автоматично генерувати оповіщення, що направляються на зазначені заздалегідь вилучені термінали.

Як бачимо, світовий ринок DCIM, що почав формуватися більше десяти років тому й у цей час складається з 6 десятків постачальників програмного забезпечення, може незабаром серйозно змінитися але кардинальних змін не варто очікувати. Перспектива одержання надприбутку залучає нових гравців із числа як амбіційних стартапів, так і великих компаній.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – динамічна інтерпретована об'єктно-орієнтована скриптова мова програмування із строгою динамічною типізацією. Офіційний сайт мови програмування Python <https://www.python.org/>. Python – багатоцільова мова

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

програмування, яка дозволяє писати код, що добре читається. Відносний лаконізм мови Python дозволяє створити програму, яка буде набагато коротше свого аналога, написаного на іншій мові. Python – багатоплатформова мова програмування. Це означає, що програми на Python можна запускати в різних операційних системах без будь-яких змін.

Ще однією перевагою Python є його стандартна бібліотека, яка встановлюється разом з Python і містить готові інструменти для роботи з операційною системою, веб-сторінками, базами даних, різними форматами даних, для побудови графічного інтерфейсу програм тощо. Програми, написані на мові програмування Python, можуть бути як невеликими скриптами, так і складними системами. Python абсолютно безкоштовний.

Швидкість виконання коду Python

Один з можливих недоліків Python – швидкість виконання коду. Python не є компільованою мовою. Код на Python спочатку компілюється у внутрішній байт-код, який потім виконується інтерпретатором Python. У більшості випадків при використанні Python виходять програми повільніші в порівнянні з такими мовами, як C.

Втім, сучасні комп'ютери мають таку обчислювальну потужність, що для більшості застосунків швидкість розробки важливіша швидкості виконання, а програми на Python зазвичай пишуться набагато швидше.

Окрім того, Python легко розширюється модулями, написаними на C або C++. Такі модулі можуть використовуватися для виконання частин програми, що створюють інтенсивне навантаження на процесор.

Використання Python

Python використовується для різних цілей: для створення ігор і веб-застосунків, розробки внутрішніх інструментів для різноманітних проектів. Мова також широко застосовується в науковій області для досліджень і розв'язування прикладних завдань.

Застосування мови програмування Python:

1. BitTorrent – протокол для обміну даними.
2. Ubuntu Software Center – вільне програмне забезпечення для пошуку, установки і видалення пакунків в системі Ubuntu Linux.
3. Blender – програма для створення тривимірної комп'ютерної графіки, що включає засоби моделювання, анімації, вимальовування, пост-обробки відео, а також створення відеоігор.
4. GIMP – растровий графічний редактор, із підтримкою векторної графіки.
5. World of Tanks.
6. Вільна енциклопедія Вікіпедія.
7. Пошукова система Google.
8. DropBox – файловий хостинг, що включає персональне хмарне сховище, синхронізацію файлів і програму-клієнт.
9. YouTube – популярне відеосховище.

Версії Python

Мови програмування з часом змінюються – розробники додають в них нові можливості, а також виправляють помилки. Так з'являються різні версії мови. Наприклад, код написаний на Python 2 у більшості випадків не буде працювати у версії Python 3 без внесення додаткових змін.

Процесор є найважливішим компонентом в комп'ютері. Одна з основних функцій процесора – це обробка даних згідно комп'ютерної програми, яка є списком інструкцій, шляхом виконання арифметичних і логічних операцій над фрагментами даних. Кожна інструкція в програмі – це команда, яка «повідомляє» процесору, яку операцію він повинен виконати. Процесор комп'ютера може розуміти лише ті інструкції, які написані на машинній мові. Машинна мова – це штучна мова, створена для передачі команд комп'ютеру. За допомогою машинної мови створюються ефективні програми, оскільки розробник отримує доступ до всіх можливостей процесора. Машинна мова – мова низького рівня.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

Інструкція машинної мови існує для кожної операції, яку процесор здатний виконати – є інструкція для додавання чисел, є інструкція для віднімання чисел і т.д. Увесь набір інструкцій, який центральний процесор може виконати, відомий як набір інструкцій процесора.

Наприклад, у вас є певна програма, яка зберігається на диску вашого комп'ютера. Для виконання програми, ви здійснюєте подвійний клік на значку програми. Це змушує програму копіюватися з диска в оперативну пам'ять, після чого процесор комп'ютера виконує копію програми, яка знаходиться в оперативній пам'яті.

Коли процесор виконує інструкції програми, він бере участь у процесі, який є відомим як цикл `fetch – decode – execute` (отримати – декодувати – виконати). Цей цикл виконується для кожної інструкції у програмі і складається з трьох кроків:

Отримати

Програма – це послідовність інструкцій на машинній мові. Першим кроком циклу є завантаження (отримання) наступної інструкції з пам'яті в процесор.

Декодувати

Інструкція машинної мови – це двійкове число, яке представляє команду, що повідомляє процесору виконати певну операцію. На цьому кроці процесор декодує інструкцію, яку було «витягнуто» з пам'яті, для визначення того, яка операція повинна виконуватись.

Виконати

Останній крок циклу – виконати операцію.

Хоча процесор комп'ютера розуміє тільки машинну мову, людині непрактично писати програми на машинній мові. Така програма може мати тисячі або навіть мільйони бінарних інструкцій, і написання такої програми буде дуже обтяжливим процесом.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

З цієї причини була створена мова асемблера як альтернатива машинній мові. Замість використання двійкових чисел для написання інструкцій, мова асемблера використовує короткі слова, відомі як мнемокоди.

Незважаючи на те, що мова асемблера не вимагає двійкових інструкцій, як у випадку машинної мови, проте вона вимагає високих знань про процесор. Використовуючи мову асемблера, навіть для найпростішої програми, необхідно написати велику кількість інструкцій.

Мова програмування високого рівня дозволяє створювати складні програми, не знаючи, як працює процесор, і не записуючи великої кількості інструкцій низького рівня. Крім того, більшість мов програмування високого рівня використовують слова, які легко зрозуміти.

Python – одна із популярних сучасних мов програмування високого рівня. Python – інтерпретована мова програмування. Python – це високорівнева інтерпретована мова програмування, на відміну від C++, яка є прикладом компільованої мови програмування. Назва Python відноситься як до мови програмування, так і до інтерпретатора – комп'ютерної програми, яка зчитує початковий код (написаний на Python) і виконує інструкції (команди).

Для перекладу мови високого рівня на машинну мову доступні два типи програм:

1. Компілятор.
2. Інтерпретатор.

Завантаження Python

Версії інтерпретатора Python для різних операційних систем доступні для безкоштовного завантаження за адресою <https://www.python.org/downloads>.

Середовище програмування для Python

Для написання програм використовують текстові редактори або інтегровані середовища розробки, які включають в себе різні інструменти для роботи з кодом: засіб для написання коду (текстовий редактор), інтерактивний інтерпретатор, відлагоджувач тощо.

Текстові редактори та інтегровані середовища програмування для Python:

- IDLE – стандартний редактор Python. Встановлюється разом з Python для користувачів Windows, окремим пакунком для користувачів Linux.

- Notepad++ – безкоштовний текстовий редактор початкового коду, який підтримує велику кількість мов, в тому числі і Python. Лише для користувачів Windows.

- Visual Studio Code – це легкий, але потужний редактор початкового коду, який розповсюджується безкоштовно і доступний у версіях для платформ Linux, Windows і macOS.

- PyScripter – інтегроване середовище розробки для мови програмування Python. Для користувачів Windows. Поширюється безкоштовно.

- Wing IDE 101 – вільне інтегроване середовище для Python, розроблене для навчання програмістів-початківців. Для користувачів Linux, Windows і macOS. Поширюється безкоштовно.

- Geany – вільний текстовий редактор з базовими елементами інтегрованого середовища розробки, доступний для операційних систем Linux, Windows і macOS.

- PyCharm – інтегроване середовище розробки для мови програмування Python. PyCharm є власницьким програмним забезпеченням. Наявна безкоштовна версія Community з усіченим набором можливостей. Для користувачів Linux, Windows і macOS.

- Thonny – IDE для вивчення програмування мовою Python. Для користувачів Linux, Windows і macOS.

- Mu – редактор коду Python для програмістів-початківців. Для користувачів Linux, Windows і macOS.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускну кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

В процесі розробки випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Перед впровадженням Operations Manager важливо зрозуміти основні можливості й функціонал, які надає продукт. Це дозволить використовувати Operations Manager для рішення багатьох загальних проблем, які виникають у хмарі або центрах обробки даних, наприклад, питання забезпечення рівня обслуговування, надання доступу й забезпечення оптимальних умов для критичних застосунків.

Operations Manager – це комплексне рішення моніторингу, що вимагає ретельного планування, перш ніж його розгорнуть у корпоративному середовищі. Необхідно зрозуміти апаратні й програмні вимоги для рішення й наслідку для системи безпеки, пов'язані з моніторингом комп'ютерів у границях локальної мережі, мережі периметра або публічній хмарі. При налаштуванні сховища для бази даних Operations Manager потрібно добре представляти обсяг комп'ютерів і пристроїв, які контролюються, і розрахувати необхідний розмір баз даних.

Розгортання нових груп керування System Center Operations Manager

При плануванні нового розгортання Operations Manager потрібно взяти до уваги вимоги до апаратного забезпечення й інші супутні фактори. Варто також урахувати фактори безпеки облікових записів, які потрібні Operations Manager і состав групи керування. Наприклад може знадобитися встановити дві окремі групи керування для масштабування Operations Manager і задоволення вимог до моніторингу.

Після установки Operations Manager також необхідно враховувати деякі з найбільш загальних параметрів, так повинні бути настроєні параметри зберігання даних (очищення бази даних) і параметри ручної установки агента. Також варто вибрати підходящий метод розгортання агента, як, наприклад, використання

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

метод установки через консоль (або примусова установка) або метод ручної установки агента Operations Manager.

Апгрейд Operations Manager

Якщо у Вас уже встановлений Operations Manager, то корисно знати, як оновити Operations Manager до System Center Operations Manager.

Модернізація основних компонентів для System Center Operations Manager може виконуватися на Operations Manager або на більше пізній версії. Більше ранні версії Operations Manager, як Operations Manager SP1 необхідно оновити до Operations Manager, перш ніж вони можуть бути оновлені до System Center Operations Manager. Додатково до установки Operations Manager необхідно запустити принаймні накопичувальне відновлення 4 (Cumulative Update 4).

Перед установкою System Center Operations Manager потрібно вирішити кілька важливих завдань. У цьому модулі розглянутий порядок установки відновлень, які повинні застосовуватися при відновленні для System Center Operations Manager.

Перш ніж може бути виконане відновлення для System Center Operations Manager, повинна бути запущена група керування System Center SP1 Operations Manager.

Налаштування моніторингу фабрик і застосунків

При розгортанні Operations Manager і установки агентів на комп'ютерах, де потрібно контролювати комп'ютери й служби, варто встановити пакети керування для початку моніторингу. Перед установкою пакетів керування необхідно розуміти конструкцію пакета керування, включаючи всі елементи пакета. Необхідно розуміти, як налаштувати фабрику й моніторинг застосунків в Operations Manager для приватних і публічних хмарних середовищ, у тому числі інтеграцію Operations Manager і диспетчера Virtual Machine Manager, а також інтеграцію Operations Manager з Windows Azure.

Також потрібно зрозуміти, як налаштувати пакети керування для спостереження за застосунками, які запущені у фабриці, як Microsoft SQL Server,

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

сервер IIS і SharePoint Server. У підсумку показано, як налаштувати інтеграцію між Operations Manager і System Center Advisor, так щоб відображати оповіщення, що ставляться до компонентів фабрики й застосункам.

Моніторинг продуктивності застосунків

Більшість організацій використовують Microsoft .NET Framework для створення користувацьких застосунків, таких як для веб-сайтів інтрамережі. Пакети керування не будуть доступні для цих користувацьких застосунків, тут розглянуто, як проводити моніторинг продуктивності застосунків і налаштовувати його в Operations Manager. Моніторинг продуктивності застосунків забезпечує розширені можливості спостереження для .NET і Java застосунків, включаючи моніторинг серверних і клієнтських систем.

На додаток до моніторингу застосунків .NET прямо через Operations Manager моніторинг продуктивності застосунків може використовуватися з повною функціональністю Visual Studio IntelliTrace Collector для збору повного застосунку. В System Center також можна інтегрувати Operations Manager з Team Foundation Server. При інтеграції Operations Manager може використовуватися для синхронізації оповіщення з робочими елементами в Team Foundation Server.

Наскрізний моніторинг служб

При моніторингу ключових бізнес-застосунків спостереження повинне виконуватися з боку центра обробки даних, так і з боку кінцевого користувача. У цьому модулі описане створення штучних транзакцій для виміру продуктивності кінцевих користувачів.

Також розглянуто, як комбінувати моніторинг компонентів зі штучними транзакціями в моделі розподілених застосунків, які описують відносини між різними компонентами застосунку. Це дозволить забезпечити єдиний погляд на визначення причини й наступної потенційної неприступності служб.

Також у модулі показано, як побудувати інформативні панелі моніторингу Visio, для демонстрації в реальному часі стану середовища зовнішнім користувачам.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

Системи показників, інтерактивні панелі й звіти

Візуалізація стану служби й додатки в Operations Manager – це ключовий функціонал, що дозволяє створюватися інтерактивні панелі швидко й легко. Ці панелі можуть миттєво відображати продуктивність і доступність для одного або декількох застосунків в одній панелі.

У модулі показано як створити системи показників і панелі моніторингу й надати різним бізнес-користувачам панелі з моніторингом показників ефективності середовища. Наприклад, власник служби може зажадати подання, де показане, чи дотримує служба параметри угоди про рівень обслуговування чи ні.

Звітність також має важливе значення в процесі моніторингу середовища. Деякі співробітники можуть не мати доступу до Operations Console, але одержувати оповіщення по електронній пошті, і вони також можуть зажадати доступ до важливої інформації про стан і продуктивність основних застосунків і служб. У цьому випадку звіти можуть використовуватися для надання інтерфейсу до зібраних даних в Operations Manager.

Власникам служб і застосунків також необхідно знати, що важливі для бізнесу служби дотримують угод про рівні обслуговування по продуктивності й доступності. У такий спосіб у модулі розказано, як налаштувати відстеження рівня обслуговування й відображати ці показники в Operations Manager.

Створення й налаштування консолей

Команди підтримки застосунків, що використовують Operations Manager, можуть виконувати запити або через локально встановлену консоль, або через веб-консоль. Звичайно в рамках підтримки застосунків існує ряди ролей, які вимагають різні рівні дозволів при використанні Operations Console. Вони варіюються від дозволів тільки для читання до просунутих операторів. На додаток до цього всі ролі повинні обмежувати відображення даних тільки про ті комп'ютери й додатки, по яких оператори несуть відповідальність. У цьому модулі розказано, як використовувати убудовану безпеку на основі ролей в Operations Manager для гранульованих політик доступу до даних, поданням і

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

завданням в Operations Manager. Групи підтримки застосунку звичайно вимагають користуvalьницьких подань у консолі для ключових застосунків. У модулі розказано, як розробити й підготувати ці подання для відповідної команди підтримки. Хоча деякі члени команди можуть завжди тримати консоль відкритої, інші команди можуть зажадати відправлення повідомлення при виявленні проблеми. Operations Manager використовує повідомлення й канали повідомлень для підключення цієї функції. Цей модуль описує, як налаштувати й відправляти оповіщення по електронній пошті ключовим командам підтримки. Також щоб зменшити навантаження на різні групи підтримки застосунків, описано як налаштувати завдання діагностики й відновлення в Operations Manager за допомогою методів діагностики й автоматизації виправлення виявлених проблем.

Створення пакетів керування

Для типових застосунків, наприклад Microsoft SQL Server і Microsoft Exchange Server, є вже повністю закінчені пакети керування. При цьому можливо розширити моніторинг за замовчуванням, що втримуються в Operations Manager, створюючи правила, монітори й групи усередині консолі оператора.

Однак, може бути потрібно створити зовсім новий пакет керування для моніторингу користуvalьницьких застосунків. Для досягнення цього формується пакет керування в Visual Studio, компілюється й розвертається. Пакет керування повинен автоматично виявити й відслідковувати компоненти застосунку.

Проблема створення пакета керування, докладно розглянута в цьому модулі, вимагає базового розуміння ключових концепцій. Даний модуль охоплює всі необхідні теми, включаючи створення пакета керування в консолі й Visual Studio.

Інтеграція Operations Manager з іншими компонентами System Center

System Center складається з декількох компонентів, які забезпечують повне керування ІТ-середовищем, включаючи серверну й клієнтську інфраструктури, а також мережеву інфраструктуру, що їх зв'язує. Тут розглянута можливість керування клієнтськими пристроями за допомогою System Center.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

Для забезпечення зручно керування ІТ-середовищем можливо впровадити інші компоненти System Center з хмарою й ЦОДом. Це забезпечує легке й більше гнучке керування ЦОДом і хмарним середовищем, а також дозволяє автоматизувати безліч ручних процесів.

У цьому модулі показані кілька основних функцій інших компонентів System Center, а також переваги, які досягаються при інтеграції з ними Operations Manager.

Усунення несправності, налаштування й відновлення після збоїв

Рішення для моніторингу повинне бути постійно доступно для ключових бізнес-застосунків і постійно відповідати угоді про рівень обслуговування. Крім того оптимізована система моніторингу допоможе запобігти перебоям і погіршення продуктивності. У цьому модулі показано, як оптимізувати сховище даних і переконатися, що обрано правильний обсяг даних для зберігання.

У всіх ключових систем важливо знати, де шукати проблему при усуненні неполадок. У цьому допоможуть розібратися звіти продуктивності, наповнювані службами й агентами.

Для підвищення загальної доступності служби також розглянуте налаштування SQL AlwaysOn для Operations Manager, що дозволить збільшити доступність бази даних.

Відмова компонента в середовищі Operations Manager не менш важливий для розуміння можливостей відновлення несправних компонентів. У цьому заключному модулі розказано, як оптимізувати виконання аварійного відновлення в Operations Manager і усунути неполадки.

3.2 Розробка структурної схеми

Інфраструктура System Center Operations Manager

При установці Operations Manager створюється група керування. Група керування є основною одиницею функціональності. Як мінімум, група керування

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

складається із сервера керування, робочої бази даних і бази дані сховища даних звітів:

– **Сервер керування** – це центральний вузол адміністрування групи керування й взаємодії з базою даних. При відкритті консолі керування й підключенні до групи керування виконується підключення до сервера керування для цієї групи керування. Залежно від розміру обчислювального середовища група керування може містити як один, так і кілька серверів керування.

– **Робоча база даних** – це база даних SQL Server, що містить всі відомості про конфігурацію для групи керування й зберігає всі дані моніторингу, зібрані й оброблені для цієї групи керування. Робоча база даних зберігає оперативні дані, за замовчуванням протягом 7 днів.

– **База даних сховища даних** – це база даних SQL Server, що містить архівні дані моніторингу й попереджень. Дані, які записуються в робочу базу даних Operations Manager, також записуються й у базу дані сховища даних, тому звіти завжди містять актуальні дані. База дані сховища даних зберігає дані за тривалий період.

Якщо встановлюється функціональність створення звітів Operations Manager, то в групі керування також буде присутній сервер Reporting Server , що будує й представляє звіти на основі даних у базі дані сховища даних.

Сервери керування

Роль сервера керування складається в адмініструванні конфігурації групи керування, в адмініструванні агентів і взаємодії з ними, а також у взаємодії з базами даних у групі керування.

Для забезпечення додаткової потужності й постійної доступності група керування може містити кілька серверів керування. Якщо в групу керування додається не менш двох серверів керування, то ці сервери керування стають частиною пула ресурсів , а вся робота розподіляється по членах цього пула. При збої члена пула ресурсів його робоче навантаження будуть підхоплювати інші члени пула ресурсів. При додаванні нового сервера керування він автоматично

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

приймає на себе частина роботи від наявних членів пула ресурсів. Кожний член пула ресурсів буде управляти своїм набором вилучених об'єктів; у кожний момент часу жоден об'єкт не буде управлятися одночасно двома членами одного пула. Особливий тип сервера керування – це сервер шлюзу. Сервер шлюзу дозволяє спостерігати за комп'ютерами в доменах, що не є довіреними.

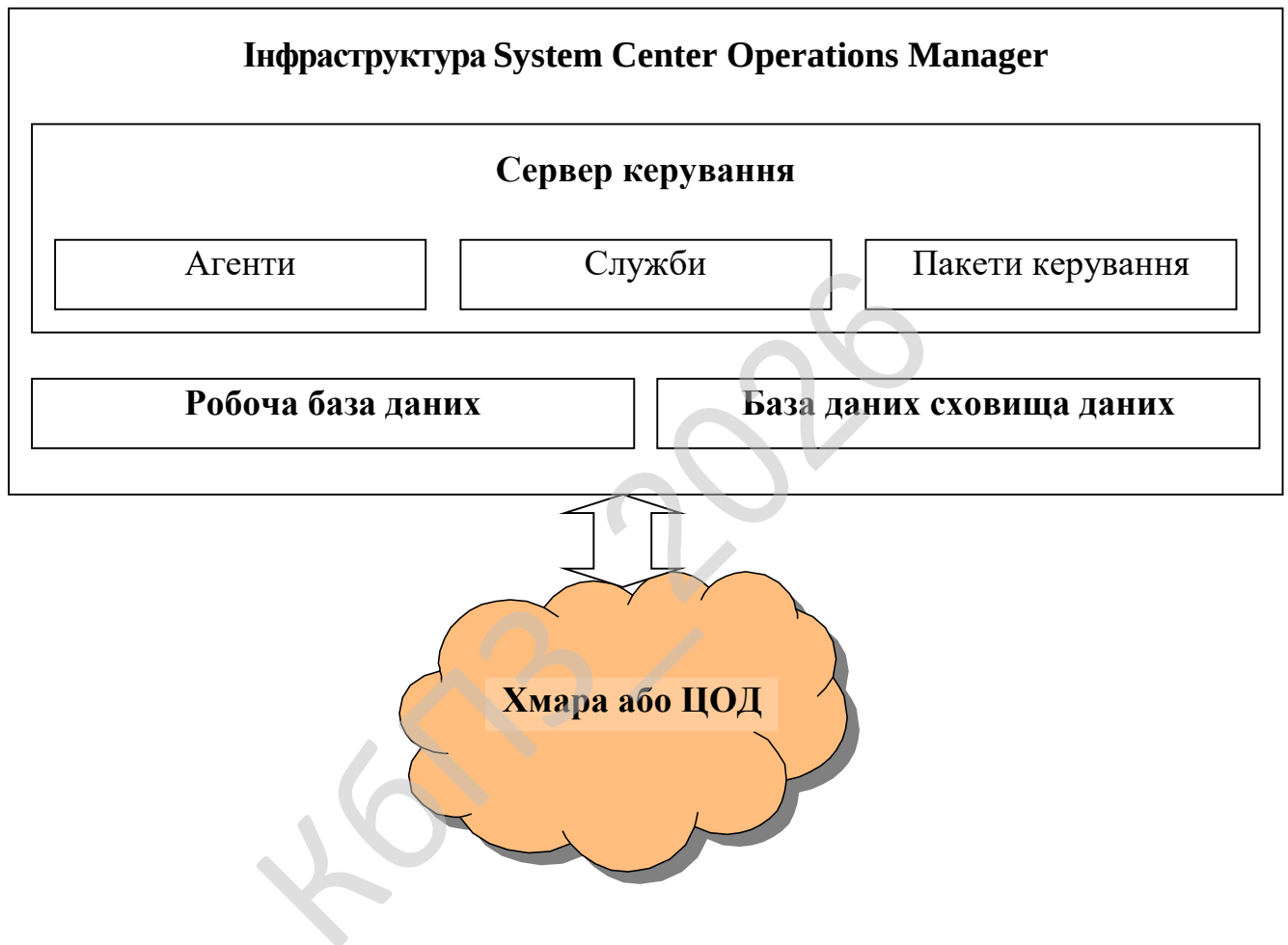


Рисунок 3.1 – Структурна схема системи

Агенти

Агент Operations Manager – це служба, що встановлюється на комп'ютер. Агент збирає дані, порівнює вибіркові дані із передвстановленими значеннями, створює попередження й реагує на запити. Сервер керування одержує конфігурації й поширює їхнім агентам на спостережуваних комп'ютерах.

Всі агенти відправляють звіти на сервер керування в групі. Цей сервер вважається основним сервером керування агента.

Агенти відслідковують джерела даних на спостережуваному комп'ютері й збирають відомості по конфігурації, що відправляється їм із сервера керування. Агент також визначає стан працездатності комп'ютера, що відслідковується, і його об'єктів і відправляє ці відомості на сервер керування. При зміні стану працездатності об'єкта або виконанні інших умов агент може створювати попередження. Це дозволяє операторам довідатися, коли щось вимагає їхньої уваги. Передаючи дані про працездатність об'єкта, що відслідковується, на сервер керування, агент надає поточну картину працездатності пристрою й всіх його застосунків.

Агент можна налаштувати як проксі-агента. Проксі-агент – це агент, що може направляти дані на сервер керування від імені комп'ютера або мережевого пристрою, відмінного від комп'ютера, на якому він розміщений. Наприклад, агент, установлений у фізичному вузлі кластера SQL, можна включити в якості проксі для спостереження за ресурсами кластера. За допомогою проксі-агентів можна спостерігати за комп'ютерами й пристроями, на яких неможливо встановити свого агента.

Служби

На спостережуваному комп'ютері агент Operations Manager відображається як служба Microsoft Monitoring Agent. Служба Microsoft Monitoring Agent збирає відомості про продуктивність, виконує завдання й інші дії. Навіть якщо ця служба не може підключитися до сервера керування, якому вона підкоряється, вона продовжує працювати й поміщає зібрані дані й події в чергу на диску спостережуваного комп'ютера. При відновленні підключення служба Microsoft Monitoring Agent відправляє зібрані дані й події на сервер керування.

Іноді службу Microsoft Monitoring Agent називають службою працездатності.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

Служба Microsoft Monitoring Agent також працює на серверах керування. На сервері керування ця служба виконує робочі процеси моніторингу й управляє обліковими даними. Для запуску робочих процесів служба викликає процеси MonitoringHost.exe з використанням зазначених облікових даних. Ці процеси виконують спостереження й збирають дані журналів подій, дані інструментарію керування Windows (WMI), а також виконують такі дії, як запуск скриптів.

На серверах керування також працює служба доступу до даних System Center і служба конфігурації керування System Center.

Служба доступу до даних System Center забезпечує консолі керування доступ до робочої бази даних і записує дані в цю базу даних.

Служба конфігурації керування System Center управляє відносинами й топологією групи керування. Вона також поширює пакети керування в спостережувані об'єкти.

Пакети керування

Робочі процеси, виконувані службою конфігурації керування System Center, задаються пакетами керування. Пакети керування задають, які відомості агент повинен збирати й відправляти на сервер керування для конкретного застосунку або технології. Наприклад, пакет керування BizTalk Server містить правила й монітори, які збирають і оцінюють події й операції, важливі для забезпечення працездатності й ефективності застосунку BizTalk Server.

Після того як Operations Manager установить агент на комп'ютері, він відправляє цьому агенту початкову конфігурацію. Початкова конфігурація включає виявлення об'єктів з пакетів керування. Пакет керування задає типи об'єктів, такі як застосунок й компоненти, які повинні спостерігатися на комп'ютерах, виявлених Operations Manager. Агенти відправляють дані на сервер керування, що визначає екземпляри об'єктів, виявлених на комп'ютері. Потім сервер керування відправляє агентам елементи пакетів керування, такі як правила й монітори, які застосовуються до виявлених об'єктів на кожному комп'ютері.

Правило задає події й дані про продуктивність, які варто збирати з комп'ютерів, і наступну обробку цих даних. Найпростіше представляти правила

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		28

як оператор If/Then (" якщо-тоді"). Наприклад, пакет керування для застосунку може містити наступні правила.

- Якщо в журналі подій з'явиться повідомлення про закриття застосунку, створити попередження.

- Якщо відправлення вихідного файлу закінчується невдачею, зібрати дані про подію, що відповідає цій помилці.

Як показано в цих прикладах, правила можуть створювати попередження й збирати події й дані про продуктивність, які агент відправляє на сервер керування. Крім того, правила можуть виконувати сценарії, наприклад для перезапуску застосунку, що відмовив.

Виявлені об'єкти мають стан працездатності, що відображається в консолі керування зеленим (успішне або працездатне), жовтим (попередження) або червоним (критичним або непрацездатне) кольором. Монітори визначають стану працездатності конкретних аспектів спостережуваного об'єкта. Наприклад, монітор для ємності диска може визначати зелений стан при заповнюванні диска менш, ніж на 85 відсотків, жовте – при заповнюванні більш, ніж на 85 відсотків і червоне – при заповнюванні більш, ніж на 90 відсотків. Монітор можна налаштувати таким чином, щоб він створював попередження при зміні стану.

Принципи виявлення об'єктів і спостереження за ними

На наступному рисунку показана спрощена ілюстрація, що демонструє виявлення об'єктів і спостереження за ними:

- Адміністратор налаштовує Operations Manager для пошуку керованих комп'ютерів. Додаткові відомості про виявлення комп'ютерів див. у розділі Агенти Operations Manager.

- Визначаються комп'ютери, які задовольняють заданим умовам, але поки не управляються.

- На виявленому комп'ютері встановлюється агент Operations Manager.

- Агент запитує дані конфігурації, після чого сервер керування відправляє дані конфігурації агента із установлених пакетів керування, включаючи класи для виявлення. Наприклад, якщо встановлені пакети керування операційної системи

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

Windows Server, те сервер керування буде відправляти агентів класи операційної системи.

– Агент порівнює дані конфігурації з комп'ютером, визначає виявлені об'єкти й відправляє ці відомості на сервер керування. Наприклад, агент буде відправляти на сервер керування відомості про те, що на комп'ютері встановлений екземпляр операційної системи Windows Server 2016.

– Сервер керування відправляє агентів всю логіку моніторингу із установлених пакетів керування, що застосовна до виявлених об'єктів. Наприклад, агент одержить всю логіку моніторингу, що застосовна до Windows Server 2016.

– Агент застосовує логіку моніторингу, таку як правила й монітори, запускає робочі процеси й відправляє дані на сервер керування.

– Якщо у виявлених об'єктах відбуваються зміни, наприклад, додаються або віддаляються додатки, то агент відправляє оновлені відомості на сервер керування, що потім відправляє агентам оновлену логіку моніторингу.

Operations Manager може також виявляти мережеві пристрої, комп'ютери, що працюють під керуванням операційної системи UNIX або Linux, і виконувати спостереження за ними, а також забезпечувати спостереження без агентів.

Взаємодія між агентами й серверами керування

Агент Operations Manager відправляє попередження й дані виявлення на основний сервер керування, що записує ці дані в робочу базу даних. Крім того, агент відправляє дані про події, продуктивність і стан на основний сервер керування, що одночасно записує ці дані в робочу базу даних і в базу даних сховища даних.

Агент відправляє дані відповідно до параметрів розклади для кожного правила й монітора. У випадку оптимізованих правил збору даних дані передаються тільки в тому випадку, якщо вибірка лічильника відрізняється від попередньої вибірки на зазначену величину допуску, наприклад на 10%. Це допомагає скоротити мережевий трафік і обсяг даних, що зберігаються в робочій базі даних.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

Крім того, всі агенти регулярно відправляють пакет даних, називаний пульсом, на сервер керування: за замовчуванням це відбувається кожні 60 секунд. Ціль пульсу складається в перевірці доступності агента й зв'язки між агентом і сервером керування.

3.3 Розробка функціональної схеми

Функціонально система складається з наступних модулів.

Керування агентами SCOM

- SCOM – Включення Agent Proxy на групі агентів.
- SCOM – Зміна статусу ManuallyInstalled для агентів.
- SCOM – Пакетний переклад агентів в Maintenance Mode.
- SCOM – Призначення Primary і Failover серверів на агентах.
- SCOM – Моніторинг серверів поза доменом.
- SCOM – Discovery & Agent Push Install для серверів Forefront TMG.
- SCOM – Автоматизуємо включення Agent Proxy.

Пакети керування/моніторингу

- System Center Core Team – System Center Management Pack Catalog.
- TechNet Wiki – Microsoft Management Packs List.
- The Unofficial System Center Catalog: OM: «Is there a management pack for ... ?».
- SCOM – Резервне копіювання Unsealed MP.
- SCOM R2 – Доступ до MP Online Catalog у консолі SCOM через проксі сервер Squid.

Створення й налаштування моніторів і правил

- SCOM – Створення моніторів для аудита змін доменних груп безпеки.
- SCOM – Дивне поведження моніторів SNMP Probe при операціях порівняння.

– SCOM – Операції порівняння при створенні монітора SNMP Probe (змінні SnmpVarBinds).

Моніторинг ОС і застосунків еко-системи Windows

– SCOM – Моніторинг серверів поза доменом.
– SCOM – Alert: Logical Disk Not Available.
– SCOM – Windows Server – The DNS service on server has stopped running.
– SCOM R2 – Exchange Server 2022 Monitoring Management Pack.
– SCOM R2 – Відключаємо монітор Logical Disk Fragmentation Level Monitor для розділів DPM Replica і DiffArea Volumes.

Моніторинг ОС Unix & Linux

– SCOM R2 – Налаштовуємо базовий моніторинг ОС Linux на прикладі Ubuntu Server 14.04 LTS.
– SCOM R2 – Налаштовуємо базовий моніторинг віртуальної машини HP ZPAR Virtual Service Processor 4.3. 0 (Red Hat Enterprise Linux Server 6.1).

Моніторинг мережевих пристроїв

– SCOM – Скрипт для одержання списку підтримуваних мережевих пристроїв SCOM.
– SCOM – Network Devices Discovery – No Response Ping.
– SCOM R2 – Моніторинг СЗД Hewlett-Packard за допомогою HP StoreFront Manager for Microsoft (HP Storage Management Pack) на прикладі HP ZPAR StoreServ 7200 і MSA P2000 G3.
– SCOM R2 – Моніторинг FC комутаторів Brocade за допомогою HP B-series SAN Network Advisor 12.4 і SCOM.

Розглянемо більш докладно описані вище функціональні блоки.

Включення Agent Proxy на групі агентів

У міру початку використання нових пакетів керування (MP) в SCOM, може виникнути необхідність включення механізму «Agent Proxy» на групі агентів, наприклад, коли ми використовуємо MP для моніторингу AD, нам необхідно включити проксірування для всіх агентів, що мають роль контролерів

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

домену. При великій кількості агентів, зробити це в UI консолі – не найкращий варіант, тому що в нас є Operations Manager Shell.

Зміна статусу **ManuallyInstalled** для агентів

У деяких випадках може виникнути ситуація, коли з якихось причин зроблена ручна установка агентів SCOM на ряд комп'ютерів. Для таких агентів в UI консолі недоступні опції зміни основного сервера керування (Change Primary Management Server), відновлення (Repair) і видалення (Uninstall). Крім цього послуг застосування нових кумулятивних відновлень (CU) такі агенти не будуть переходити в режим примусового відновлення, що знову ж зажадає їхні відновлення «у рукопашну».

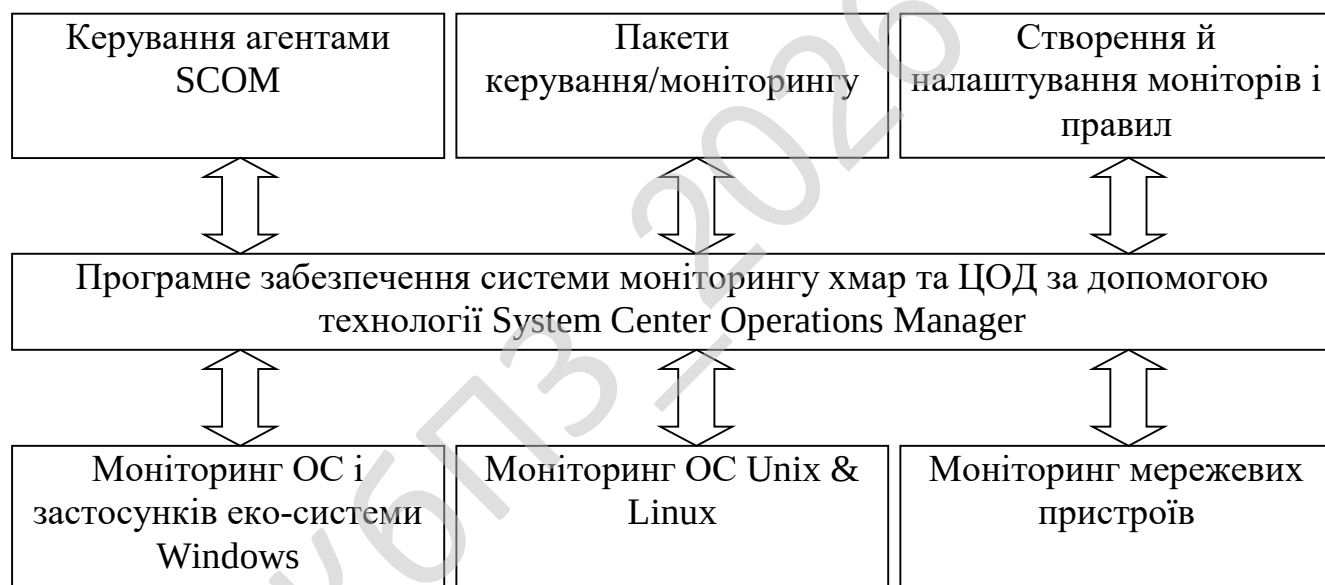


Рисунок 3.2 – Функціональна схема системи

Пакетний переклад агентів в **Maintenance Mode**

В оточенні з більшою кількістю агентів моніторингу SCOM масовий переклад цих агентів у режим обслуговування (Maintenance Mode) через UI консоль може стати досить заморочливим заняттям. На допомогу в такому випадку як завжди приходить Operations Manager Shell.

Призначення Primary і Failover серверів на агентах

У міру розширення інфраструктури SCOM і збільшення серверів керування (Management Server) може виникнути необхідність у форсованому призначенні значень **Primary Management Server** і **Failover Management Server** для агентів, щоб уникнути ситуації коли при недоступному найближчому первинному сервері керування агенти почнуть звертатися на сервера керування на вилучених площадках навантажуючи при цьому WAN-Канали там де це не бажано. Таке поведіння агентів у конфігурації за замовчуванням може бути обумовлено налаштуваннями, які можна бачити в конфігураційному файлі клієнта в кеші коннектора відповідної йому групи керувань. За замовчуванням цей файл – **OpsMgrConnector. Config. xml** можна знайти в папці %ProgramFiles%System Center Operations ManagerHealth Service StateConnector Configuration Cache<Назва групи керування>

Моніторинг серверів поза доменом

Для моніторингу серверів поза доменом (робітники групи або недовірений домен) архітектура SCOM, як використання основного методу, припускає використання додаткової ролі – Gateway Server. Підхід з розгортанням додаткового сервера SCOM з роллю Gateway Server має сенс у тому випадку, якщо кількість об'єктів, що підключаються до системи моніторингу, велико, і Gateway Server виконує в такому випадку роль довіреного посередника між агентами поза доменом й доменним сервером керування. У випадку ж якщо кількість серверів у робочій групі (або ж у недовіреному домені) невелике – можна використовувати інший метод – взаємна автентифікація клієнтів поза доменом (агентом моніторингу) і сервера керування в домені на основі цифрових сертифікатів.

Опис процедури підключення сервера в складі робочої групи до сервера керування SCOM у складі домену.

Основні етапи:

1. Імпортування кореневого сертифіката довіреного центра сертифікації.
2. Створення й установка сертифіката на сервер керування.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

3. Створення й установка сертифіката на сервер у складі робочої групи
4. Експорт сертифіката для подальшого використання в SCOM
5. Включення дозволу ручної установки агентів моніторингу на сервері керування
6. Установка агента моніторингу на сервер у складі робочої групи
7. Імпорт сертифіката в агент моніторингу за допомогою утиліти MOMcertimport
8. Схвалення агента на сервері керування

Discovery & Agent Push Install для серверів Forefront TMG

У процесі процедури виявлення (**Discovery**) для вилученої установки агента (**Push-installation**) SCOM на сервера з **Forefront TMG** ми одержимо помилку через блокування необхідних нам з'єднань.

Автоматизуємо включення Agent Proxy

Велика кількість пакетів керування (**Management Pack**) в **System Center Operations Manager** для коректної роботи механізмів виявлення й моніторингу вимагають наявності включеної опції **Agent Proxy**. При додаванні нового керованого агента в SCOM ця опція за замовчуванням виключена.

Резервне копіювання Unsealed MP

У процесі роботи з **System Center Operations Manager (SCOM)** може створюватися деяка кількість пакетів керування (**Management Pack, MP**), у яких ми зберігаємо перевизначення (**Overrides**) або ж власні монітори (**Monitors**) / правила (**Rules**) і т.п. об'єкти. Час від часу ці кастомізовані налаштування можуть мінятися, і для того щоб організувати їхню схоронність, можна скористатися процедурою експорту цих незапакованих (**Unsealed**) пакетів керувань в **xml** файли. От приклад простого **PowerShell** скрипта, що буде виконувати процедуру експорту MP створюючи структуру каталогів відповідно до назви й номером версії цих MP.

Доступ до MP online catalog у консолі System Center R2 Operations Manager Console через проксі сервер Squid

– Раніше описувалося обхідне рішення для можливості доступу до онлайн каталогу пакетів керування **SCOM** з консолі **System Center R2 Operations Manager Console** з використанням **TMG Client**. Після переходу на проксі-сервер **Squid** ми змушені переглянути зазначене рішення, адаптувавши його під Squid.

Перше, що було виявлено, – описаний раніше спосіб виправлення . config файлу консолі SCOM не дав бажаного результату. Для того, щоб з'ясувати які ще можливі варіанти налаштування даного файлу, скористаємося документом MSDN Library – Network Programming in the .NET Framework – Accessing the Internet Through a Proxy. У моєму випадку робочим варіантом виявився варіант із явною вказівкою проксі-сервера наприкінці секції <configuration>...</configuration> файлу C:\Program Files\Microsoft System Center R2\Operations Manager\Console\Microsoft.EnterpriseManagement.Monitoring.Console.exe.config

Аудит змін доменних груп безпеки

У великих доменних інфраструктурах доменних адміністраторів, що мають трохи, може бути досить актуальним питання аудита змін, вироблених у членстві визначених адміністративних груп.

Для рішення цього завдання скористаємося можливостями SCOM і на прикладі доменної групи “**Domain Admins**” створимо правила, які будуть відслідковувати події, реєструємі в журналі “**Security**” на контролерах домену в момент додавання й видалення користувачів у цю доменну групу безпеки.

У консолі SCOM “**Operations Console**” перейдемо на закладку “**Authoring**” і викличемо майстер створення правил через меню “**Create a new rule...**”

На кроці “**Select a Rule Type**” виберемо “**Alert Generating Rules**” > “**Event Based**” > “**NT Event Log (Alert)**” і обов'язково виберемо **Management Pack** у якому буде створене це правило (по об'єктивних причинах використовувати “**Default Management Pack**” не рекомендується)

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

На кроці **“Rule Name and Description”** уведемо ім'я нашого правила й при бажанні його короткий опис. Ім'я бажано вводити виходячи зі значеннєвого призначення правила так, щоб надалі при необхідності це правило можна було легко знайти серед більше десятка тисяч інших правил за допомогою фільтра. Як об'єкт націлювання **“Rule Target”** у цьому випадку можна буде вибрати визначений клас **“Windows Domain Controller”**, тому що ловити події аудита доменних груп безпеки нам має сенс лише на контролерах домену. Правило залишаємо включеним.

Якщо ж у нас із якоїсь причини існує потреба стежити лише за якимось конкретним сервером, наприклад просто рядовим сервером, то як клас можна вибрати наприклад **“Windows Server Operating System”** і правило створити виключеним, щоб у наслідку включити його тільки для цього конкретного сервера за допомогою перевизначення (Override).

На наступному кроці **“Event Log Name”** вибираємо ім'я журналу, що цікавить нас, **“Security”**

На кроці **“Build Event Expression”** нам потрібно задати умови по яких ми будемо відловлювати потрібні нам події. У параметр **“Event ID”** уведемо ідентифікатор події. Наш приклад має на увазі що як контролери домену використовуються сервера з Windows Server і тому як ідентифікатор події, що реєструється в журналі безпеки в момент додавання облікового запису користувача в доменну групу безпеки ми будемо використовувати – **4728**. От приклад даних доступних усередині самої події

Потрібно враховувати, що подія з таким кодом реєструється при зміні будь-якої глобальної групи безпеки й тому нам потрібно додати умову, що буде відбирати лише ті події, у яких як третій параметр є присутнім назва групи, що цікавить нас, – **“Domain Admins”**

В події в параметрі 5 зберігається SID групи безпеки, членство якої змінилося ... і я по початку, подумавши що використовувати маску SID визначених груп буде більш правильно, чим просте ім'я групи – пустився у

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

вправи з регулярними вираженнями й підстановочними знаками, тому що відомо що SID групи “Domain Admins” у будь-якому домені має вигляд 1-5-21-%ID домену%-512. З подивом для себе виявив що різне звертання до параметрів події видає різні результати:

\$Data/Params/Param[5]\$ – повертає ім'я а не оригінальне значення SID

\$Data/EventData/DataItem/EventData/Data[5]\$ – повертає вихідне значення SID

На жаль мені так і не вдалося досягти логікові заповнення параметрів Expression і змусити цей механізм працювати так, щоб в умові відбору брало участь не перетворене значення ім'я групи а його SID так ще щоб все це й укладалося в який-небудь wildcard типу 1-5-21-%-512... видиме я не так багато випив зеленого чаю в позі лотоса як напевно це робили розроблювачі коли ліпили це “чудо”.

Ідемо далі...На кроці “**Configure Alerts**” задаємо ім'я події яке буде з'являтися на консолі SCOM і із цим же заголовком ми будемо одержувати оповіщення наприклад у пошту. В опис включаємо другий параметр події у вигляді \$Data/Params/Param[2]\$ який поверне нам ім'я користувача перетворене з SID доданого в групу безпеки а також третій параметр \$Data/Params/Param[3]\$ у якому буде назва групи, що...ну й додаємо параметр \$Data/EventDescription\$ який поверне нам основний уміст опису з тіла події. Пріоритет і вага алерту виставляємо “по смаку”. На цьому створення закінчене. Перед тим як тестувати нове правило дочекаємося коли на контролерах домену в балці SCOM з'являться події з кодом **1200**, **1201** і **1210** мовці про те, що нова версія MP з нашими змінами отримана й застосувалася до конфігурації.

Після цього можна спробувати змінити состав групи доменних адміністраторів і переконатися в тому що на SCOM реєструється відповідне повідомлення. Наприклад такий от досить наочний і інформативний вид буде мати лист із повідомленням у нашій випадку:

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

За аналогією з описаною процедурою створюємо окреме правило, що буде відловлювати події з кодом **4729**, які реєструються на контролерах домену в журналі безпеки при видаленні члена з доменної групи безпеки, що цікавить нас.

Дивне поведження моніторів SNMP Probe при операціях порівняння.

Якщо виникає необхідність моніторингу мережевих пристроїв підтримуючий протокол SNMP, – в **SCOM** ми можемо через консоль **“Operations Console”** на закладці **“Authoring”** за допомогою помічника **“Create a unit monitor”** створити **SNMP Probe Based Monitor**, що буде через зазначені нами інтервали часу виконувати до мережевого пристрою запит певних значень OID і на підставі отриманих даних змінювати статус цього мережевого пристрою за принципом Healthy/Unhealthy

Однак на практиці ви можете зіткнутися із ситуацією коли створений вами монітор поводить себе не зовсім тому що ви цього від нього очікуєте. Наприклад ви створили монітор, що раз у кілька минут опитує джерело безперебійного живлення (ДБЖ) у серверній на предмет значення конкретного OUD вхідної напруги, що повертається поточне значення. Якщо цей OID опитати по SNMP за допомогою будь-якого стороннього застосунку (наприклад MIB Browser) то можливо ми побачимо що тип повертаемого значення – цілочисельне значення – **“Integer”**, у той час як візард SCOM **“Create a unit monitor”** за замовчуванням задає тип значення – рядок – **“String”**.

Відповідно одержуване від пристрою значення буде перетворюватися в рядок і вже надалі піддаватися наприклад некоректним операціям порівняння які можуть бути присутнім у нашій моніторі, що саме по собі й стає причиною непояснених спрацьовувань алертів.

Для виправлення цієї ситуації нам необхідно вивантажити Management Pack у якому збережений наш SNMP Probe Based Monitor в XML файл і знайти в ньому секції **Expression** і в тегах **XPathQuery Type** і **Value Type** у які задається тип значення...

SNMPProbeMonitor:Операції порівняння при створенні монітора (змінні SnmpVarBinds)

При створенні монітора (**SNMP Probe**) через UI консолі **System Center Operations Manager Console** ми як правило оперуємо отриманими по SNMP значеннями в операціях порівняння з якимись конкретно заданими величинами. В інтернеті можна знайти масу прикладів того як створити монітор, що одержує по SNMP значення певного OID і порівнює це значення з конкретно заданої нами величиною, наприклад по SNMP одержуємо значення поточної вхідної напруги на ДБЖ і порівнюємо його з конкретними значеннями припустимих границь, заданих у моніторі. Але як бути якщо їсти бажання зробити монітор більше універсальним?

Наприклад якщо говорити про опитування тих же ДБЖ, то значення верхньої й нижньої границі припустимого вхідної напруги як правило розрізняється навіть у рамках одного вендора устаткування в розрізі моделей ДБЖ. Звичайно можна створити окремий монітор для кожної моделі ДБЖ, але знаючи те, що самі значення верхньої й нижньої границі вхідної напруги зберігаються в налаштуваннях ДБЖ і можуть бути отримані по SNMP таке рішення ставати неоптимальним. Так народилася думка про те, що можна спробувати в рамках одного монітора запросити з ДБЖ по SNMP відразу трьох значення – поточна напруга, верхню й нижню границі ... і потім зрівняти їх. Тому що пізнаннями в області рукопашного написання MP я не володію, була почата спроба зробити це через майстер створення монітора в консолі SCOM. Власне кажучи, спроба успіхом не увінчалася, тому що виявилось що подібного роду операції порівняння неможливо сконструювати через UI.

Моніторинг серверів поза доменом

Для моніторингу серверів поза доменом (робітники групи або недовірений домен) архітектура SCOM, як використання основного методу, припускає використання додаткової ролі – Gateway Server. Підхід з розгортанням додаткового сервера SCOM з роллю Gateway Server має сенс у тому випадку,

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		40

якщо кількість об'єктів, що підключаються до системи моніторингу, велико, і Gateway Server виконує в такому випадку роль довіреного посередника між агентами поза доменом й доменним сервером керування. У випадку ж якщо кількість серверів у робочій групі (або ж у недовіреному домені) невелике – можна використовувати інший метод – взаємна автентифікація клієнта поза доменом (агентами моніторингу) і сервера керування в домені на основі цифрових сертифікатів.

Опис процедури підключення сервера в складі робочої групи до сервера керування SCOM у складі домену.

Основні етапи:

1. Імпортування кореневого сертифіката довіреного центра сертифікації.
2. Створення й установка сертифіката на сервер керування.
3. Створення й установка сертифіката на сервер у складі робочої групи
4. Експорт сертифіката для подальшого використання в SCOM
5. Включення дозволу ручної установки агентів моніторингу на сервері керування
6. Установка агента моніторингу на сервер у складі робочої групи
7. Імпорт сертифіката в агент моніторингу за допомогою утиліти MOMcertimport
8. Схвалення агента на сервері керування

Alert: Logical Disk Not Available

У деяких випадках на SCOM може з'явиться попередження, що збиває з користі, про неприступність того або іншого логічного диска.

Це може бути викликане тим, що WMI клас **Win32_LogicalDisk** повертає значення **TRUE** для властивості **VolumeDirty** для проблемного логічного диска при опитуванні ОС скриптом SCOM. Для того щоб переконатися в тому, що це дійсно так, використовуємо утиліту **wbemtest** для підключення до простору імен WMI проблемного комп'ютера

SCOM R2 & Exchange Server 2022 Monitoring Management Pack

Налаштовуючи нову інфраструктуру **System Center R2 Operations Manager (SCOM)** і встановлюючи пакети керування **Management Pack (MP)** у черговий раз довелося згадувати підводні камені **Exchange Server 2022 Monitoring Management Pack**.

Як уже давно відомо, основна проблема цього MP полягає в тому, що служба кореляції **Microsoft Exchange Monitoring Correlation**, що додається на сервер SCOM у процесі розпакування/установки MP, має залежність від ролі **RMS**, що мала місце бути в **SCOM**. Починаючи з SC архітектура OM значно перероблена, і роль RMS залишилася лише у вигляді емуляції (**RMS Emulator**) з метою сумісності зі старими MP.

Зважаючи на те, що **Exchange Server 2022** ще багато де в побуті, а розвиток відповідного MP уже перестало “іти в ногу” з розвитком System Center, вирішив написати маленьку шпаргалку про те, як подружити цей MP з новою версією SCOM.

System Center R2 Operations Manager – Відключаємо монітор Logical Disk Fragmentation Level Monitor для розділів DPM Replica і DiffArea Volumes

Пакет керування/базового моніторингу (**Management Pack**) ОС **Windows Server** для **System Center R2 Operations Manager (SCOM)** у конфігурації за замовчуванням виконує виявлення й моніторинг стану фрагментації всіх логічних дисків системи. Функціональність це безумовно корисна, але є ситуації, коли потрібно відключити такого роду моніторинг (і відповідно його оповіщення) для певних систем, як наприклад сервери **System Center Data Protection Manager (DPM)**, у яких може бути безліч логічних розділів диска, створюваних механізмами DPM і “живучих за своїми законами”.

Налаштовуємо базовий моніторинг віртуальної машини HP 3PAR Virtual Service Processor 4.3. 0 (Red Hat Enterprise Linux Server 6.1) на базі System Center R2 Operations Manager

У минулій замітці ми згадали тему інтеграції віртуальної машини **HP 3PAR Virtual Service Processor (VSP) 4.3. 0** в існуючу інфраструктуру **System**

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		42

Center R2. У цьому розділі ми продовжимо цю тему, розглянувши процес установки агента **Operations Manager (SCOM)** у віртуальну машину VSP для забезпечення базового моніторингу операційної системи **Red Hat Enterprise Linux Server (RHEL) 6.1**. Налаштування SCOM і керованої системи на базі ОС Linux уже розглядалися раніше на прикладі Ubuntu Server 14.04 LTS. Тому щоб повністю не повторювати минулу замітку, постараюся викласти ті дії, які будуть відрізнятися від попереднього опису в силу специфіки використання іншої версії Linux з обліком “закарлюк” VSP.

SCOM Network Devices Discovery – No Response Ping

У процесі процедури виявлення мережевих пристроїв (**Network Devices Discovery**) в **SCOM** можливо одержати статус **No Response Ping**, тобто нібито пристрій не відповідає на запити **ICMP**, хоча насправді пристрій доступно.

Це може бути причиною того, що на сервері керування SCOM запущений **Windows Firewall**, але не настроєні потрібні правила. У процесі установки SCOM створює ряд визначених правил, деякі з яких перебувають у виключеному стані за замовчуванням. Тобто нам потрібно відкрити оснащення керування Windows Firewall і включити розв'язні правила для процесу **MonitoringHost.exe** (які виключені за замовчуванням) у розділі правил для вхідних підключень -**Inbound Rules**...а також у розділі правил для вихідних підключень – **Outbound Rules**

Послу цього спробуємо заново запустити процедуру виявлення й переконатися в тому, що наші мережеві пристрої успішно додані в SCOM

System Center R2 Operations Manager – Моніторинг ЦЗД Hewlett-Packard за допомогою HP StoreFront Manager for Microsoft (HP Storage Management Pack) на прикладі HP 3PAR StoreServ 7200 і MSA P2000 G3

Всі сучасні моделі систем зберігання даних (ЦЗД) від **Hewlett-Packard**, з якими мені доводилося зіштовхуватися на практиці, мають убудований функціонал оповіщень, що дозволяє організувати базовий моніторинг подій що відбуваються на цьому устаткуванні. Крім цього, маючи в інфраструктурі функціонуючий **System Center R2 Operations Manager (SCOM)**, для цілей

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		43

моніторингу СЗД можна скористатися пакетом керування **HP Storage Management Pack (MP)** із состава **HP StoreFront Manager for Microsoft**. У цьому розділі я розгляну приклад налаштування моніторингу СЗД наявних під рукою моделей – **HP 3PAR StoreServ 7200** і **MSA P2000 G3** в SCOM.

Завантажити відповідний пакет керування для SCOM можна по посиланню **HP StoreFront Manager for Microsoft (HP Storage Management integration with MS System Center)**. Поточна версія пакета керування **4.1** підтримує виявлення й моніторинг наступного устаткування:

- HP 3PAR StoreServ Storage System.
- HP Storage ESL E-Series, EML E-Series, ESL 6000/9000/1 2000, MSL Series.
- HP P2000 MSA G3 and HP MSA 2040 Storage.
- HP Storage EV A P6000, 4x00, 6x00, 8x00, P6300, P6500, and HP P6000 Command View EVA Software.
- HP Storage XP 2 4000 and P9500 Arrays.
- HP Store Virtual Storage with SANi 9.5, 1 0.5, and 1 1.0.
- HP E5000 Messaging System.
- HP StoreEasy 5000 Storage System.
- HP StoreOnce Backup.
- HP StoreEasy 1000 and HP StoreEasy 3000 Storage.
- HP 3PAR StoreServ File Controller.

Моніторинг FC комутаторів Brocade за допомогою HP B-series SAN Network Advisor 12.4 і Microsoft System Center R2 Operations Manager

У цьому розділі ми розглянемо приклад налаштування моніторингу оптичних комутаторів фірми **Brocade** за допомогою **Microsoft System Center R2 Operations Manager (SCOM)**.

Пропонована Brocade архітектура моніторингу комутаторів така, що нам буде потрібно на виділеному сервері встановити й налаштувати ПЗ **Brocade Network Advisor**, підключити до цього ПЗ наші комутатори, і потім інтегрувати

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		44

пакет керування (**Management Pack**), що поставляється в комплекті із цим ПЗ, є одним з наших сервером керування SCOM. Таким чином, агент SCOM, встановлений на цьому виділеному сервері буде звітувати перед сервером керування SCOM виходячи з інформації, одержуваної з Brocade Network Advisor.

У моєму прикладі комутатори Brocade представлені у вигляді ребрендингових моделей Hewlett-Packard **StorageWorks**, тому замість нативного пакета Brocade Network Advisor я буду використовувати пропонуваний компанією Hewlett-Packard пакет **HP B-series SAN Network Advisor**. Як я зрозумів, ці два пакети з технічної точки зору нічим принциповим друг від друга не відрізняються (HP додає в нативний пакет Brocade своя ліцензійна угода).

Відразу хочу відзначити той факт, що HP B-series SAN Network Advisor існує як у вигляді платної ліцензуємої редакції, так і у вигляді безкоштовної редакції, що має ряд обмежень. Одним з таких обмежень, досить істотним у контексті теми цієї замітки, є відсутність можливості використання процедури інтеграції Network Advisor з SCOM. Саме тому нам оведеться використовувати **пробну повнофункціональну 120-денну версію (Trial)** HP B-series SAN Network Advisor.

3.4 Розробка діаграми процесів

Розглянемо розроблену діаграму процесів яка зображена на рисунку 3.3. Основна будова діаграми процесів полягає у графічному представленні складу сукупностей даних, що характеризуються як співвідношення різних частин кожної з сукупностей.

Склад статистичної сукупності графічно може бути представлений як за допомогою абсолютних, так і відносних показників. Графічне зображення складу сукупності по абсолютними і відносними показниками сприяє проведенню більш глибокого аналізу і дозволяє проводити аналіз системи.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		45



Рисунок 3.3 – Діаграма взаємодії процесів

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування).

Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи.

Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

При детальному її розгляді можна побачити як саме проходить взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграми потоків даних містять чотири типи елементів:

- Зовнішні по відношенню до системи сутності.
- Потoki даних між елементами трьох попередніх типів.

– Процеси які являють собою трансформацію даних в рамках описуваної системи.

– Сховища даних (репозиторії).

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

К6ПЗ_2026

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Розглянемо реалізацію бакалаврської дипломної роботи. Були проведені розрахунки і підібрані набори тестових даних для перевірки правильності реалізації проектних рішень.

Було створено блок-схеми роботи системи. Перед їх розглядом необхідно провести роз'яснення який саме тип блок-схем використовується. Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо.

Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму. Блок-схеми показують весь процес роботи системи з підсистемами та частково доказують правильність вибраних проектних рішень. Тому від точності і детальної блок-схеми залежить результат всієї програми. При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об'єктно-орієнтована що вимагає високого рівня декомпозиції задач на класи.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

підсистеми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ.

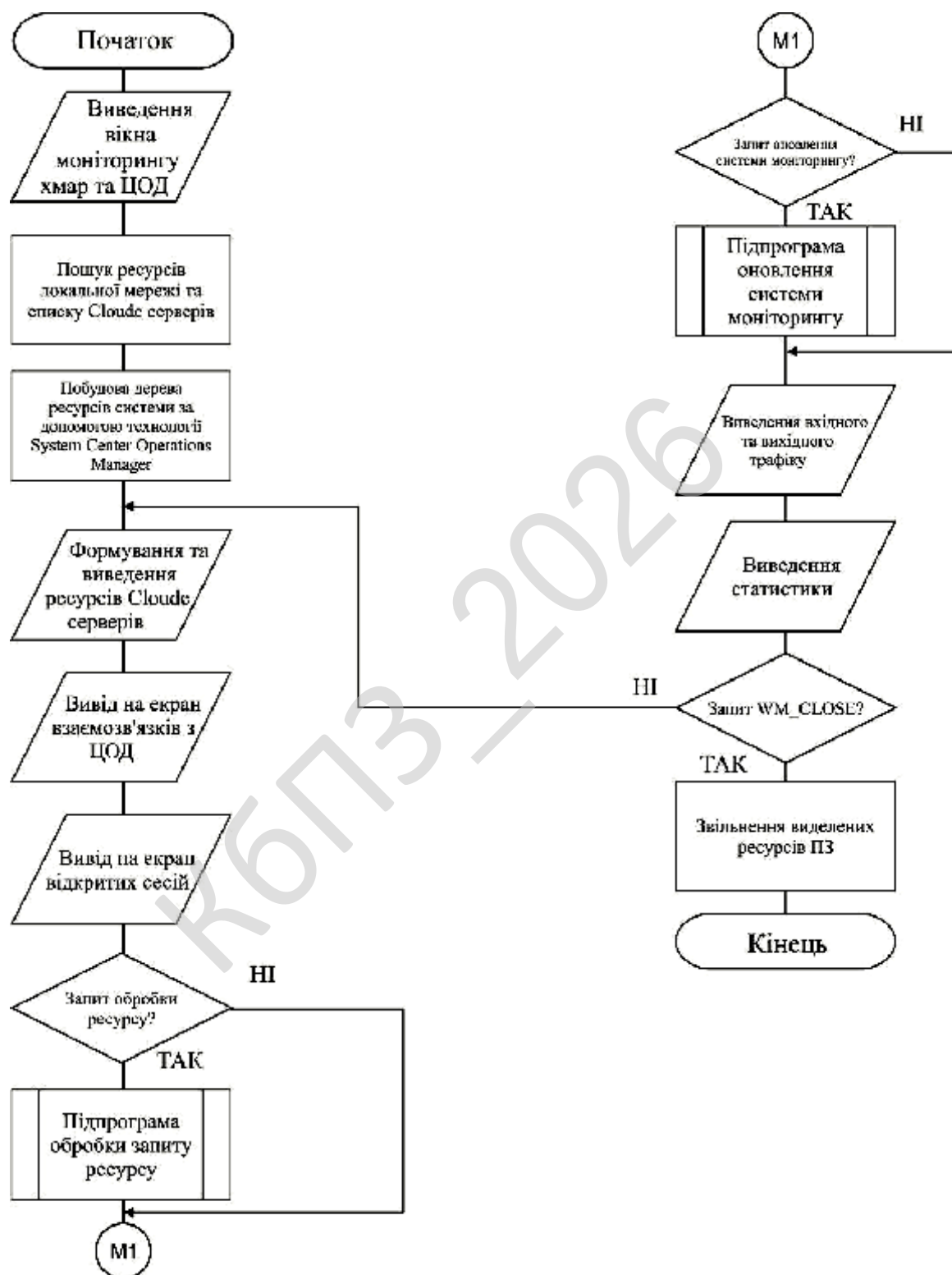


Рисунок 4.1 – Блок-схема основної програми

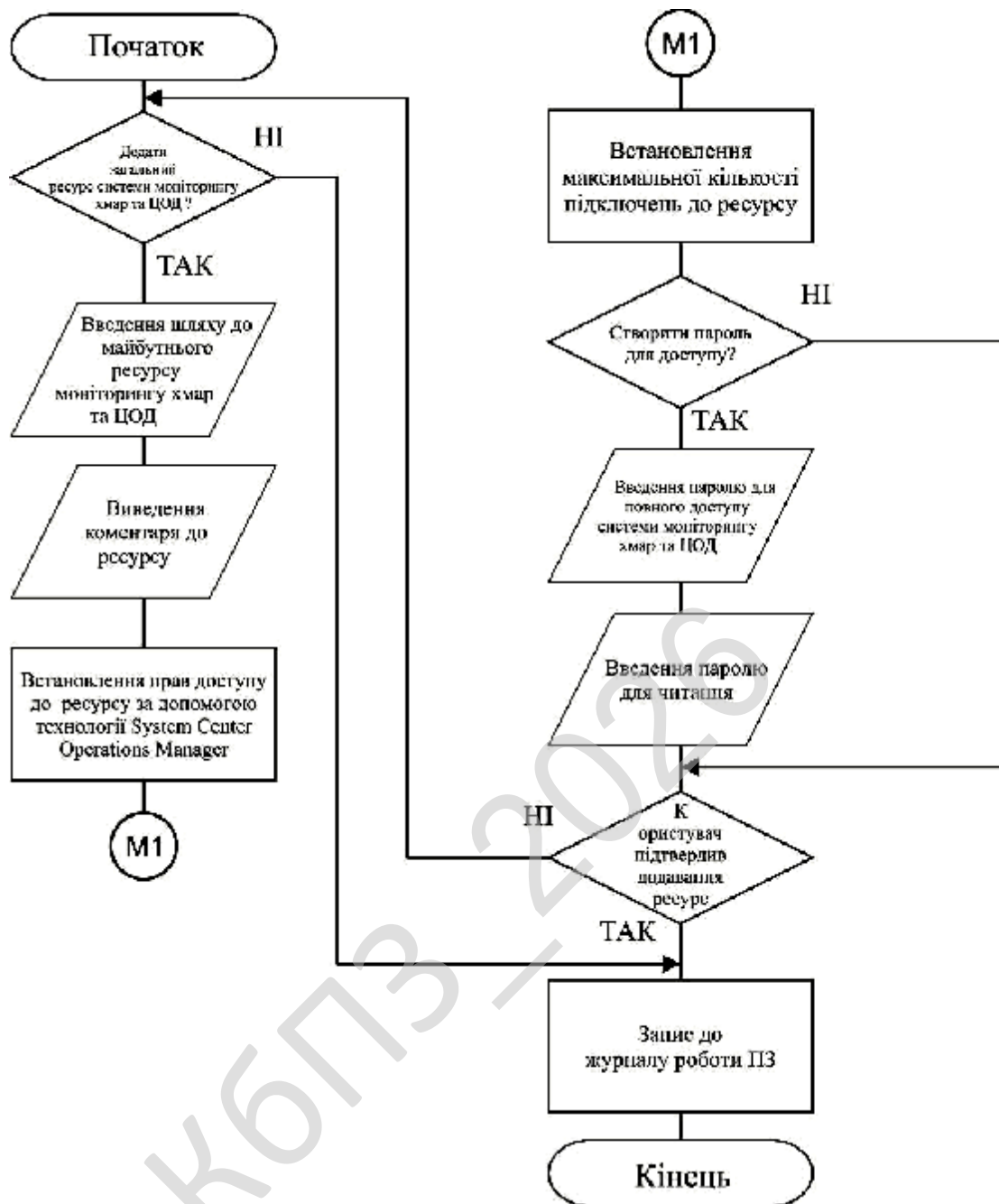


Рисунок 4.2 – Блок-схема роботи підпрограми

При роботі підпрограми виконується основний функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

Microsoft SQL Server – комерційна система керування базами даних, що розповсюджується корпорацією Microsoft.

Мова, що використовується для запитів – Transact-SQL, створена спільно Microsoft та Sybase.

Transact-SQL є реалізацією стандарту ANSI/ISO щодо структурованої мови запитів SQL із розширеннями. Використовується як для невеликих і середніх за розміром баз даних, так і для великих баз даних масштабу підприємства. Багато років вдало конкурує з іншими системами керування базами даних.

Походження

Базовий код MS SQL Server (до версії 7.0) ґрунтувався на коді Sybase SQL Server. Це дозволило Microsoft вийти на ринок баз даних для підприємств, де конкурували Oracle, IBM, і, пізніше, сама Sybase. Microsoft, Sybase і Ashton-Tate спочатку об'єдналися для створення і випуску на ринок першої версії програми, що отримала назву SQL Server 1.0 для OS/2 (близько 1989 року), яка фактично була еквівалентом Sybase SQL Server 3.0 для Unix, VMS та ін. Microsoft SQL Server 4.2 був випущений у 1992 році та входив до складу операційної системи Microsoft OS/2 версії 1.3.

Офіційний реліз Microsoft SQL Server версії 4.21 для ОС Windows NT відбувся одночасно з релізом самої Windows NT (версії 3.1). Microsoft SQL Server 6.0 був першою версією SQL Server, створеною виключно для архітектури NT і без участі в процесі розробки Sybase.

До того часу, як вийшла на ринок ОС Windows NT, Sybase і Microsoft розійшлися та створювали вже власні моделі цього програмного продукту. Microsoft намагалася отримати виняткові права на всі версії SQL Server для Windows. Пізніше Sybase змінила назву свого продукту на Adaptive Server Enterprise щоб уникнути плутанини з Microsoft SQL Server. До 1994 року Microsoft отримала від Sybase три повідомлення про авторські права як натяк на походження Microsoft SQL Server.

Після розділення компанії зробили декілька самостійних релізів програм. SQL Server 7.0 був першим сервером баз даних зі справжнім графічним інтерфейсом адміністрування. Для усунення претензій з боку Sybase у порушенні

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

авторських прав, весь успадкований код в сьомій версії був переписаний. Це забезпечило також й успіх SQL Server 2000, який був першою редакцією, орієнтованою на архітектуру IA-64.

Протягом подальших шести років корпорація Microsoft працювала над вдосконаленням вже існуючої версії SQL Server 2000 доки не збудувала зручнішу систему Microsoft SQL Server 2005. Були вдосконалені продуктивність, клієнтські інструменти інтегрованого середовища розробки, а також у декількох додаткових системах, що встановлюються разом із SQL Server 2005. Змінено: інструментарій процесів керування сховищами даних (SQL Server Integration Services або SSIS), сервер звітів, сервер OLAP та інтелектуального аналізу даних (Analysis Services), а також декілька технологій повідомлень, особливо Service Broker та Notification Services.

SQL Server

SQL Server 2012 включає низку вдосконалень для роботи з критичними бізнес-застосунками і бізнес-аналітикою як в традиційних дата-центрах, так і в приватних, публічних і гібридних хмарних середовищах. Серед нових можливостей SQL Server 2012 виділяються SQL Server AlwaysOn (рішення підтримки високого рівня доступності даних та аварійного відновлення), xVelocity (технологія збільшення продуктивності сховищ даних та програм бізнес-аналітики), нові рішення в області візуалізації PowerPivot і PowerView для створення звітів і аналітичних програм з Excel і SharePoint, покращені інструменти для інтеграції даних і управління ними, включаючи SQL Server Data Quality Services і Master Data Services, нова семантична модель бізнес-аналітики та інструмент для адміністраторів баз даних і розробників застосунків SQL Server Data Tools.

Також Microsoft зробила значні інвестиції в області Big Data, а саме в інтеграцію SQL Server і популярних інструментів для бізнес-аналітики з неструктурованою інформацією.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		52

Microsoft робить SQL Server доступним у різноманітних варіантах, які різняться наборами властивостей в залежності від цілей кінцевого користувача. Це такі редакції як:

SQL Server Compact Edition (SQL CE) – Компактний механізм бази даних. Завдяки малому обсягу (2 Мб для DLL) має зменшені властивості у порівнянні з іншими варіантами. Розмір бази даних обмежений 4 Гб і не може використовуватися як служба Windows.

SQL Server Express Edition – Раніше відомий під назвою MSDE (Microsoft SQL Server Desktop Engine), Microsoft SQL Server Express є вільно поширюваною версією SQL Server. Дана версія має деякі технічні обмеження, також відсутні графічні інструменти адміністрування. Такі обмеження роблять її непридатною для розгортання великих баз даних. В основному вона використовується у застосунках, при проектуванні, або для самостійного вивчення. Розмір бази даних обмежений 4 Гб, розмір пам'яті, що може бути адресованою – 1 Гб, підтримує лише один процесор.

SQL Server Workgroup Edition – SQL Server Workgroup Edition включає функціональність ядра бази даних, але не включає додаткові сервіси.

SQL Server Standard Edition – SQL Server Standard edition включає механізми ядра бази даних, а також автономні сервіси. Відрізняється від варіанту Enterprise Edition тим, що підтримує менше активних вузлів та не включає деякі функції збільшення продуктивності.

SQL Server Enterprise Edition – SQL Server Enterprise Edition – це повнофункціональна версія SQL Server

SQL Server Developer Edition – SQL Server Developer Edition включає ті самі функції, що й SQL Server Enterprise Edition, але містить обмеження щодо використання його лише для розробки та тестування.

Його ліцензія не дозволяє використання як виробничого сервера.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		53

Функціональність

Microsoft SQL Server як мову запитів використовує версію SQL, що отримала назву Transact-SQL (скорочено T-SQL), яка є реалізацією SQL-92 (стандарт ISO для SQL) з багатьма розширеннями. T-SQL дозволяє використовувати додатковий синтаксис процедур, що зберігаються і забезпечує підтримку транзакцій (взаємодія бази даних з керуючим застосунком).

Microsoft SQL Server та Sybase ASE для взаємодії з мережею використовують протокол рівня застосунка під назвою Tabular Data Stream (TDS, протокол передачі табличних даних).

Microsoft SQL Server також підтримує Open Database Connectivity (ODBC)—інтерфейс взаємодії застосунків з СУБД. Версія SQL Server 2005 надає можливість підключення користувачів через веб-сервер-сервіси, що використовують протокол SOAP. Це дозволяє клієнтським програмам, не призначеним для Windows, кроссплатформенно з'єднуватися з SQL Server. Microsoft також випустила сертифікований драйвер JDBC, що дозволяє застосункам під керування Java (таким як BEA і IBM Websphere) з'єднуватися з Microsoft SQL Server 2000 і 2005.

SQL Server підтримує дзеркалювання та кластеризацію баз даних. Кластер серверу SQL—це сукупність однаково конфігурованих серверів; така схема допомагає розподілити робоче навантаження між декількома серверами. Усі сервери мають одне віртуальне ім'я, а дані розподіляються за IP-адресами машин кластеру протягом робочого циклу. Також у разі відмови або збою на одному з серверів кластеру доступне автоматичне перенесення навантаження на інший сервер.

SQL Server підтримує надлишкове дублювання даних за трьома сценаріями:

1. Знімок: Виконується «знімок» бази даних, який сервер відправляє одержувачам.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		54

2. Історія змін: Всі зміни бази даних безперервно передаються користувачам.

3. Синхронізація з іншими серверами: Бази даних декількох серверів синхронізуються між собою. Зміни усіх баз даних відбуваються незалежно на кожному сервері, а під час синхронізації відбувається звірка даних. Дублювання такого типу передбачає можливість вирішення протиріч між базами даних.

SQL Server 2005 має вбудовану підтримку .NET Framework. Завдяки цьому, процедури бази даних, що зберігаються, можуть бути написані на будь-якій мові платформи .NET з використанням повного набору бібліотек, доступних для .NET Framework. На відміну від інших процесів, .NET Framework виділяє додаткову пам'ять і будує засоби керування SQL Server, не використовуючи вбудовані засоби Windows. Це підвищує продуктивність порівняно із загальними алгоритмами Windows, оскільки алгоритми розподілу ресурсів спеціально налагоджені для використання у структурах SQL Server.

Система керування базами даних (СКБД, Database Management System, DBMS) – набір взаємопов'язаних даних (база даних) і програм для доступу до цих даних. Надає можливості створення, збереження, оновлення та пошуку інформації в базах даних з контролем доступу до даних.

Першим поколінням СКБД прийнято вважати ієрархічні й мережеві системи. Ці системи отримали широке поширення в 1970-х роках, а першою комерційною системою цього типу була система IMS компанії IBM.

У 1980-х роках ці системи були витіснені системами другого покоління – повсюдно використовуваними і донині реляційними СКБД. У цих системах використовувалися непроцедурні мови управління даними (SQL) і передбачався значний ступінь незалежності даних.

Реляційні системи внесли значні удосконалення в управління даними: графічний користувацький інтерфейс (GUI), клієнт-серверні застосунки, розподілені бази даних, паралельний пошук даних та інтелектуальний аналіз даних.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		55

Але вже до кінця 1980-х років існуюча тоді реляційна модель перестала задовольняти розробників в силу низки обмежень. Відповіддю на зростаючу складність програм баз даних стали два нових напрямки розвитку СКБД: об'єктно-орієнтовані СКБД і об'єктно-реляційні СКБД.

У 1991 був утворений консорціум ODMG, основною метою якого стало вироблення промислового стандарту об'єктно-орієнтованих баз даних. Між 1993 та 2001 роками ODMG опублікувала п'ять ревізій своїх специфікацій. Остання версія стандарту має індекс 3.0, після чого група розпустилася. До кінця 1990-х існувало близько десяти компаній, що виробляли комерційні продукти, що позиціонуються на ринку як ООСКБД. Найбільш відомими системами даного класу стали Objectivity, Versant виробництва однойменних компаній, а також СКБД Jasmine, випущена компанією CA. Незважаючи на переваги, що дозволяють ефективніше вирішувати певний ряд завдань, об'єктно-орієнтовані системи так і не змогли завоювати значущу частку ринку СКБД, залишившись «нішевим» продуктом.

Постачальниками традиційних реляційних СКБД також була проведена значна робота з об'єднання об'єктно-орієнтованих і реляційних систем. Розробники постаралися розширити мову SQL, щоб включити в неї концепції об'єктно-орієнтованого підходу, зберігаючи переваги реляційної моделі (об'єктні розширення мови SQL були зафіксовані в стандарті SQL:1999).

Основний принцип – це еволюційний розвиток можливостей СКБД без поломки попередніх підходів та зі збереженням наступності з системами попереднього покоління.

Поняття СКБД третього покоління, якими, власне кажучи, і є об'єктно-реляційні СКБД, з'явилося після опублікування групою відомих фахівців в області баз даних «Маніфесту систем баз даних третього покоління». Основні принципи СКБД третього покоління, позначені в маніфесті:

Крім традиційних послуг з управління даними, СКБД третього покоління повинні забезпечити підтримку розвиненіших структур об'єктів і правил.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

Розвинутіша структура об'єктів характеризує засоби, необхідні для зберігання і маніпулювання нетрадиційними елементами даних (тексти, просторові дані, мультимедіа).

СКБД третього покоління повинні включити в себе СКБД другого покоління. Системи другого покоління внесли вирішальний вклад у двох областях – непроцедурний доступ за допомогою мови запитів SQL і незалежність даних. Ці досягнення обов'язково повинні враховуватися в системах третього покоління.

СКБД третього покоління повинні бути відкриті для інших підсистем. Це включає оснащення різноманітними інструментами підтримки прийняття рішень, доступом з багатьох мов програмування, інтерфейсами до існуючих популярних систем і бізнес-застосунків, можливістю запуску програм з бази даних на іншій машині і розподілені СКБД. Весь набір інструментів і СКБД має ефективно функціонувати на різноманітних апаратних платформах з різними операційними системами.

Крім того, СКБД, що розраховує на широку сферу застосування, повинна бути оснащена мовою четвертого покоління (4GL).

У середині 1990 років було лише кілька дослідних прототипів СКБД, які поєднали найкращі риси реляційних і об'єктно-орієнтованих СКБД. Першим комерційним продуктом, якому були властиві об'єктно-реляційні риси, став Universal Server компанії Informix(згодом була поглинена IBM). В даний час більшість цих ідей вже втілено в реальних комерційних рішеннях, в тому числі і в продуктах основних постачальників СКБД (Oracle Database і IBM DB2).

Розвиток індустрії систем керування базами даних базується на значних фундаментальних наукових дослідженнях. Найчастіше, між самими дослідженнями та їхньою конкретною реалізацією в прикладних рішеннях минають роки, а іноді й десятиліття. Роботу в області управління даними проводять як університетські дослідницькі групи (MIT, Berkeley), так і центри розробок основних постачальників СКБД (Oracle, IBM, Microsoft). Інвестування в

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

управління даними – це довгострокове, і разом з тим, вигідне вкладення коштів. В даний час дослідники мають у своєму розпорядженні засоби, що дозволяють ефективно реалізувати найскладніші запити, що маніпулюють терабайтами й петабайтами різних даних.

Основними тенденціями, які дали привід для проведення різних масштабних досліджень в області баз даних стали:

Експонентний ріст даних. Обсяг даних, у тому числі синтетичних, що генеруються автоматизованими системами, значно зріс. Збільшилося і число прикладних областей, в яких вимагається обробка великих обсягів даних. До таких областей тепер відносяться не тільки традиційні корпоративні програми, пошук у веб, але також і наукові дослідження, обробка природних мов, аналіз соціальних мереж тощо.

Значне ускладнення структур використовуваних даних. Прості види даних у вигляді чисел і символьних рядків стали доповнятися численною мультимедійною інформацією, просторовими, процедурними даними та великою кількістю інших складних форматів.

Широке поширення дешевих високопродуктивних апаратних засобів. Щорічно ми спостерігаємо зростання обчислювальних можливостей мікропроцесорів, збільшення ємності і зниження вартості доступних і зручних в експлуатації пристроїв дискової оперативної пам'яті.

Активний розвиток засобів комунікації та «всесвітньої павутини» World Wide Web. WWW стає єдиним інформаційним середовищем, що пронизує весь світ і об'єднує величезне число користувачів та електронних пристроїв.

Поява нових важливих областей застосування СКБД. У першу чергу, це пов'язано з інтелектуальним аналізом даних, сховищами даних, а останнім часом – з паралельними обчисленнями і хмарними технологіями.

Основні характеристики СКБД

1. Контроль за надлишковістю даних.
2. Несуперечливість даних.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		58

3. Підтримка цілісності бази даних (коректність та несуперечливість).
4. Цілісність описується за допомогою обмежень.
5. Незалежність прикладних програм від даних.
6. Спільне використання даних.
7. Підвищений рівень безпеки.

Можливості СКБД

1. Дозволяється створювати БД (здійснюється за допомогою мови визначення даних DDL (Data Definition Language)).
2. Дозволяється додавання, оновлення, видалення та читання інформації з БД (за допомогою мови маніпулювання даними DML, яку часто називають мовою запитів)
3. Можна надавати контрольований доступ до БД за допомогою: Системи забезпечення захисту, яка запобігає несанкціонованому доступу до БД; Системи керування паралельною роботою прикладних програм, яка контролює процеси спільного доступу до БД; Система відновлення – дозволяє відновлювати БД до попереднього несуперечливого стану, що був порушений в результаті збою апаратного або програмного забезпечення.

Основні компоненти середовища СКБД

1. Апаратне забезпечення.
2. Програмне забезпечення.
3. Дані.
4. Процедури – інструкції та правила, які повинні враховуватись при проектуванні та використанні БД.
5. Користувачі: адміністратори даних (керування даними, проектування БД, розробка алгоритмів, процедур) та БД (фізичне проектування, відповідальність за безпеку та цілісність даних); розробники БД; прикладні програмісти; кінцеві користувачі.

Архітектура СКБД

Існує трирівнева система організації СКБД ANSI-SPARC, при якій існує незалежний рівень для ізоляції програми від особливостей представлення даних на нижчому рівні.

Рівні:

1. Зовнішній – представлення БД з точки зору користувача.
2. Концептуальний – узагальнене представлення БД, описує які дані зберігаються в БД і зв'язки між ними. Підтримує зовнішні представлення, підтримується внутрішнім рівнем.
3. Внутрішній – фізичне представлення БД в комп'ютері.

Логічна незалежність – повна захищеність зовнішніх моделей від змін, що вносяться в концептуальну модель.

Фізична незалежність – захищеність концептуальної моделі від змін, які вносяться у внутрішню модель.

Подійно-орієнтована архітектура (Event-driven architecture, надалі EDA) – шаблон архітектури програмного забезпечення, який призначений для створення подій, їх виявлення, споживання і реагування на них.

Подія може бути визначена як значна зміна стану. Наприклад, коли споживач купує автомобіль, стан автомобіля змінюється з "на продаж" до "продано". Архітектура системи дилера автомобілів може трактувати цю зміну стану як подію, поява якої може стати відомою іншим програмам даної архітектури.

З формальної точки зору, те, що виробляється, публікується, поширюється, виявляється і споживається (як правило, асинхронно) є повідомленням, яке називають сповіщенням про подію (або нотифікацією), а не самою подією, яка є зміною стану, що викликає появу повідомлення.

Події не подорожують, вони просто відбуваються. Проте термін подія часто використовується метонімічно для позначення самого нотифікаційного повідомлення, що може призвести до певної плутанини.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

Цей архітектурний шаблон може застосовуватися при проектуванні і реалізації ПЗ і систем, які передають події між слабкозв'язаними компонентами програмного забезпечення і сервісами (службами).

Подійно-орієнтована система як правило складається з емітерів подій (або агентів) і споживачів подій (або стоків).

Стоки несуть відповідальність за здійснення реагування на появу події. Реакція не завжди може бути повністю забезпечена самим стоком. Наприклад, стік, може бути відповідальним лише за фільтрацію, трансформацію і відправку події до іншого компонента або він може забезпечити повністю самостійну реакцію на таку подію. Перша категорія стоків може бути заснована на традиційних компонентах, таких як проміжне програмне забезпечення, орієнтоване на обробку повідомлень (message oriented middleware, MOM), в той час, як друга категорія стоків (самостійна реакція в режимі он-лайн) може вимагати більш придатної платформи (фреймворку) для виконання транзакцій.

Розробка ПЗ і систем в подійно-орієнтованій архітектурі дозволяє їм бути сконструйованими способом, який більш відповідає вимогам до їх створення, оскільки такі системи в більшій мірі пристосовуються до непередбачуваних і асинхронних середовищ.

Подійно-орієнтована архітектура (EDA) може доповнювати сервісно-орієнтовану архітектуру (SOA), оскільки сервіси (служби) можуть бути активовані тригерами, які ініціюються при настанні подій.

Ця парадигма особливо корисна, коли стік не забезпечує власного виконання будь-яких дій.

Подійно-орієнтована SOA (SOA-2) розвиває архітектури SOA і EDA для забезпечення більш глибокого і надійного рівня сервісів за рахунок використання раніше невідомих причинно-наслідкових зв'язків, щоб сформувати новий шаблон подій. Цей новий шаблон бізнес-аналітики дає поштовх до небаченого раніше зростання рівня автоматизації підприємства за рахунок привнесення додаткової цінної інформації в описану раніше модель діяльності.

Обчислювальна техніка та сенсорні пристрої (сенсори, датчики, контролери) можуть виявляти зміни стану об'єктів або умов і створювати події, які потім можуть бути оброблені сервісом (службою) або системою.

Подія може складатися з двох частин: заголовка події та тіла події. Заголовок події може включати в себе інформацію таку як, наприклад, назва події, часова мітка події і тип події. Тіло події – це частина, яка описує факт, що стався в дійсності. Тіло події не слід плутати з шаблоном або логікою, яка може бути застосована як реакція на саму подію.

Архітектура, керована подіями, складається з чотирьох логічних рівнів (шарів). Вона починається з виявлення факту, його технічного подання у формі події і закінчується непустою множиною реакцій на цю подію.

Генератор подій. Першим логічним шаром є генератор подій, який виявляє факт і представляє цей факт подією. Оскільки фактом може бути практично все, що може бути сприйнято, то ним може бути і генератор подій. Наприклад, генератором може бути клієнт електронної пошти, система електронної комерції або певний тип датчика.

Перетворення різних даних, отриманих від датчиків, в єдину стандартизовану форму даних, які можуть бути оцінені, є основною проблемою при розробці та реалізації цього шару. Однак, враховуючи, що подія є строго декларативною, можна легко застосовувати будь-які операції трансформації, тим самим усуваючи необхідність забезпечення високого рівня стандартизації.

Канал подій. Канал подій – це механізм, через який інформація від генератора подій передається до обробника подій (event engine) або стоку.

Це може бути з'єднання TCP/IP або вхідний файл будь-якого типу (простий текст, формат XML, e-mail тощо). В один і той же час може бути відкрито кілька каналів подій. Як правило, оскільки обробник подій повинен працювати в режимі, наближеному до реального часу, канали подій зчитуються асинхронно. Події зберігаються в черзі, очікуючи наступної обробки механізмом обробки подій.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

Механізм обробки подій. Механізм обробки подій (event processing engine) є місцем, де подія ідентифікується і вибирається відповідна реакція на нього, яка потім виконується. Це також може призвести до породження ряду тверджень. Якщо подія, яка надійшла до механізму обробки подій, є наприклад такою «Запаси продукту ID досягли нижнього допустимого рівня», це може ініціювати, наприклад, такі реакції як «Замовити продукт ID» і «Сповістити персонал».

Наступна подійно-орієнтована дія (післядія).

Щодо того, як можуть проявлятися наслідки події, слід відмітити, що вони можуть проявитись багатьма різними способами і у різноманітних формах (наприклад, повідомлення електронної пошти, надіслане комусь, або ПЗ, що виводить деяке попередження на екран). Залежно від рівня автоматизації, який забезпечується стоком (механізмом обробки подій), ці дії можуть виявитись зайвими.

Є три основні стилі обробки подій: простий, потоковий і складний. Часто ці три стилі використовуються спільно у розвинутій подійно-орієнтованій архітектурі.

Проста обробка подій.

Проста обробка подій стосується подій, які безпосередньо належать до специфічних вимірних змін умов. У випадку простої обробки подій, мають справу з появою відомих подій, що ініціюють післядію (післядії). Проста обробка подій зазвичай використовується для управління потоком робіт в реальному часі, скорочуючи тим самим час затримки і вартість робіт.

Наприклад, прості події можуть створюватись (породжуватись) датчиком, що виявляє зміну тиску в шині або температуру навколишнього середовища.

Обробка потоку подій.

При обробці потоку подій (event stream processing, далі ESP) відбуваються як звичайні, так і відомі події. Звичайні події (заявки, передачі RFID) перевіряються на те, чи є вони відомими, і передаються інформаційним передплатникам. Обробка потоку подій зазвичай використовується для

управління потоком інформації в реальному часі і на рівні підприємства, що дозволяє своєчасно приймати рішення.

Обробка складних подій.

Обробка складних подій (Complex event processing (CEP)) дозволяє за шаблонами простих і звичайних подій проводити аналіз того, чи напустила складна подія. Обробка складних подій полягає в оцінюванні взаємного впливу подій і в наступному виконанні дій. При цьому, типи подій (відомих або звичайних) можуть перетинатись, а події можуть виникати протягом тривалого періоду часу.

Кореляція подій може бути причинною, тимчасовою або просторовою. CEP вимагає використання складних інтерпретаторів подій, визначення і підбору шаблонів подій, а також відповідних кореляційних методів. Обробка складних подій зазвичай використовується для виявлення і реагування на аномальну поведінку, загрози і можливості у бізнесі.

Розглянемо формат що використовується – **JSON** (JavaScript Object Notation, укр. запис об'єктів JavaScript, вимовляється джейсон) – це текстовий формат обміну даними між комп'ютерами.

JSON базується на тексті, може бути прочитаним людиною. Формат дозволяє описувати об'єкти та інші структури даних. Цей формат головним чином використовується для передачі структурованої інформації через мережу (завдяки процесу, що називають серіалізацією). Розробив і популяризував формат Дуглас Крокфорд.

JSON знайшов своє головне призначення у написанні веб-програм, а саме при використанні технології AJAX. JSON, що використовується в AJAX, виступає як заміна XML (використовується в AJAX) під час асинхронної передачі структурованої інформації між клієнтом та сервером. При цьому перевагою JSON перед XML є те, що він дозволяє складні структури в атрибутах, займає менше місця і прямо інтерпретується за допомогою JavaScript в об'єкти.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64

JSON з'явився через необхідність обміну даними з сервером у реальному часі без використання плагінів для браузерів, flash-додатків або Java-апплетів, які використовувались скрізь на початку 2000-х років. Дуглас Крокфорд був тим, хто активно просував новий на той час формат. Він з колегами хотів створити технологію, яка використовувала б можливості звичайного браузера давала б веб-розробникам можливість створювати веб-додатки із постійним двостороннім зв'язком із веб-сервером.

JSON вперше був використаний в проєкті в Communities.com для Cartoon Network, він дозволяв обмінюватись повідомленнями і одночасно маніпулювати DHTML-елементами.

Веб-сайт JSON.org було запущено 2002 року. У грудні 2005-го року Yahoo! почав переводити деякі зі своїх веб-сервісів на роботу з JSON. Google взялася до роботи з технологією у своєму веб-протоколі GData у грудні 2006-го року.

Використання. За рахунок своєї лаконічності в порівнянні з XML, формат JSON може бути більш придатним для серіалізації складних структур.

Якщо говорити про веб-застосунки, в такому ключі він доречний в задачах обміну даними як між браузером і сервером (AJAX), так і між самими серверами (програмні HTTP-інтерфейси).

Формат JSON так само добре підходить для зберігання складних динамічних структур в реляційних базах даних або файловому кеші.

Синтаксис. JSON будується на двох структурах:

1. Набір пар назва/значення. У різних мовах це реалізовано як об'єкт, запис, структура, словник, хеш-таблиця, список з ключем або асоціативним масивом.

2. Впорядкований список значень. У багатьох мовах це реалізовано як масив, вектор, список, або послідовність.

Це універсальні структури даних. Теоретично всі сучасні мови програмування підтримують їх у тій чи іншій формі. Оскільки JSON

використовується для обміну даними між різними мовами програмування, то є сенс будувати його на цих структурах.

У JSON використовуються такі їхні форми:

1. Об'єкт – це послідовність пар назва/значення. Об'єкт починається з символу { і закінчується символом } . Кожне значення слідує за : і пари назва/значення відділяються комами.

2. Масив – це послідовність значень. Масив починається символом [і закінчується символом]. Значення відділяються комами.

3. Значення може бути рядком в подвійних лапках, або числом, або логічними true чи false, або null, або об'єктом, або масивом. Ці структури можуть бути вкладені одна в одну.

4. Рядок – це послідовність з нуля або більше символів юнікода, обмежена подвійними лапками, з використанням escape-послідовностей, що починаються зі зворотної косої риски \. Символи представляються простим рядком. Тип Рядок (String) дуже схожий на String в мовах C і Java. Число теж дуже схоже на C- або Java-число, за винятком того, що вісімкові та шістнадцяткові формати не використовуються. Пропуски можуть бути вставлені між будь-якими двома лексемами.

Використання JSON в AJAX. Наступний фрагмент коду JavaScript показує, як клієнт може використати XMLHttpRequest для запиту об'єктів в форматі JSON з серверу. (Серверна частина коду пропущена, вона просто повертає на запит URL рядок у JSON форматі).

Треба відзначити, що тут використання XMLHttpRequest не є кросс-браузерним (за деталями звертайтеся на сторінку XMLHttpRequest), і код зазнаватиме незначних модифікацій в різних версіях оглядачів Internet Explorer, Opera, Safari або Mozilla. Використання запиту XMLHttpRequest обмежено правилом одного джерела (same origin policy): URL, що відповідає на запит, має посилатися на той же сайт, що обслуговує сторінку, що ініціювала запит.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

Оглядачі можуть також використовувати тег `<iframe>` для асинхронного запиту JSON-даних в кросс-браузерному варіанті, або використати просте перенаправлення `<form action="url_to_cgi_script" target="name_of_hidden_iframe">`. Такий підхід був поширений до приходу популярного нині запиту XMLHttpRequest.

Динамічний тег `<script>` також можна використати для підвантаження JSON-даних. Ця техніка можлива, щоб обійти надто суворе правило одного джерела, але вона не є безпечною. Запит XMLHttpRequest пропонується, як безпечніша альтернатива.

Питання безпеки. Хоча JSON призначений для передачі даних в серіалізованому вигляді, його синтаксис відповідає синтаксису JavaScript і це створює низку проблем безпеки. Часто для обробки даних, отриманих від зовнішнього джерела у форматі JSON, до них застосовується функція `eval()` без якої-небудь попередньої перевірки.

JavaScript `eval()`. Оскільки JSON представляється синтаксично правильним фрагментом коду JavaScript, природним способом розбору JSON-даних в JavaScript-програмі є використання вбудованої в JavaScript функції `eval()`, яка призначена для обчислення JavaScript-виразів. При цьому підході відпадає необхідність у використанні додаткових парсерів.

Техніка використання `eval()` робить систему вразливою, якщо джерело JSON-даних, що використовуються, не відноситься до надійних. Такими даними може виступати шкідливий JavaScript-код для атак за допомогою ін'єкції коду. За допомогою цієї вразливості можливо здійснювати крадіжку даних, підробку автентифікації. Проте, вразливість можна усунути за рахунок використання додаткових засобів перевірки даних на коректність. Наприклад, до виконання `eval()` отримані від зовнішнього джерела дані можуть перевірятися за допомогою регулярних виразів.

Вбудований JSON. Останні версії веб-браузерів мають вбудовану підтримку JSON і здатні його обробляти без виклику функції `eval()`, що

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

призводить до описаної проблеми. Обробка JSON у такому разі зазвичай здійснюється швидше.

Принаймні дві популярні бібліотеки JavaScript використовують вбудований JSON у разі його доступності: jQuery; Dojo.

Підробка крос-доменного запиту. Непродумане використання JSON робить сайти вразливими до підробки міжсайтових запитів (CSRF або XSRF). Оскільки тег <script> допускає використання джерела, що не належить до того ж домену, що і використовуваний ресурс, це дозволяє виконувати код даних, представлених у форматі JSON, в контексті довільної сторінки, що робить можливою компрометацію паролів або іншої конфіденційної інформації користувачів, що пройшли авторизацію на іншому сайті.

Це є проблемою тільки у разі вмісту в JSON-даних конфіденційної інформації, яка може бути компрометована третьою стороною і якщо сервер розраховує на політику одного джерела, блокуючи доступ до даних при виявленні зовнішнього запиту. Це не є проблемою, якщо сервер визначає допустимість запиту, надаючи дані тільки у разі його коректності. HTTP cookie не можна використовувати для визначення цього. Виключне використання HTTP cookie використовується підробкою міжсайтових запитів.

Розширення, JSONP. JSONP або «JSON з підкладкою» є розширенням JSON, коли назва функції зворотного виклику вказується як вхідний аргумент. Спочатку ідея була запропонована в блозі MacPython в 2005 році, і в наш час використовується багатьма Web 2.0 застосунками, такими, як Dojo Toolkit Applications, Google Toolkit Applications і zanox Web Services.

Подальші розширення цього протоколу були запропоновані з урахуванням введення додаткових аргументів, як, наприклад, у разі JSONPP за підтримки S3DB вебсервісів.

Оскільки JSONP використовує скрипт-теги, виклики по суті відкриті світові. З цієї причини JSONP може бути недоречними для зберігання конфіденційних даних.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		68

4.2 Захист розробленого програмного забезпечення

Захист розробленого програмного забезпечення буде відбуватися за допомогою CRYPTON – алгоритм симетричного блочного шифрування (розмір блоку 128 біт, ключ довжиною до 256 біт), розроблений південнокорейським криптологом Чьо Лім Хун з південнокорейської компанії Future Systems, яка з кінця 1980-х років працює на ринку забезпечення мереж і захисту інформації. Алгоритм був розроблений в 1998 році в якості шифру – учасника конкурсу AES. Як зізнавався автор, конструкція алгоритму спирається на алгоритм SQUARE[1]. В алгоритмі Crypton немає традиційних для блочних шифрів мережі Фейстеля. Основу даного шифру становить так звана SP-мережа (повторювана циклова функція, що складається із замін-перестановок, орієнтована на розпаралелену нелінійну обробку всього блоку даних). Крім високої швидкості, перевагами таких алгоритмів є полегшення дослідження стійкості шифру до методів диференціального та лінійного криптоаналізу, що є на сьогодні основними інструментами розтину блочних шифрів. На конкурс AES була представлена версія алгоритму Crypton v0.5. Однак, як казав Чьо Лім Хун, йому не вистачало часу для розробки повної версії. І вже на першому етапі конкурсу AES в ході аналізу алгоритмів, версія Crypton v0.5 була замінена на версію Crypton v1.0. Відмінність нової версії від первинної полягала в зміні таблиці замін та в модифікації процесу розширення ключа.

Як і інші учасники конкурсу AES, Crypton призначений для шифрування 128-бітових блоків даних[2]. При шифруванні використовуються ключі шифрування для декількох фіксованих розмірів – від 0 до 256 біт з кратністю 8 бітів. Структура алгоритму Crypton – структура «Квадрата» – багато в чому схожа на структуру алгоритму Square, створеного в 1997 році. Криптографічні перетворення для алгоритмів з даною структурою можуть бути виконані як для цілих рядків і стовпців масиву, так і над окремими його байтами. (Варто зазначити, що алгоритм Square був розроблений авторами майбутнього

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		69

переможця конкурсу AES – авторами алгоритму Rijndael – Вінсентом Ріджменом і Джоан Дейменом.)

Шифрування

Алгоритм Crypton являє 128-бітовий блок шифруємих даних у вигляді байтового масиву 4×4 , над якими в процесі шифрування проводиться кілька раундів перетворень. У кожному раунді передбачається послідовне виконання наступних операцій:

- Таблична заміна γ ;
- Лінійне перетворення π ;
- Байтова перестановка τ ;
- Операція σ .

Таблична заміна γ

Алгоритм Crypton використовує 4 таблиці замін. Кожна з яких заміщає 8-бітне вхідне значення на вихідне такого ж розміру.

Лінійне перетворення π

Тут використовується 4 спеціальні константи. Ці константи об'єднані в маскуючі послідовності

Байтова перестановка τ

Дана перестановка перетворює найпростішим чином рядок даних у стовпець.

Операція σ

Дана операція є побітовим складанням всього масиву даних з ключем раунду. Зауважимо, саме 12 раундів шифрування рекомендується автором алгоритму Чьо Хун Лімом, проте сувора кількість раундів не встановлена.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено розроблене у бакалаврської дипломної роботі система моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи: Функціональних кнопок ПЗ; Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші; Верхнього меню; Розділу обрання групи; Розділу виведення результату роботи системи.

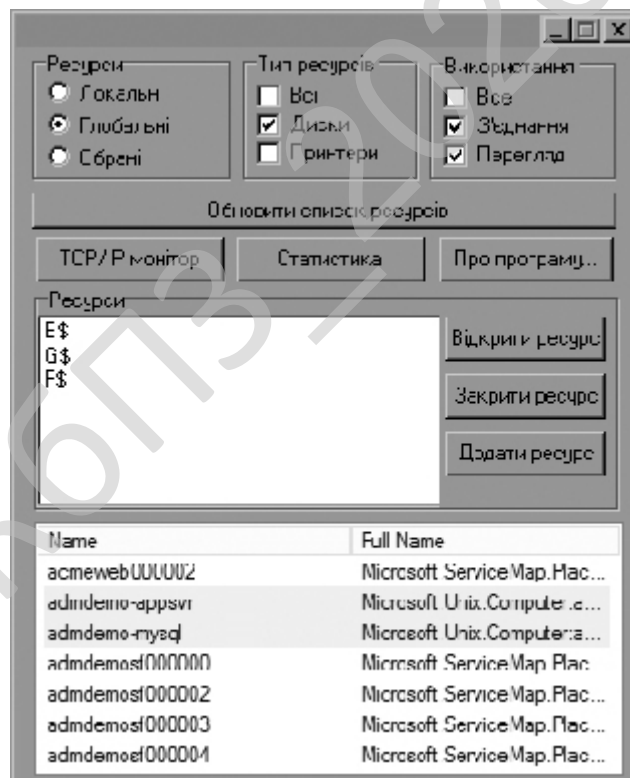


Рисунок 5.1 – Головне вікно ПЗ

Розроблена програма має дуже простий і зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним

середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий. Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

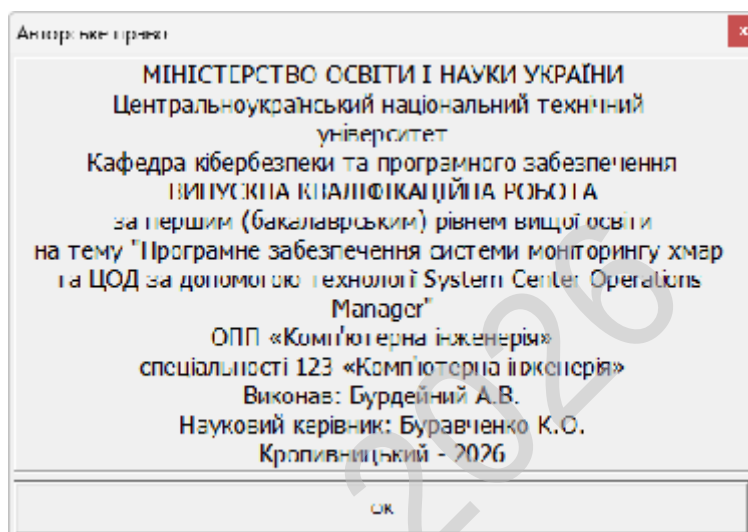


Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку виробника так і з боку споживача. Оскільки кожна програмна система є унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

- Випуск.
- Встановлення та активація.
- Деактивація.
- Адаптація.
- Оновлення.
- Вмонтування.
- Відстежування версій.
- Видалення.
- Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати ІТ рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене ІТ рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження;

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються в ІТ рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		73

невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

- відповідності поставленим вимогам;
- правильна відповідь для усіх можливих вхідних даних;
- виконання функцій за прийнятний час;
- практичність;
- сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом білої скриньки засноване на аналізі керуючої структури програми. Програма вважається повністю перевіреною, якщо проведено вичерпне тестування маршрутів (шляхів) її графа управління.

У цьому випадку формуються тестові варіанти, в яких:

- Гарантується перевірка всіх незалежних маршрутів програми.
- Знаходяться гілки True, False для всіх логічних рішень.
- Виконуються всі цикли (у межах їхніх кордонів та діапазонів).
- Аналізується правильність внутрішніх структур даних.

Недоліки тестування "білої скриньки":

- Кількість незалежних маршрутів може бути дуже велика.

– Повне тестування маршрутів не гарантує відповідності програми вихідним вимогам до неї.

– У програмі можуть бути пропущені деякі маршрути.

– Не можна виявити помилки, поява яких залежить від даних.

Переваги тестування "білої скриньки" пов'язані з тим, що принцип «білої скриньки» дозволяє врахувати особливості програмних помилок:

– Кількість помилок мінімально в «центрі» і максимально на «периферії» програми.

– Попередні припущення про ймовірність потоку керування або даних у програмі часто бувають некоректними. У результаті типовим може стати маршрут, модель обчислень за яким опрацьована слабо.

– При записі алгоритму програмного забезпечення у вигляді тексту на мові програмування можливе внесення типових помилок трансляції (синтаксичних та семантичних).

– Деякі результати в програмі залежать не від вихідних даних, а від внутрішніх станів програми.

Проводилось тестування чорної скриньки. Основне місце програми тестів «чорної скриньки» — інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

– Як виконуються функції програми.

– Як приймаються вихідні дані.

– Як виробляються результати.

– Як зберігається цілісність зовнішньої інформації.

При тестуванні «чорної скриньки» розглядаються системні характеристики програм, ігнорується їхня внутрішня логічна структура. Вичерпне тестування, як правило, неможливе.

Наприклад, якщо в програмі 10 вхідних величин і кожна приймає по 10 значень, то кількість тестових варіантів становитиме 10^{10} . Тестування «чорної скриньки» не реагує на багато особливостей програмних помилок.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, які забезпечують повну перевірку всіх функціональних вимог до програми.

Програмний виріб тут розглядається як «чорна скринька», чію поведінку можна визначити тільки дослідженням його входів та відповідних виходів. При такому підході бажано мати:

- Набір, утворений такими вхідними даними, які призводять до аномалій у поведінці програми (назвемо його ІТс).

- Набір, утворений такими вхідними даними, які демонструють дефекти програми (назвемо його ОТ).

Будь-який спосіб тестування «чорної скриньки» повинен:

- Виявити такі вхідні дані, які з високою ймовірністю належать набору ІТс.
- Сформулювати такі очікувані результати, які з високою ймовірністю є елементами набору ОТ.

Принцип «чорної скриньки» не альтернативний принципу «білої скриньки». Скоріше це доповнює підхід, який виявляє інший клас помилок.

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

- Некоректних чи відсутніх функцій.
- Помилки інтерфейсу.
- Помилки у зовнішніх структурах даних або в доступі до зовнішньої бази даних.
- Помилки характеристик (необхідна ємність пам'яті і т.д.).
- Помилки ініціалізації та завершення.

Обрано умови розповсюдження – Freeware. Це власницьке програмне забезпечення, котре можна Безоплатно використовувати протягом необмеженого терміну без обмежень у функціональності, і поширюване без сирцевих кодів.

Автори такого програмного забезпечення, як правило, хочуть «дати щось спільноті», але хочуть також контролювати його подальшу розробку. Іноді, коли програмісти вирішують припинити розробку, вони передають сирцевий код іншим програмістам, або ж спільноті як вільне програмне забезпечення.

Дуже часто плутають поняття «безплатне програмне забезпечення» та «вільне програмне забезпечення», хоча вони суттєво відрізняються.

Безплатне програмне забезпечення можна безоплатно встановлювати та використовувати (іноді з певними обмеженнями, як, наприклад, «безплатне для домашнього або некомерційного вжитку»), в той час як вільне програмне забезпечення можна продавати за будь-яку суму, але при тому, у користувача, котрий його отримує, повинні бути права на вивчення, модифікацію та поширення сирцевих кодів одержаної програми.

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

Рішення завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.
- Досліджена система моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.
- На основі отриманих результатів досліджень створена програмна реалізація системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані призначені для

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

системи моніторингу хмар та ЦОД за допомогою технології System Center Operations Manager. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм CRYPTON.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		79

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ramon Nastase «Computer Networking: The Beginner's guide for Mastering Computer Networking, the Internet and the OSI Model». 2018. – 186 p.
2. Russ White & Ethan Banks «Computer Networking Problems and Solutions: An Innovative Approach to Building Resilient, Modern Networks». 2017. – 832 p.
3. Вінтенко Б., Смірнов О., Миронець І., Смірнова Т., Смірнов С. «Імітаційна модель шляхів вхідних даних комп'ютерної інтелектуальної системи підтримки оператора енергоблоку АЕС». *Комбінаторні конфігурації та їхні застосування: Матеріали XXVII Міжнародного науково-практичного семінару, присвяченого 125-річчю Національного університету «Запорізька політехніка»* (Запоріжжя-Кропивницький-Київ, 4-6 червня 2025 р.). Запоріжжя: НУ «Запорізька політехніка», 2025. С.82-91.
4. Al-Azzeh, J., Ayyoub, B., Mesleh, A., Smirnova, T., Gnatyuk, S., Drieiev, O., Smirnov, O., Dorenskyi, O. «Cloud-Based Information System for Evaluating Caverns in the Process of Blasting Metal Surfaces of Details». *International Review on Modelling and Simulations* 18 (1), 2025. pp. 32-42.
5. Смірнова Т.В., Коноплицька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка*. 2024. №4(24), С. 6-27.
6. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем ІР-телефонії». *Підводні технології*, 2024, № 13, с. 28-35.
7. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		80

8. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.

9. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchев, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings*, Volume 3530, 2023, pp. 256-265.

10. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». *Lecture Notes on Data Engineering and Communications Technologies*, 2023, 178, pp. 208–223.

11. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». *Сучасні інформаційні системи*, 2023, том 7, № 2, С. 49-56.

12. Smirnova, T., Gnatyuk, S., Yudin, O., Sydorenko, V., Polozhentsev, A., «The Model for Calculating the Quantitative Criteria for Assessing the Security Level of Information and Telecommunication Systems». *CEUR Workshop Proceedings* Volume 3156, 2022, Pages 390-399.

13. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління*, № 2(70). 2022. С. 28-37.

14. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 3(69). С. 93-98.

15. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»*, № 2 (307). С. 46-52. 2022.

16. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 1(67). С. 84-89.

17. Smirnova T., Gnatyuk S., Berdibayev R., Avkurova Zh., Iavich M. «Cloud-Based Cyber Incidents Response System and Software Tools». *Communications in Computer and Information Science*, 2021, vol 1486. Springer, Cham. pp 169-184.

18. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings*. Volume 2740, 2020, Pages 102-114.

19. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.

20. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

21. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) *Data-Centric Business and*

Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

22. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 125-136.

23. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 366-379.

24. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», *CEUR Workshop Proceedings* Volume 2608, 2020, Pages 633-645.

25. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». *International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019*; Odessa; Ukraine; 9-13 September 2019. P.22-28.

26. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». *International Journal of Computing*; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

27. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

28. Smirnov, O., Odarchenko, R., Abakumova, A., Usik, P., Kundyz, M., «QoE optimization technique for media delivery in 5G networks». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, Kyiv, Ukraine, 8 – 11 October 2019. P.597-601.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		83

29. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». *CEUR Workshop Proceedings*, Vol 2588, P. 90-106, 2019.

30. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», *2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019*, P. 395-399.

31. Smirnov, O., Kuznetsov, A., Kiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», *2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS)*, Kyiv, Ukraine April 17-19, 2019 P. 353-358.

32. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», *2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS)*, Kyiv, Ukraine April 17-19, 2019 P. 347-352.

33. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», *CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019*, Pages 618-629.

34. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», *Telecommunications and Radio Engineering*. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.

35. Смірнов О.А., Смірнова Т.В., Буравченко К.О., Кравченко С.С., Горбов В.О., «Хмарна система підтримки прийняття рішень технологічного процесу відновлення поверхонь конструкцій і деталей машин». *Сучасні інформаційні системи*. 2021. Т. 5, № 4. С. 79-95

36. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у

комп'ютерних системах операторів стільникового зв'язку» *Вісник Черкаського державного технологічного університету. Технічні науки.* №4. С. 103-110. 2020.

37. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», *Кібербезпека: освіта, наука, техніка.* № 3(7). С. 43-62. 2020.

38. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В., Поліщук Л.І. Інформаційна безпека в комп'ютерних мережах. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2020. – 294 с.

39. О.А. Смірнов, П.С. Усік, «Дослідження перспектив використання технологічних рішень в мережах 5G» у *Кібербезпека та інформаційні технології: монографія.* – Х. : ТОВ «ДІСА ПЛЮС», 2020.С. 122-135.

40. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». *Центральноукраїнський науковий вісник. Технічні науки.* № 2(33). с. 161-172, 2019.

41. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В. Поліщук Л.І. Проектування комп'ютерних систем та мереж. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2019. – 264 с.

42. Smirnov, O., Kuznetsov, A., Kuznetsova., K. Synthesis of Discrete Signals with Improved Correlation Properties. Монографія: In.: ISCI'2019: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2019, pp. 281-299. – ISBN: 978-0-9989826-8-7 (Hardback), ISBN: 978-0-9989826-9-4 (Ebook).

43. Смірнов О.А., Дреєва Г.М. Метод генерування фрактального трафіку за допомогою моделі генератора на графі. Монографія: Інформаційна безпека та інформаційні технології : монографія / за заг. ред. В. С. Пономаренка. – Х. : Вид. Рожко С.Г. 2019. С. 123-139

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		85

44. Дреєва Г.М., Смірнов О.А., Дреєв О.М. Метод генерування фрактальноподібної числової послідовності на основі скінченного автомату для моделювання трафіку у мережі. Центральнуукраїнський науковий вісник. Технічні науки. № 1(32). с. 173-183, 2019.

45. Смірнова Т.В., Солових Є.К., Смірнов О.А., Дреєв О.М. Побудова хмарних інформаційних технологій оптимізації технологічного процесу відновлення та зміцнення поверхонь деталей. Центральнуукраїнський науковий вісник. Технічні науки. № 1(32). с. 184-194, 2019.

46. Смірнов О.А., Смірнов С.А., Поліщук Л.І., Смірнова Т.В., Коноплицька-Слободенюк О.К. Метод формування антивірусного захисту даних з використанням безпечної маршрутизації метаданих. Кібербезпека: освіта, наука, техніка. – Том 3 № 3. – Київ: КУ ім. Бориса Грінченка. – 2019. – С. 63-87.

47. Смірнов О.А., Гнатюк С.О., Кавун С.В., Терейковський І.А., Жмурко Т.О., Смірнов С.А., Коваленко А.С. Основи безпеки в комп'ютерних мережах. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2018. – 177 с.

48. Смірнов О.А., Котелянець В.В. Стійкі до колізій стохастичні моделі функціонування безпроводових сенсорних мереж. Вісник інженерної академії України, №3, с. 145-152, 2018

49. Смірнов О.А., Смірнов С.А., Дідик А.К., Дреєв А.М. Алгоритми формування безлічі маршрутів передачі метаданих у антивірусні хмарні системи. Збірник наукових праць "Системи обробки інформації". - Випуск 5 (142). - Х.: ХУПС - 2016. - С. 148-152.

50. Смірнов О.А., Смірнов С.А. Дідик А.К., Дреєв О.М. Моделі системи нейромережових експертів безпечної маршрутизації у хмарних антивірусних системах. Збірник наукових праць "Системи обробки інформації". - Випуск 3 (140). - Х.: ХУПС - 2016. - С. 36-39.

51. Смірнов О.А., Смірнов С.А., Дідик А.К., Дреєв А.М. Спосіб контролю ліній зв'язку телекомунікаційної системи антивірусу. Збірник наукових

праць Харківського університету Повітряних Сил. Випуск 2 (47). – Харків: ХУПС. - 2016. - С. 121-127.

52. Смірнов О.А., Смірнов С.А., Дідик А.К. Метод безпечної маршрутизації метаданих у хмарні антивірусні системи. Системи озброєння та військова техніка. - Випуск 2 (46) - Х.: ХУПС - 2016. - С. 146-149.

53. Смірнов О.А., Кавун С.В., Доренський О.П., Вялкова В.І. Інформаційна безпека в комп'ютерних мережах. Навчальний посібник – Кіровоград: РВЛ КНТУ, 2016. – 151 с.

54. Смірнов О.А., Кавун С.В., Коваленко О.В., Дреєв О.М. Мережні інформаційні технології. Навчальний посібник – Кіровоград: РВЛ КНТУ, 2016. – 159 с.

55. Смірнов О.А., Кавун С.В., Коваленко О.В., Доренський О.П., Дреєв О.М., Вялкова В.І. Комп'ютерні мережі. Навчальний посібник – Кіровоград: РВЛ КНТУ, 2016. – 233 с.

					ВКРБ-123.26.0034.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		87