

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**«ОРГАНІЗАЦІЯ ТА РОЗГОРТАННЯ СИСТЕМИ РЕЗЕРВНОГО КОПЮВАННЯ
З ВИКОРИСТАННЯМ BORG»**

Виконав: здобувач 4 курсу

Рівень бакалавр

Галузь знань 12 «Інформаційні технології»,
спеціальність 125 «Кібербезпека»

Хорус Олександр Сергійович

Керівник: к.т.н., доцент

Бойко Віктор Дмитрович

Рецензент: к.ф.-м.н. доцент

Чепурна Олена Євгеніївна

Одеса - 2025

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»

Факультет кібербезпеки та інформаційних технологій

Кафедра **кібербезпеки**

Галузь знань: **12 Інформаційні технології**

Спеціальність: **125 Кібербезпека**

Назва освітньої програми: **Кібербезпека**

ЗАТВЕРДЖУЮ

Завідувач кафедри кібербезпеки

професор С.А. Горбаченко

_____ «____» _____ 20__ року

З А В Д А Н Н Я

**на кваліфікаційну (бакалаврську) роботу
здобувачу Хорусу Олександр Сергійовичу**

1. Тема роботи: «Організація та розгортання системи резервного копіювання з використанням Borg»

Керівник роботи: к.т.н., доцент Бойко Віктор Дмитрович

затверджені наказом НУ ОЮА від «18» березня 2025 року № 548-6

2. Строк подання здобувачем роботи «19» травня 2025 року

3. Дата видачі завдання «16» вересня 2024 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка

Здобувач _____

(підпис)

(прізвище та ініціали)

Керівник роботи _____

(підпис)

(прізвище та ініціали)

АНОТАЦІЯ

Кваліфікаційна робота на тему «Організація та розгортання системи резервного копіювання з використанням Borg», для здобуття першого (бакалаврського) рівня вищої освіти за спеціальністю 125 Кібербезпека. Робота виконана на 48 сторінках загального тексту та 42 сторінках основного тексту.

У сучасних інформаційних системах надійне резервне копіювання є критично важливим для забезпечення збереження даних та відновлення після можливих збоїв або атак. У цій кваліфікаційній роботі розглядається процес організації та розгортання системи резервного копіювання з використанням Borg. Borg - ефективний інструмент для дедуплікації, стиснення та шифрування резервних копій.

У роботі аналізуються основні принципи резервного копіювання, переваги та недоліки існуючих рішень, а також особливості використання Borg для автоматизації процесів створення та управління резервними копіями. Описано методи налаштування Borg для різних сценаріїв використання.

Практична частина містить розробку та тестування системи резервного копіювання на основі Borg, що включає налаштування сховищ, політики резервування та механізми відновлення даних.

Результати дослідження підтверджують доцільність використання Borg як сучасного, безпечного та зручного інструменту резервного копіювання, що дозволяє мінімізувати ризики втрати даних і забезпечити безперервність роботи інформаційних систем.

РЕЗЕРВНЕ КОПІЮВАННЯ, BORG, ДЕДУПЛІКАЦІЯ, ШИФРУВАННЯ, ІНФОРМАЦІЙНА БЕЗПЕКА.

ABSTRACT

Qualification thesis on the topic "organizing and deploying a backup system using Borg" for obtaining the first (bachelor's) level of higher education in the specialty 125 Cybersecurity. The work was completed on 48 pages of the general text and 42 pages of the main text.

In modern information systems, reliable backup is critically important for ensuring data preservation and recovery after potential failures or attacks. This qualification thesis examines the process of organizing and deploying a backup system using Borg. Borg is an efficient tool for deduplication, compression, and encryption of backups.

The thesis analyzes the fundamental principles of backup, the advantages and disadvantages of existing solutions, and the specifics of using Borg for automating backup creation and management processes. Methods for configuring Borg for various usage scenarios are described.

The practical part includes the development and testing of a backup system based on Borg, covering storage configuration, backup policies, and data recovery mechanisms.

The research results confirm the feasibility of using Borg as a modern, secure, and convenient backup tool that minimizes the risks of data loss and ensures the continuity of information systems.

BACKUP, BORG, DEDUPLICATION, ENCRYPTION, INFORMATION SECURITY.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	6
ВСТУП	7
1 ТЕОРЕТИЧНІ АСПЕКТИ РЕЗЕРВНОГО КОПІЮВАННЯ	9
1.1 Основи резервного копіювання	10
1.2 Основні методи резервного копіювання.....	14
1.3 Проблеми та виклики у сфері резервного копіювання	17
1.4 Аналіз існуючих методів та засобів резервного копіювання	19
1.5 Borg Backup як сучасний інструмент резервного копіювання	21
2 АНАЛІЗ СИСТЕМИ РЕЗЕРВНОГО КОПІЮВАННЯ Borg Backup	24
2.1 Загальна характеристика Borg Backup	24
2.2 Архітектура та принцип роботи Borg Backup	26
2.3 Ключові можливості та переваги Borg Backup	28
2.4 Обмеження та недоліки	30
2.5 Теоретичні основи впровадження Borg Backup для різних сценаріїв використання	33
3 РОЗГОРТАННЯ ТА ТЕСТУВАННЯ СИСТЕМИ РЕЗЕРВНОГО КОПІЮВАННЯ З ВИКОРИСТАННЯМ BORG	36
3.1 Розгортання системи.....	36
3.2 Тестування системи	43
3.3 Практичні рекомендації щодо впровадження системи	44
ВИСНОВКИ.....	46
ПЕРЕЛІК ПОСИЛАНЬ	47

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД – база даних

Система – Система резервного копіювання

Копіювання – Резервне копіювання

Borg – Borg Backup

ВСТУП

У сучасному цифровому світі інформація є одним із найцінніших ресурсів. Втрата важливих даних через апаратні збої, кібератаки, людський фактор або природні катастрофи може призвести до серйозних фінансових і репутаційних втрат для організацій та окремих користувачів. Саме тому питання розробки та впровадження ефективних систем резервного копіювання є актуальним і важливим для забезпечення безперебійного функціонування інформаційних систем.

Копіювання є основним способом захисту даних від втрати, дозволяючи зберігати їх у безпечному місці, та відновлювати у разі потреби. Однак традиційні методи резервного копіювання можуть бути застарілими та неефективними через високі вимоги до дискового простору, тривалість процесу копіювання та складність управління резервними копіями. У зв'язку з цим постає необхідність використання сучасних інструментів, які оптимізують процес копіювання, підвищують його продуктивність та забезпечують високий рівень безпеки.

Одним із таких рішень є Borg, який забезпечує гнучкість, ефективність і надійність процесу резервного копіювання. Borg дозволяє забезпечити резервне копіювання з використанням сучасних методів дедуплікації, інкрементного збереження даних і шифрування, що робить його особливо актуальним для організацій, які потребують високого рівня захисту і збереження великих обсягів інформації. Система Borg Backup (або просто Borg) є інструментом для створення резервних копій даних з особливим акцентом на ефективність, безпеку та економію місця для зберігання. Borg підтримує інкрементне резервне копіювання, дедуплікацію та шифрування, що робить її потужним рішенням для управління резервними копіями великих обсягів даних [1].

У даній дипломній роботі розглядаються теоретичні та практичні аспекти впровадження системи резервного копіювання з використанням Borg. Аналізуються існуючі методи та інструменти, розробляється та тестується модель

резервного копіювання, яка дозволяє забезпечити надійне збереження даних і їх оперативне відновлення.

Метою роботи є розробка та впровадження системи автоматизованого резервного копіювання, яка забезпечить високий рівень надійності та безпеки збережених даних.

Для досягнення цієї мети поставлено наступні завдання:

1. Аналіз існуючих методів та засобів резервного копіювання, визначення їх переваг та недоліків, вибір системи резервного копіювання серед актуальних.
2. Дослідження можливостей обраної системи, та оцінка її ефективності у порівнянні з іншими рішеннями.
3. Розробка рекомендацій щодо впровадження обраної нами системи резервного копіювання для різних сценаріїв використання.
4. Розгортання та тестування запропонованої системи та оцінка її продуктивності, надійності та безпеки.

Об'єктом дослідження є системи резервного копіювання в інформаційних технологіях.

Предметом дослідження є процес організації та розгортання системи з використанням Borg Backup.

У роботі застосовуються методи аналізу та синтезу, порівняльного аналізу програмних рішень, експериментального моделювання та тестування систем резервного копіювання.

Результати дослідження можуть бути корисними для організацій, які прагнуть впровадити ефективні механізми копіювання для забезпечення безпеки та цілісності своїх даних.

1 ТЕОРЕТИЧНІ АСПЕКТИ РЕЗЕРВНОГО КОПІЮВАННЯ

Резервне копіювання (Backup) – це процес створення та збереження копій даних для їх відновлення у випадку втрати, пошкодження або знищення оригіналів. Воно є невід'ємною частиною інформаційної безпеки та забезпечує безперервність роботи інформаційних систем.

Основні цілі резервного копіювання: захист від втрати даних унаслідок збоїв устаткування. Відновлення після програмних помилок або вірусних атак. Архівування даних для довготривалого збереження. Захист від людського фактору (ненавмисне видалення або зміна файлів).

Існують різні методи резервного копіювання, кожен із яких має свої переваги та недоліки.

Основні методи резервного копіювання – повне резервне копіювання, додаткове резервне копіювання, диференційне резервне копіювання, змішане резервне копіювання, дзеркальне копіювання.

Повне резервне копіювання (Full Backup) - Передбачає створення копії всіх даних у системі, де перевагами є простота відновлення, незалежність від попередніх копій, недоліками – великий обсяг збережених даних, значний час виконання.

Додаткове (інкрементне) резервне копіювання (Incremental Backup) це вид копіювання де зберігаються тільки ті файли, які були змінені з моменту останнього резервного копіювання. Переваги – менший обсяг даних, швидке створення копій навідріз від повного резервного копіювання. Недоліки - відновлення потребує всіх попередніх копій.

Диференційне резервне копіювання (Differential Backup) – копіює всі файли, що змінилися з моменту останнього повного резервного копіювання. Переваги - швидше відновлення порівняно з інкрементним способом. Недоліки: збільшений обсяг копій у порівнянні з інкрементним методом.

Змішане (комбіноване) резервне копіювання - використання різних методів у поєднанні для досягнення найкращої ефективності.

Дзеркальне копіювання (Mirror Backup) – Створюється точна копія поточного стану даних без збереження попередніх версій. Недоліки: не дозволяє повернутися до попередніх версій файлів [2].

Сучасні організації стикаються з низкою викликів у сфері копіювання, серед яких: великий обсяг даних де постійне зростання обсягів інформації ускладнює процес зберігання, через, що потрібно контролювати оптимізацію, безпеку даних та швидкість відновлення. Для подолання цих викликів використовуються новітні технології, зокрема дедуплікація, шифрування, хмарні сховища та автоматизація процесів.

Borg Backup – це програмне рішення для резервного копіювання, що поєднує ефективність, безпеку та зручність у використанні.

Основні можливості Borg Backup: дедуплікація - уникнення дублювання даних, що дозволяє значно зменшити використання дискового простору. Стиснення - зменшення розміру резервних копій без втрати якості. Шифрування - гарантує захист даних від несанкціонованого доступу. Інкрементне резервне копіювання - швидке створення копій без дублювання незмінних файлів. Гнучке управління сховищем - підтримка локальних і віддалених сховищ [3].

Порівняно з іншими інструментами (наприклад, rsync, Duplicity, Bacula) [10-12], Borg Backup забезпечує кращу ефективність завдяки дедуплікації на рівні блоків, що дозволяє значно зменшити використання сховищ та пришвидшити процеси резервного копіювання.

1.1 Основи резервного копіювання

Резервне копіювання, або бекап (від англ. backup) – процес створення копії даних з носія (жорсткого диска, флешки тощо). Його метою є відновлення даних у разі їх ушкодження або видалення [4].

Основне призначення створення резервної копії - забезпечення можливості відновлення інформації, якщо оригінал буде втрачено. Втратою вважається будь-яка подія, що призводить до зміни або видалення даних з носія, внаслідок чого

вони втрачають цінність. Приклад: навмисне видалення важливих даних підприємства з метою завдання шкоди.

Об'єкти резервного копіювання – це дані чи набори даних, з яких можна створити резервну копію. Приклади об'єктів: файли та каталоги, дані прикладних програм, дані операційної системи чи сама ОС (наприклад, Windows System State або AIX System Backup), образи віртуальних машин та дисків віртуальних машин, файлові системи та інші.

Вимоги для відновлення даних. Щоб визначити вимоги до швидкості відновлення та періодичності створення резервних копій, використовують наступні поняття: Recovery Point Objective (RPO) – цільова точка відновлення. Визначає періодичність створення резервних копій. У разі пошкодження або видалення даних, відновлення до «найсвіжішої» копії можливе тільки до останнього RPO. Залежно від технологій, використаних при створенні резервних копій, можливе відновлення між різними RPO, така технологія називається Point In Time Restore, що властива СУБД. Recovery Time Objective (RTO) - цільовий час відновлення. Визначає час, потрібний для відновлення даних з резервної копії. Час від втрати даних до початку роботи фахівця з відновлення, а також час від відновлення з резервної копії до початку використання даних користувачами, може не включатися до RTO.

У разі катастрофи (відмова сайту або подвійна відмова зарезервованих компонентів) існують додаткові визначення: Disaster Recovery Point Objective (DRPO) - цільова точка відновлення під час катастрофи. RPO у випадку катастрофи [5].

Рівні резервного копіювання: Повне резервне копіювання (Full Backup або L0) - повна копія даних. Забезпечує створення повної копії об'єкта резервного копіювання. Цей рівень гарантує максимальну відповідність оригіналу даних копії. Диференційне резервне копіювання (Differential Backup або L1) - копіювання змін, що були внесені після останньої повної копії. Створення такої копії потребує більше часу та обсягу, ніж додаткове копіювання, але прискорює процес відновлення. Загалом, альтернатива між повною та додатковою копією.

Додаткове резервне копіювання (Incremental Backup або L2) - копіювання змін, що відбулися з моменту останнього повного, диференційного або додаткового копіювання. На додаткове копіювання зазвичай витрачається менше часу, оскільки копіюється менше файлів. Проте, процес відновлення даних займає більше часу, оскільки спочатку потрібно відновити дані останньої повної копії, а потім – всі резервні копії, від яких залежить додаткова копія. Lx (розшифровується як Level X) – стандарт опису методів резервного копіювання. Оскільки різні джерела можуть по-різному трактувати диференційне та додаткове копіювання, може використовуватись нотація Lx, що для перелічених типів буде означати L1 та L2 відповідно. Рівень Lx завжди залежить від попереднього Lx-1 або Lx-n, якщо він був останнім. Наприклад, у разі послідовності L0, L5, L3, L2, L4, процедура відновлення на останню найсвіжішу резервну копію буде виглядати так: L0, L2, L4.

Цілісність резервних копій даних полягає у відповідності даних копії та оригіналу на момент її створення. Неправильний вибір методу може зробити відновлення неможливим. Щоб забезпечити цілісність, потрібно: Забезпечити незмінність даних під час копіювання, інакше неможливо перевірити відповідність копії: контрольні суми копії та оригіналу будуть відрізнятися. Розблокувати дані для читання - ОС може блокувати спроби читання файлів. Записати зміни з оперативної пам'яті на жорсткий диск (у Linux - команда sync) - інакше деякі частини об'єкта копіювання можуть не відповідати змінам, що вже були внесені.

Приклади: при створенні резервної копії файлів Windows, деякі системні файли, що використовуються в системі, недоступні. Але завдяки технології Volume Shadow Copy Services, перед початком копіювання створюється тіньова копія тому, що дозволяє використовувати її замість оригінального тому. У випадку копіювання образу віртуальної машини VMware vSphere, для забезпечення цілісності використовується VMware Tools, що сповіщає операційну систему про підготовку даних до резервного копіювання. З боку ОС (Windows), це також активує компоненти VSS.

Методи передачі резервних копій: з використанням агента, off-host, serverless.

З використанням агента. Копіювання відбувається за допомогою програмного забезпечення, встановленого на захищеній системі. Цей метод потребує контролю за встановленням та оновленням, але дозволяє інтегруватися в програмне середовище. Для інших методів також може використовуватися агент для координації процесів всередині ОС.

Off-host. Метод, що дозволяє створювати резервну копію без значного навантаження на клієнтську систему. Дані зазвичай передаються зі сховища даних безпосередньо на сервер резервного копіювання.

Serverless. Метод, що передає дані від клієнта у сховище без участі сервера резервного копіювання. Сервер отримує інформацію про об'єкти, що були скопійовані. Використовується для зменшення навантаження на сервер.

Методи створення копії даних: тіньова копія, snapshot дисків, snapshot дискового розділу.

Тіньова копія (Shadow Copy) – створення миттєвої копії з дискового розділу ОС, з використанням властивостей файлової системи.

Snapshot дисків – фіксація змін на основному диску та перенаправлення їх в окремий файл при створенні резервної копії.

Snapshot дискового розділу – створення миттєвої копії даних безпосередньо на дисковому масиві, з використанням можливостей дискових масивів.

Зберігання резервної копії. Резервні копії зберігаються на різних носіях залежно від потреб. Зазвичай використовується сервер резервного копіювання, що координує процеси та керує сховищами. Сховищами можуть бути: Стрічкова бібліотека або стример, дискове сховище, Virtual Tape Library, «хмарний» бекап.

Стрічкова бібліотека або стример – запис копій на магнітну стрічку. Переваги: швидкість запису та читання, відносно низька вартість зберігання, можливість зберігання off-line. Недоліки: потреба у штучній надмірності, повільний доступ до файлів, обмежена компресія, обмежена кількість одночасних операцій.

Дискове сховище – запис на диски, які можуть бути об'єднані в RAID або на дискову систему зберігання. Переваги: швидкий доступ, необмежена кількість одночасних операцій, можливість використання усунення дуплікації. Недоліки: менша швидкість запису та читання порівняно зі стрічками, відносно вища вартість.

Virtual Tape Library – запис на диски, але сховище виглядає для клієнта як стрічкова бібліотека. Переваги: можливість використання усунення дуплікації, велика кількість одночасних операцій, висока швидкість доступу.

«Хмарний» бекап – запис даних за «хмарною» технологією через онлайн-сервіси провайдерів. Сервер резервного копіювання також може використовувати хмару. Переваги: низька вартість, швидке розгортання та масштабування, Pay-as-you-Go. Недоліки: низька швидкість запису та читання, недостатня безпека [6].

1.2 Основні методи резервного копіювання

Резервне копіювання (англ. backup) - це процес створення копій даних з метою їх відновлення у випадку втрати, пошкодження або знищення оригіналів. Враховуючи постійне збільшення обсягу інформації та критичну важливість її збереження, резервне копіювання стало одним з ключових аспектів політики безпеки будь-якої сучасної інформаційної системи.

Існує декілька основних методів резервного копіювання, кожен з яких застосовується відповідно до конкретних потреб організації, технічних можливостей та вимог до часу відновлення (RTO – Recovery Time Objective) та втрати даних (RPO – Recovery Point Objective).

Повне резервне копіювання (Full Backup). Повне резервне копіювання передбачає створення повної копії усіх файлів та даних, які підлягають зберігання. Це найпростіший для реалізації метод, але водночас і найбільш ресурсомісткий. Принцип роботи: усі дані з обраного джерела (диск, база даних, сервер) копіюються в окрему резервну область або на інший носій інформації (наприклад, зовнішній диск, NAS-сервер, хмарне сховище та ін.).

Переваги: повна незалежність від інших резервних копій. Простота відновлення: для відновлення даних потрібна лише одна копія. Висока надійність збереження даних.

Недоліки: тривалий час копіювання великих обсягів інформації. Велике споживання дискового простору. Часте повне копіювання може створювати додаткове навантаження на мережу та сервери.

Сценарії використання: періодичне повне резервне копіювання (наприклад, щотижневе або щомісячне) часто використовується в поєднанні з іншими методами для оптимізації ресурсів.

Інкрементне резервне копіювання (Incremental Backup). Інкрементне резервне копіювання дозволяє зменшити обсяг копій та прискорити процес, оскільки зберігаються лише файли, які були змінені або створені після останнього резервного копіювання - незалежно від його типу (повного або інкрементного).

Принцип роботи: після створення повної копії всі наступні копії зберігають лише зміни з попереднього моменту. Це дозволяє економити місце, але вимагає послідовності при відновленні.

Переваги – значне зменшення розміру резервних копій. Швидке виконання операції збереження. Мінімальне навантаження на систему.

Недоліки – складний процес відновлення: необхідно мати всі попередні копії (повну + усі інкрементні). Вища ймовірність пошкодження резервного ланцюга.

Сценарії використання: ідеально підходить для щоденного або навіть щогодинного збереження змін у великих системах з динамічним контентом (наприклад, бази даних, поштові сервери тощо).

Диференціальне резервне копіювання (Differential Backup). Диференціальне резервне копіювання є проміжним варіантом між повним та інкрементним. Воно дозволяє зберігати всі зміни, які відбулися з моменту останнього повного резервного копіювання.

Принцип роботи: кожна диференціальна копія містить усі файли, змінені після повного резервного копіювання, незалежно від кількості попередніх диференціальних копій.

Переваги: швидше відновлення, ніж при інкрементному копіюванні (потрібні лише дві копії: повна і остання диференціальна). Простіший процес відновлення.

Недоліки: обсяг копії зростає з кожним днем до наступного повного резервного копіювання. Триваліший час створення копії в порівнянні з інкрементним методом.

Сценарії використання: широко використовується в системах, де важливий баланс між швидкістю відновлення та ефективністю збереження простору.

Дзеркальне копіювання (Mirror Backup). Дзеркальне резервне копіювання - це процес постійного або періодичного синхронізування оригінальних даних з копією, створюючи повне дзеркало (ідентичну копію) даних у реальному часі або з певним інтервалом.

Принцип роботи: будь-яка зміна в основному джерелі автоматично дублюється в резервному сховищі. Найчастіше використовуються спеціалізоване програмне забезпечення або RAID-технології.

Переваги: мінімальний час відновлення. Дані завжди актуальні.

Недоліки: відсутність історії змін – при видаленні чи пошкодженні даних вони також автоматично видаляються/псуваються в копії. Потребує захисту від збоїв синхронізації. Часто вимагає додаткового захисту (наприклад, збереження версій).

Сценарії використання: підходить для систем, де критично важливо мати постійно доступні актуальні дані (наприклад, у фінансових установах або дата-центрах).

Кожен з методів резервного копіювання має свої особливості, і вибір залежить від багатьох факторів: обсягу даних, частоти їх зміни, доступного місця для зберігання, бюджету на інфраструктуру та вимог до швидкості відновлення. Найчастіше на практиці використовуються гібридні стратегії, що поєднують

повне, інкрементне та диференціальне копіювання, дозволяючи досягти максимальної ефективності.

Наприклад, типовою практикою є створення повного бекапу раз на тиждень, інкрементного – щодня та дзеркального копіювання - у режимі реального часу для найкритичніших даних. Такі комбінації дозволяють оптимізувати ресурси, мінімізувати ризики втрати інформації та забезпечити стабільну роботу ІТ-систем [7].

1.3 Проблеми та виклики у сфері резервного копіювання **Ошибка! Закладка не определена.**

Основні проблеми безпеки резервних копій: Ransomware: Сучасні атаки шифрують не тільки робочі дані, а й резервні копії, якщо ті зберігаються на підключених носіях. Наприклад, вірус може знайти і зашифрувати backup на мережевому диску. Ненадійне зберігання: резервні копії іноді зберігають на незахищених FTP-серверах або дисках без шифрування. Відсутність багаторівневого контролю доступу (ACL): всі працівники можуть мати доступ до критичних резервних копій - це ризик.

Потенційні рішення: використання Air-Gapped backup (фізично від'єднаний носій). Впровадження Zero Trust моделі доступу. Автоматичне шифрування копій (наприклад, AES-256). Механізми Immutable backup - резервні копії, які неможливо змінити чи видалити певний період.

Основні проблеми надійності і цілісності копій: "фантомні" копії: бекап створюється, але містить пошкоджені або неповні файли. Несумісність між версіями програм: наприклад, копія з SQL Server 2012 не завжди сумісна з SQL Server 2022. Застарілі носії: старі стрічкові системи часто мають проблеми з читанням.

Потенційні рішення: регулярне тестування відновлення (Disaster Recovery Drill). Автоматичні перевірки цілісності (checksum, hash-валидація). Ведення журналу перевірок і успішного відновлення.

Основні проблеми обсягу даних і масштабованості: великі обсяги даних ускладнюють резервування, особливо для організацій з великими відео-графічними файлами або базами даних. Повне копіювання займає багато часу та ресурсів. Недостатнє місце на сховищах - доводиться або видаляти старі копії, або купувати додаткові сервери.

Потенційні рішення: впровадження інкрементальних або диференційних копій (копіюються лише зміни). Дедуплікація даних – усунення дублювання. Перехід на масштабовані хмарні сховища (Amazon S3, Azure Blob).

Основні проблеми швидкості процесів копіювання та відновлення: мале backup-вікно: часто організація не може дозволити зупинку сервісів на ніч. Повільне відновлення може паралізувати бізнес (наприклад, RTO = 8 годин, коли критичний максимум - 1 год). Пропускна здатність мережі: при передачі великих обсягів до хмари - вузьке місце.

Потенційні рішення: оптимізація за допомогою snapshot-ів (миттєві знімки). Розгортання локального кешу відновлення. Балансування мережевого трафіку, використання QoS (Quality of Service).

Основні проблеми інтеграції з хмарними сервісами: складність управління гібридним середовищем: частина даних у хмарі, частина - локально. Різні API та формати даних: не всі хмарні рішення однаково підтримуються. Нестабільне інтернет-з'єднання: може зірвати резервне копіювання.

Потенційні рішення: використання BaaS (Backup as a Service) - резервне копіювання як сервіс. Автоматизація процесів через інтеграції з API хмарних платформ. Встановлення гарантованого SLA з провайдером хмарних сервісів.

Основні проблеми юридичних та регуляторних вимог: невиконання GDPR або локальних законів про зберігання персональних даних. Зберігання даних у недозволених юрисдикціях (наприклад, сервери у країнах, які не відповідають вимогам законодавства). Архівація даних не відповідає періоду зберігання за законом (наприклад, потрібно 7 років, а видаляються через 3).

Потенційні рішення: ведення ретеншн-політик (Retention Policies) – визначення термінів зберігання. Вибір сертифікованих сховищ, які відповідають стандартам (ISO, SOC2). Юридичний аудит політик резервного копіювання.

Основні проблеми людського фактору: помилки у налаштуваннях: неправильно вказано розклад, джерела, або тип копії. Відсутність відповідального персоналу: backup «висить» на IT-відділі без конкретної відповідальності. Відсутність моніторингу: проблеми виявляються лише під час спроби відновлення.

Потенційні рішення: призначення відповідального за резервне копіювання (Backup Owner). Використання моніторингових систем з повідомленнями (email, Slack, SMS). Навчання персоналу - регулярні тренінги з disaster recovery [8].

1.4 Аналіз існуючих методів та засобів резервного копіювання

У сучасних умовах захист інформації стає пріоритетним завданням для будь-якої інформаційної системи. Одним з ключових методів забезпечення безпеки даних є резервне копіювання - процес створення копій даних з метою їхнього відновлення у випадку втрати або пошкодження. У цьому розділі буде проведено аналіз трьох популярних засобів резервного копіювання: Bacula, BorgBackup (Borg) та rsync.

Bacula - це комплексне, мережеве рішення для автоматизації резервного копіювання, відновлення та архівації даних. Воно підтримує роботу у великих середовищах з численними клієнтами та серверами.

Основні характеристики: архітектура клієнт-сервер - складається з декількох компонентів (директор, демон зберігання, демон файлів, консоль). Підтримка різних типів носіїв (диски, стрічки). Гнучке планування завдань. Підтримка баз даних (MySQL, PostgreSQL) для зберігання метаданих. Можливість інкрементного, диференціального та повного копіювання.

Переваги: підходить для великих інфраструктур, централізоване керування, високий рівень автоматизації та масштабованості, докладне логування та звітність.

Недоліки: висока складність конфігурації, високі вимоги до адміністрування, необхідність глибоких знань про архітектуру системи [10].

BorgBackup, або Borg, – це утиліта командного рядка для резервного копіювання, яка використовує дедуплікацію, стиснення та шифрування.

Основні характеристики: надійна дедуплікація на рівні блоків, підтримка стиснення (lz4, zlib, zstd), вбудоване шифрування (AES-256), можливість монтування резервної копії через FUSE, проста реалізація в однокористувацькому середовищі.

Переваги: висока ефективність зберігання за рахунок дедуплікації, простота у розгортанні на одному вузлі, безпека завдяки шифруванню, підходить для резервного копіювання локальних та віддалених даних (через SSH).

Недоліки: обмежена підтримка мережових конфігурацій (у порівнянні з Bacula), відсутність централізованого управління, не підтримує стрічкові пристрої.

Rsync – утиліта синхронізації файлів і каталогів, що часто використовується для створення резервних копій за допомогою скриптів або cron-завдань.

Основні характеристики: синхронізація локальних або віддалених даних через SSH, передача тільки змінених частин файлів, висока швидкість передачі, підтримка фільтрів та включення/виключення файлів.

Переваги: простота використання, легка інтеграція у скрипти, мінімальні вимоги до ресурсів, ідеально підходить для регулярного копіювання на локальні або мережеві накопичувачі.

Недоліки: відсутність вбудованого стиснення та шифрування (потрібна додаткова конфігурація), немає механізму управління версіями, відсутність журналювання операцій резервного копіювання [11].

Нижче наведена порівняльна таблиця (таблиця 1.1) цих засобів, для більш зручного сприйняття.

Таблиця 1.1 – Порівняння сучасних методів та засобів резервного копіювання.

Параметр	Bacula	BorgBackup	rsync
Архітектура	Клієнт-сервер	Одновузлова	Одновузлова
Дедуплікація	Х	✓	Х
Шифрування	Обмежене	✓ (AES-256)	Х (зовнішнє)
Стиснення	Х	✓	Х
Централізоване керування	✓	Х	Х
Простота налаштування	Х	✓	✓
Підтримка стрічок	✓	Х	Х
Відновлення окремих файлів	✓	✓	✓
Підтримка інкрементів	✓	✓	Частково (через опції)

1.5 Borg Backup як сучасний інструмент резервного копіювання

Що таке Borg Backup? Borg Backup (Borg) – це потужний CLI-інструмент для резервного копіювання, розроблений з урахуванням сучасних вимог до збереження даних. Він забезпечує високу ефективність, безпеку та гнучкість, підходить для локального та віддаленого бекапу і ідеально пристосований до роботи в умовах обмежених ресурсів.

Ключові особливості Borg Backup: Дедуплікація, стиснення, шифрування, інкрементність і швидкодія, гнучкість сховища, підтримка віддалених репозиторіїв, перевірка цілісності,

Borg виконує дедуплікацію на рівні блоків - це означає, що під час створення резервної копії повторювані частини файлів зберігаються лише один раз. Дедуплікація дозволяє суттєво зменшити об'єм резервного сховища, особливо при частих резервних копіях одного й того самого каталогу або системи. Зміни в одному файлі не призводять до повторного збереження всього файлу - лише змінені частини.

Borg підтримує кілька алгоритмів стиснення, що дозволяє зменшити розмір архівів без втрати продуктивності. Користувач може обирати між швидкими алгоритмами (наприклад, lz4) або більш ефективними, але повільнішими (наприклад, zlib, zstd). Стиснення є гнучким параметром - його можна застосовувати вибірково.

Borg забезпечує клієнтське шифрування - дані шифруються до того, як вони покидають комп'ютер користувача. Використовується сильне симетричне шифрування (AES-256), а також механізми автентифікації, що гарантують цілісність даних. Навіть у випадку доступу до сховища зломисником, без ключа розшифрування архіву залишаться захищеними.

Borg створює повну резервну копію лише один раз, а всі наступні копії є інкрементними. Завдяки дедуплікації та оптимізації процесів, нові бекапи займають набагато менше часу та місця. Зміни фіксуються швидко і з мінімальним навантаженням на систему.

Borg використовує власний формат архівів, який можна монтувати як файлову систему. Це дозволяє зручно переглядати та вибірково витягати файли з резервної копії без необхідності повного відновлення.

Borg чудово працює з віддаленими серверами через SSH. Він не потребує спеціального серверного програмного забезпечення - лише доступ до звичайного облікового запису на сервері. Це робить Borg зручним вибором для резервного копіювання у хмару, на інші сервери або навіть Raspberry Pi.

Borg має вбудовані інструменти перевірки, які дозволяють впевнитися, що резервні копії не пошкоджені. Регулярна перевірка допомагає завчасно виявити потенційні проблеми із зберіганням.

Чому Borg вважається сучасним рішенням? Він ефективно використовує ресурси, уникаючи дублювання даних. Забезпечує високий рівень безпеки без необхідності складної серверної інфраструктури. Підтримує автоматизацію та сценарії, що важливо для DevOps-підходу. Його можна інтегрувати у різноманітні середовища: від особистого ПК до складних серверних систем [9].

2 АНАЛІЗ СИСТЕМИ РЕЗЕРВНОГО КОПІЮВАННЯ Borg Backup

Borg Backup (BorgBackup) – це ефективна система резервного копіювання, орієнтована на швидке, надійне і стиснуте зберігання даних. Вона призначена для створення резервних копій даних на серверах або в розподілених мережах.

Основні характеристики та принципи роботи: дедуплікація – Borg забезпечує дедуплікацію даних, що дозволяє зберігати тільки унікальні блоки даних, зменшуючи обсяг збережених даних і, відповідно, економить місце на носії. Шифрування: Borg підтримує шифрування даних за допомогою AES-256, що дозволяє зберігати резервні копії у захищеному вигляді, гарантуючи конфіденційність інформації. Стиснення: Програмне забезпечення використовує ефективні алгоритми стиснення (наприклад, LZ4), що дозволяє зменшити розмір резервних копій без втрати продуктивності. Швидкість та ефективність: Завдяки використанню алгоритмів дедуплікації та стиснення, Borg може створювати резервні копії великих обсягів даних за мінімальний час. Інкрементальні резервні копії: Borg створює лише інкрементальні копії після першого повного бекапу, що дозволяє зменшити витрати на зберігання. Мережеве зберігання: Borg підтримує резервне копіювання як на локальних пристроях, так і на віддалених серверах, що дозволяє зберігати копії на будь-якому сервері, доступному по SSH. Відновлення: відновлення даних можливе як частково, так і повністю, за допомогою простого інтерфейсу командного рядка.

В цілому, Borg Backup є надійним і ефективним рішенням для резервного копіювання великих обсягів даних, особливо для серверів і розподілених систем.

2.1 Загальна характеристика Borg Backup

Borg Backup – це потужна система для створення резервних копій даних з відкритим кодом. Вона відрізняється високою ефективністю, зручністю використання та надійністю. Borg Backup була розроблена для збереження та відновлення великих обсягів даних у різноманітних умовах, таких як сервери,

робочі станції або навіть персональні комп'ютери. Вона підтримує різноманітні механізми стиснення та шифрування, що робить її ідеальним вибором для забезпечення безпеки і конфіденційності даних.

Основні характеристики Borg Backup: інкрементальні резервні копії: Borg Backup використовує інкрементальний підхід до створення копій, що дозволяє зберігати лише зміни з моменту останнього бекапу, значно зменшуючи обсяг використовуваного простору для збереження даних.

Стиснення: програма підтримує використання алгоритмів стиснення, таких як zlib, LZ4 або Zstandard, що дозволяє економити місце на носіях для збереження даних.

Шифрування: Borg Backup підтримує надійне шифрування даних за допомогою алгоритмів AES-256, що гарантує захист інформації від несанкціонованого доступу.

Простота використання: для користувача Borg Backup має зручний інтерфейс командного рядка, який дозволяє легко налаштовувати і запускати процеси резервного копіювання, відновлення даних, а також моніторити стан резервних копій.

Масштабованість та підтримка віддалених сховищ: Borg Backup дозволяє працювати з віддаленими серверами через SSH або з іншими віддаленими сховищами, що забезпечує можливість централізованого зберігання та доступу до резервних копій.

Ефективність: завдяки використанню механізмів deduplication (пошук і видалення дубльованих даних) Borg Backup мінімізує зайве використання дискового простору, що є важливим фактором при роботі з великими обсягами даних.

Можливість створення архівів з різними параметрами: програма дозволяє створювати резервні копії за різними політиками, зокрема з різними періодичностями, що дає змогу користувачам обирати оптимальні налаштування для їх конкретних потреб.

Підтримка автоматизації та скриптів: Borg Backup можна інтегрувати з різними системами автоматизації, такими як cron, що дозволяє створювати регулярні резервні копії без участі користувача.

Переваги Borg Backup: висока ефективність використання ресурсів і економія місця завдяки інкрементальним бекапам і deduplication. Підтримка шифрування і стиснення для забезпечення конфіденційності та збереження місця на диску. Легкість у використанні завдяки зручному інтерфейсу командного рядка. Надійна підтримка віддалених сховищ для зберігання даних. Підтримка широкого спектру операційних систем, таких як Linux, macOS, та Windows.

Недоліки: вимагає певних технічних знань для налаштування, зокрема при роботі з віддаленими серверами. Може бути не такою зручною для користувачів, які віддають перевагу графічним інтерфейсам.

Borg Backup є відмінним вибором для тих, хто шукає ефективну, надійну та безпечну систему для створення резервних копій даних, зокрема в середовищах, де потрібна висока економія місця і захист інформації [3].

2.2 Архітектура та принцип роботи Borg Backup

Основні компоненти Borg Backup: клієнт (Borg Client), сховище (Repository), резервні копії (Backups).

Клієнт (Borg Client) – це інтерфейс командного рядка (CLI), який відповідає за виконання команд створення, відновлення та управління резервними копіями. Клієнт може працювати як локально, так і віддалено, за допомогою SSH для з'єднання з віддаленими серверами.

Сховище (Repository) – це місце, де зберігаються резервні копії. Може бути локальним або віддаленим (через SSH чи інші протоколи). Репозиторій складається з кількох файлів і каталогів, що зберігають дані в спеціалізованому форматі.

Резервні копії (Backups) - резервні копії складаються з блоків даних. Borg виконує дедуплікацію, тому тільки унікальні блоки даних зберігаються в

репозиторії. Резервні копії можуть містити метадані, таке як інформація про версії файлів та каталоги.

У принцип роботи Borg Backup входить: дедуплікація, шифрування, стиснення, інкрементальне резервне копіювання, робота з репозиторіями, автоматизація та скрипти.

Дедуплікація - Borg використовує дедуплікацію на рівні блоків, що означає, що однакові блоки даних, які зустрічаються в різних версіях резервних копій, зберігаються лише один раз. Це дозволяє значно зменшити обсяг зайнятого простору.

Шифрування – Borg підтримує шифрування резервних копій з використанням AES-256. Ключі шифрування можуть бути визначені користувачем або створюватися автоматично.

Стиснення – Borg використовує ефективне стиснення даних для зменшення розміру резервних копій. Різні алгоритми стиснення можуть бути використані (наприклад, LZ4, Zstandard).

Інкрементальне резервне копіювання: Borg підтримує інкрементальні резервні копії, що означає, що після першої повної резервної копії наступні будуть містити тільки змінені або нові файли, що значно скорочує обсяг даних для збереження.

Робота з репозиторіями: користувачі можуть створювати, видаляти, переглядати та відновлювати резервні копії з репозиторіїв. Репозиторії можуть бути локальними або зберігатися на віддалених серверах, підключених через SSH.

Автоматизація та скрипти: Borg Backup часто використовується у поєднанні з системами автоматизації для регулярного створення резервних копій за розкладом.

Ключові переваги

1. Ефективність: знижує необхідний обсяг зберігання завдяки дедуплікації та стисненню.
2. Безпека: підтримка шифрування забезпечує захист даних.
3. Простота використання: завдяки командному рядку та документації,

Borg є зручним інструментом для адміністраторів систем.

4. Підтримка віддаленого зберігання: можна налаштувати резервне копіювання на віддалені сервери через SSH.

2.3 Ключові можливості та переваги Borg Backup

Ключові можливості які має Borg Backup: інкрементальне резервне копіювання, шифрування даних, дедуплікація даних, швидкість резервного копіювання та відновлення, підтримка віддаленого зберігання, підтримка різних файлових систем.

Інкрементальне резервне копіювання. Однією з головних особливостей Borg Backup є інкрементальне резервне копіювання. Після початкового повного резервного копіювання наступні резервні копії містять лише зміни, що відбулися з моменту попередньої копії. Це дозволяє значно зекономити місце для зберігання, оскільки лише нові або змінені файли додаються до резервної копії, тоді як залишок даних не дублюється.

Інкрементальне резервне копіювання має ряд переваг: зменшення обсягу зберігання: оскільки зберігаються лише зміни. Швидкість копіювання: виконання інкрементальних резервних копій займає менше часу, ніж повне резервне копіювання.

Шифрування даних. Borg Backup підтримує вбудоване шифрування даних, що забезпечує їх захист від несанкціонованого доступу. Використовуючи сучасні криптографічні алгоритми, Borg гарантує, що резервні копії будуть надійно захищені як при зберіганні, так і при передачі через мережу. Переваги шифрування: безпека даних: збереження конфіденційності резервних копій.

Простота налаштування: шифрування в Borg легко налаштовується через параметри командного рядка.

Захист при передачі: шифрування гарантує безпеку даних навіть при їх передачі по незахищених каналах, наприклад, через інтернет.

Дедуплікація даних. Однією з унікальних особливостей Borg Backup є

дедуплікація - це процес уникання дублювання однакових блоків даних. Дедуплікація дозволяє зберігати тільки один екземпляр ідентичних блоків даних, навіть якщо вони зустрічаються в кількох резервних копіях. Це значно зменшує розмір резервних копій. Переваги дедуплікації: економія місця на диску, особливо корисно при резервному копіюванні великих обсягів схожих або однакових даних. Зменшення часу на резервне копіювання: оскільки зберігаються тільки нові або змінені дані.

Швидкість резервного копіювання та відновлення. Borg використовує ефективні алгоритми стиснення та дедуплікації, що дозволяє значно прискорити процес резервного копіювання, а також відновлення даних. Швидкість є важливою характеристикою для будь-якої системи резервного копіювання, оскільки з часом резервні копії можуть ставати дуже великими. Переваги швидкості: швидке створення резервних копій: для забезпечення безперервної роботи системи. Швидке відновлення даних: що критично важливо в разі аварії.

Підтримка віддаленого зберігання. Borg Backup дозволяє зберігати резервні копії на віддалених серверах через SSH або інші мережеві протоколи. Це дозволяє забезпечити резервне копіювання в безпечному середовищі, навіть у випадку фізичної втрати локальних носіїв. Переваги віддаленого зберігання: безпека: дані зберігаються в захищеному місці, недоступному для локальних загроз. Масштабованість: можна створювати резервні копії для кількох серверів або баз даних на віддалених хостах.

Підтримка різних файлових систем. Borg Backup працює з різними файловими системами, що дозволяє зберігати резервні копії для різних операційних систем, таких як Linux, macOS та Windows (через WSL).

Простота використання та автоматизація. Borg Backup має інтерфейс командного рядка, що дозволяє створювати резервні копії та налаштовувати автоматизовані процеси за допомогою скриптів. Це робить інструмент гнучким і підходить для використання як в малих, так і в великих інфраструктурах.

Основні переваги використання Borg Backup: економія місця для зберігання, висока надійність, продуктивність, гнучкість у налаштуванні, відкритий код і

безкоштовне використання.

Економія місця для зберігання - завдяки інкрементальним копіям і дедуплікації Borg дозволяє значно зменшити обсяг використовуваного дискового простору. Це дає можливість оптимізувати ресурси для зберігання великих обсягів даних.

Висока надійність – Borg Backup гарантує цілісність та безпеку даних завдяки шифруванню та дедуплікації. Зберігаючи резервні копії на віддалених серверах, можна уникнути фізичних пошкоджень локальних носіїв.

Продуктивність – Borg працює швидко навіть з великими наборами даних завдяки використанню алгоритмів стиснення та дедуплікації, що дозволяє зберігати та відновлювати дані за мінімальний час.

Гнучкість у налаштуванні - Borg Backup є інструментом з відкритим вихідним кодом, що дозволяє адаптувати його до специфічних потреб користувачів, включаючи налаштування для роботи в конкретних мережах або з конкретними типами даних.

Відкритий код і безкоштовне використання - Borg Backup є безкоштовним і має відкритий вихідний код, що дає можливість користуватися ним без ліцензійних обмежень і вносити власні модифікації. Це ідеальне рішення для тих, хто потребує гнучкості та економії коштів на комерційних рішеннях для резервного копіювання.

Borg Backup є потужним інструментом для резервного копіювання даних, який поєднує в собі високу ефективність, безпеку, швидкість та гнучкість. Його можливості для інкрементального резервного копіювання, шифрування, дедуплікації та роботи з віддаленими серверами роблять його ідеальним вибором для захисту даних як в малих, так і в великих системах.

2.4 Обмеження та недоліки

Звичайно, кожна із систем має свої недоліки, не прорахування всіх можливих сценаріїв, обмеження, давайте розглянемо кожен з пунктів детальніше.

Складність налаштування: Borg Backup є інструментом командного рядка (CLI), і його налаштування може бути незручним для користувачів, які не мають досвіду роботи з командним рядком або терміналом. Щоб налаштувати резервне копіювання, потрібно вміти працювати з командними інтерфейсами, розуміти, як використовувати SSH для доступу до віддалених серверів, налаштовувати ключі для аутентифікації та інше. Інструкції для налаштування зазвичай передбачають базові знання в області Linux або інших операційних систем, що може бути проблемою для користувачів, які не працюють з системами на базі Unix.

Потреба в збереженні резервного з'єднання. Використання SSH: Для того, щоб використовувати Borg для резервного копіювання на віддалений сервер, потрібно налаштувати доступ через SSH. Це може включати генерування SSH ключів, налаштування доступу через публічні ключі та відповідне налаштування серверів.

Ризики: Якщо доступ до віддаленого серверу буде перервано, то резервне копіювання не зможе виконуватись. Додатково, якщо не налаштовано автоматичне відновлення з'єднання або перевірка з'єднання, резервне копіювання може не відбутись без явної помилки.

Обмеження на підтримку платформ. Підтримка основних операційних систем: Borg підтримує основні операційні системи, такі як Linux, macOS та Windows. Однак, якщо ви працюєте на менш популярних системах (наприклад, FreeBSD або інші Unix-подібні системи), вам доведеться шукати способи ручної інтеграції Borg, і це може бути не так зручно. Windows: Встановлення Borg на Windows є можливим, але через необхідність використання Cygwin або Windows Subsystem for Linux (WSL), цей процес може бути не таким простим. І хоча WSL і Cygwin дозволяють користуватись Borg, ці платформи мають обмеження у продуктивності та сумісності з певними функціями.

Обмеження в плануванні задач. Відсутність вбудованого планувальника: Borg сам по собі не має вбудованих засобів для автоматизації завдань, таких як регулярне виконання резервних копій. Щоб налаштувати автоматичне резервне копіювання, вам потрібно використовувати інші інструменти, такі як cron на

Linux або Task Scheduler на Windows. Необхідність додаткової конфігурації - це означає, що користувачам потрібно вручну налаштовувати розклад, що додає складність у використання для тих, хто шукає більш інтуїтивно зрозуміле рішення для автоматизації.

Висока вимогливість до ресурсів при великих обсягах даних. Процеси резервного копіювання і дедуплікації: Резервне копіювання великих обсягів даних, особливо з включенням дедуплікації, може вимагати значних обчислювальних ресурсів (пам'яті та процесора). Чим більший обсяг даних, тим більше часу і ресурсів знадобиться для обробки. На слабших системах цей процес може бути повільним і втомлюючим, що особливо помітно при великих або постійно змінюваних даних. Наприклад, якщо у вас великий масив даних, резервне копіювання може зайняти кілька годин або навіть днів.

Відсутність графічного інтерфейсу. Інтерфейс командного рядка: Borg - це інструмент, що працює тільки через командний рядок. Для користувачів, які звикли до графічних інтерфейсів, це може бути великим недоліком. Альтернативи - хоча існують сторонні графічні оболонки для Borg (наприклад, Vorta для Linux/macOS), вони не є частиною основного пакету і можуть не мати повної підтримки всіх функцій Borg. Це може обмежити гнучкість і зручність для користувачів, які шукають більш зручні та прості у використанні інтерфейси.

Складність в управлінні великими сховищами. Коли зберігається багато копій резервних даних (зокрема, через дедуплікацію або при великому обсязі даних), це може стати важким для управління. Наприклад, підтримка різних версій резервних копій, відновлення конкретних файлів або перевірка доступних точок відновлення може бути трудомістким процесом. Масштабованість - якщо вам потрібно зберігати резервні копії на кількох різних серверах або сховищах, може виникнути складність у керуванні і обслуговуванні таких розподілених систем.

Обмежена підтримка для відновлення на рівні файлів. Резервне копіювання і відновлення Borg дозволяє відновлювати файли з резервних копій, але цей процес може бути не таким простим, як у інших системах. Наприклад, вам

доведеться використовувати команду для "монтування" резервної копії, щоб доступити окремі файли. Складність доступу до окремих файлів - замість того, щоб мати інтуїтивно зрозумілий інтерфейс для відновлення окремих файлів, вам доведеться працювати з командним рядком, що може бути незручно, особливо якщо вам потрібно часто відновлювати окремі файли.

Низька швидкість шифрування і розшифрування. Шифрування даних: Borg підтримує шифрування резервних копій, що є важливим для забезпечення безпеки даних. Однак шифрування і розшифрування можуть сповільнити процес резервного копіювання та відновлення, особливо якщо ви працюєте з великими обсягами даних. Вплив на продуктивність - це може бути проблемним для систем з обмеженими ресурсами або для користувачів, які хочуть отримати швидке і ефективне резервне копіювання [9].

Borg Backup є потужним інструментом для резервного копіювання, але має кілька обмежень, особливо для тих, хто шукає простіший або більш інтуїтивно зрозумілий процес. Висока вимогливість до ресурсів, складність налаштування та відсутність графічного інтерфейсу можуть бути значними недоліками для певних користувачів. Тому важливо враховувати ці фактори при виборі Borg для використання у своїй інфраструктурі.

2.5 Теоретичні основи впровадження Borg Backup для різних сценаріїв використання

Система резервного копіювання є невід'ємною частиною будь-якої інформаційної інфраструктури, яка прагне забезпечити надійне збереження даних та їх відновлення у разі втрати, пошкодження або кібератак. Одним із сучасних і ефективних інструментів для реалізації цієї задачі є Borg Backup - система резервного копіювання з відкритим вихідним кодом, яка поєднує високу швидкодію, безпеку та ефективне використання дискового простору.

Borg Backup використовує дедуплікацію на рівні блоків, що дозволяє зменшити обсяг збережених даних, зберігаючи лише унікальні блоки. Крім того,

вона підтримує сильне шифрування, що забезпечує конфіденційність резервних копій, навіть якщо вони зберігаються у потенційно незахищених середовищах (наприклад, у хмарі або на віддалених серверах).

Для впровадження Borg Backup ефективно та раціонально, необхідно враховувати специфіку цільового середовища. Залежно від масштабу системи, обсягу даних, кількості клієнтських пристроїв, а також вимог до доступності й безпеки, слід обирати відповідну архітектуру резервного копіювання. Теоретично можна виокремити кілька типових сценаріїв:

Персональне резервне копіювання - це найпростіший сценарій, де резервні копії створюються для одного користувача або одного комп'ютера. Borg Backup може бути встановлений локально, а репозиторій зберігатися на зовнішньому накопичувачі або у хмарному середовищі через SSH. Основна увага приділяється автоматизації процесу та безпеці збереження ключів шифрування.

Резервне копіювання в межах малої організації. У цьому випадку кілька комп'ютерів або серверів мають централізовано створювати резервні копії у спільне сховище. Borg дозволяє це реалізувати за допомогою віддаленого репозиторію через SSH або за допомогою спеціальних серверів резервного копіювання. Ключовими аспектами є контроль доступу, моніторинг стану копій та ротація резервних архівів.

Віддалене або хмарне резервне копіювання. Зростання популярності хмарних сервісів спричинило необхідність реалізації резервного копіювання у віддалені центри обробки даних. Borg може працювати у поєднанні з інструментами тунелювання або хмарними файловими системами (наприклад, через rclone чи SFTP). Особлива увага приділяється шифруванню та збереженню паролів та ключів.

Резервне копіювання на рівні корпоративної IT-інфраструктури. Для великих організацій Borg може використовуватися як частина складнішої системи, наприклад, у поєднанні з інструментами централізованого управління (Ansible, Puppet), моніторингу та журналювання. У цьому випадку Borg Backup

може бути інтегрований у CI/CD-процеси, політики безперервної безпеки та інші автоматизовані системи.

Але одним з головних елементів при розробці рекомендацій для різних сценаріїв резервування є урахування навантаження системи, для подальшого вибору часу і методу резервування, що залежить від графіку та інтенсивності проєкту, зазвичай резервування відбувається у час коли система є досить ненавантаженою, як правило вночі, та як правило зберігаються, щоночі, щотижня.

Таким чином, Borg Backup є гнучким рішенням, яке може бути адаптоване до найрізноманітніших умов – від домашнього використання до складних корпоративних середовищ. Теоретичне обґрунтування вибору архітектури, політик безпеки та автоматизації має передувати будь-якому практичному впровадженню.

У межах розділу буде детально розглянуто процес розгортання системи, налаштування резервного копіювання, а також виконано тестування працездатності рішень у типових сценаріях.

Розгортання системи резервного копіювання з використанням Borg backup, на основі робочої станції та серверу за протоколом зв'язу ssh, де автоматизація здійснюється за допомогою планувальника linux cron, після розгортання системи тестуємо її на працездатність та правильність формування стеку бекапів

1. Встановлення Borg backup на сервер, та сервер для backup (дивитися рисунок 3.1)

```
(serve@serve)-[~]
$ sudo wget https://github.com/borgbackup/borg/releases/download/1.1.6/borg-linux64 -O /usr/local/bin/borg
sudo chmod +x /usr/local/bin/borg
[sudo] password for serve:
--2025-04-10 13:06:32-- https://github.com/borgbackup/borg/releases/download/1.1.6/borg-linux64
Resolving github.com (github.com)... 140.82.121.4
Connecting to github.com (github.com)|140.82.121.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/35517126/a6b2a108-6d78
aws4_request&X-Amz-Date=20250410T170720Z&X-Amz-Expires=300&X-Amz-Signature=e5b4dd5395b87d0c43cc264e0738566c3d
sponse-content-type=application%2Foctet-stream [following]
--2025-04-10 13:06:34-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/3551712
-east-1%2F352faws4_request&X-Amz-Date=20250410T170720Z&X-Amz-Expires=300&X-Amz-Signature=e5b4dd5395b87d0c43c
borg-linux64&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.108.133, 185.199.109.133,
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.108.133|:443... connected
HTTP request sent, awaiting response... 200 OK
Length: 15057400 (14M) [application/octet-stream]
Saving to: '/usr/local/bin/borg'

/usr/local/bin/borg      100%[=====
```

Рисунок 3.1 – Встановлення Borg backup

2. На сервері бекапів створюємо користувача borg (дивитися рисунок 3.2)

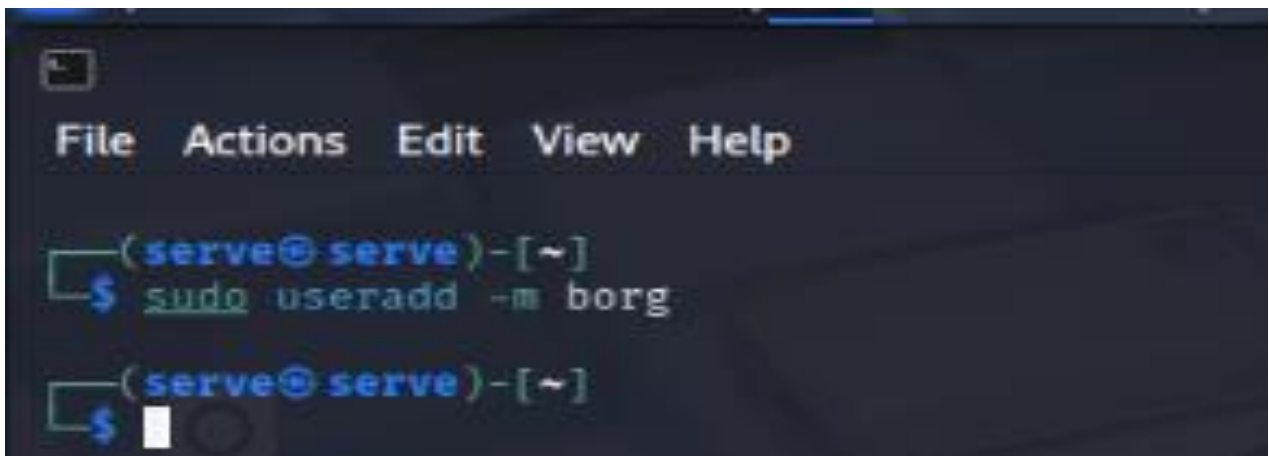


Рисунок 3.2 – Створення користувача borg

3. На клієнті генеруємо SSH-ключ(дивитися рисунок 3.3)

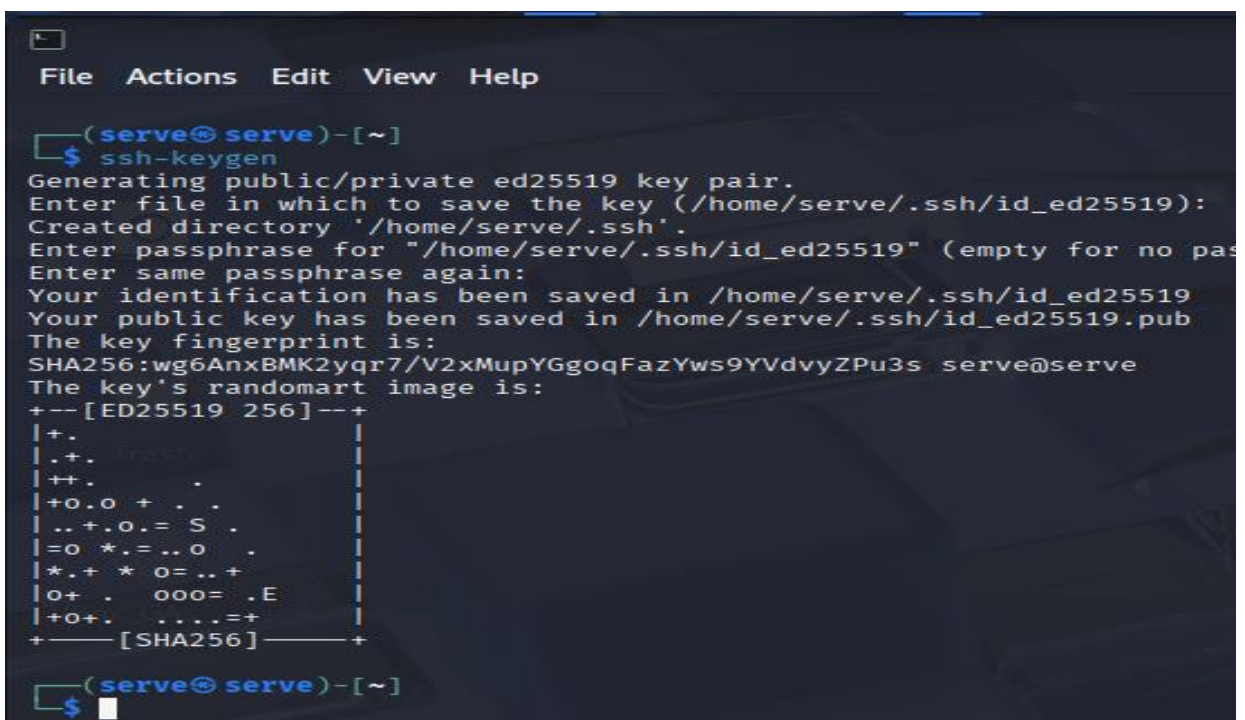


Рисунок 3.3 – Генерація SSH-ключів

4. Додаємо ключі SSH на сервері (дивитися рисунок 3.4)

```

serve@serve
File Actions Edit View Help

(serve@serve)-[~]
$ mkdir ~borg/.ssh
echo 'command="/usr/local/bin/borg serve" ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDNdaDfQUUf/XmSVWfF7PfjG1bKW00MJ63za1/E/m
e00clzx684t9d6BhMvFE9w9r+c76aV8IdbEyrkloiYd+vzt79nRkFE4CoxkpvpMtrAgbx563fRmNSPH8H5dEad44/Xb5uARiYhdLI145QuNSpAdc0adp46f
chown -R borg:borg ~borg/.ssh

```

Рисунок 3.4 – Додавання ключів SSH

5. Ініціалізуємо репозиторій на сервер із клієнта(дивитися рисунок 3.5)

```

(khorus@serve)-[~]
$ borg init -e none borg@192.168.0.106:MyBorgRepo
borg@192.168.0.106's password:
(khorus@serve)-[~]
$

```

Рисунок 3.5 – Ініціалізація репозиторію

6.Створюємо перший бекап за допомогою Borg(дивитися рисунок 3.6)

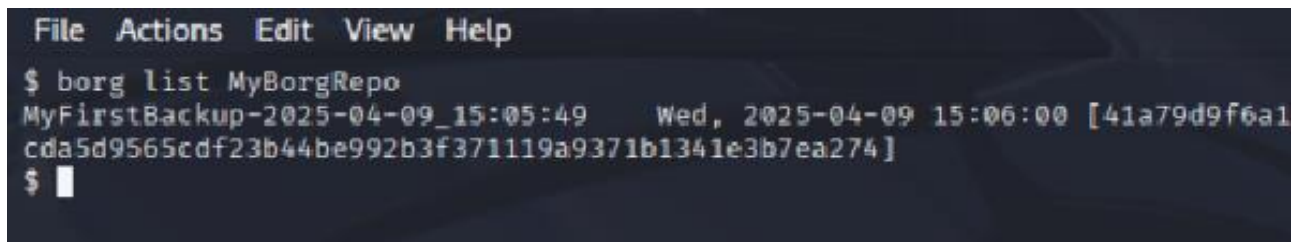
```

(khorus@serve)-[~]
$ borg create --stats --list borg@192.168.0.106:MyBorgRepo::"MyFirstBackup-{now:%Y-%m-%d_%H:%M:%S}" /etc /root
borg@192.168.0.106's password:
A /etc/console-setup/compose.ARMSCTII-8.inc
A /etc/console-setup/compose.CP1251.inc
A /etc/console-setup/compose.CP1255.inc

```

Рисунок 3.6 – Перший бекап за допомоги Borg

7.Перевіряємо чи коректно зберігся наш бекап (дивитися рисунок 3.7)



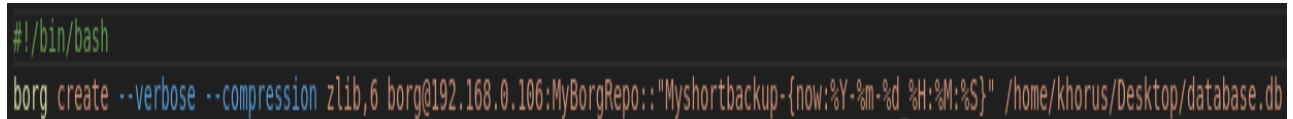
```
File Actions Edit View Help
$ borg list MyBorgRepo
MyFirstBackup-2025-04-09_15:05:49    Wed, 2025-04-09 15:06:00 [41a79d9f6a1
cda5d9565cdf23b44be992b3f371119a9371b1341e3b7ea274]
$
```

Рисунок 3.7 – Перевірка коректності зберігання

Після того, як ми зробили попередні дії можна переходити до більш серйозних систем резервування даних.

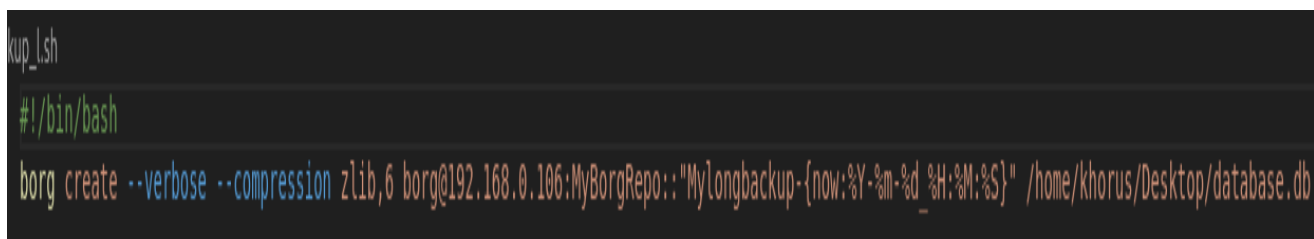
Спробуємо налагодити резервування даних з клієнта на сервер для бекапів таким чином: налагодимо все для створення так званих «hotbackup», тобто бекапів створених у режимі реального часу для відносно недовгого зберігання, та для «coldbackup» бекапів, для довгого зберігання створених з певною періодичністю.

Написання скриптів за якими буде відбуватися резервування даних (дивитися рисунки 3.8-3.9)



```
#!/bin/bash
borg create --verbose --compression zlib,6 borg@192.168.0.106:MyBorgRepo::"Myshortbackup-{now:%Y-%m-%d_%H:%M:%S}" /home/khorus/Desktop/database.db
```

Рисунок 3.8 – Скрипт для «гарячого» резервування



```
kup_ls
#!/bin/bash
borg create --verbose --compression zlib,6 borg@192.168.0.106:MyBorgRepo::"Mylongbackup-{now:%Y-%m-%d_%H:%M:%S}" /home/khorus/Desktop/database.db
```

Рисунок 3.9 – Скрипт для «холодного» резервування

Після зробленого, потрібно ці скрипти запускати на клієнті з певним періодом (для «гарячого» ми обрали 1 хвилину, «холодного» - 1 годину, час був обраний нами для тестування та наочності), щоб це зробити ми скористалися crontab, де було зроблено наступне (дивитися рисунок 3.10)

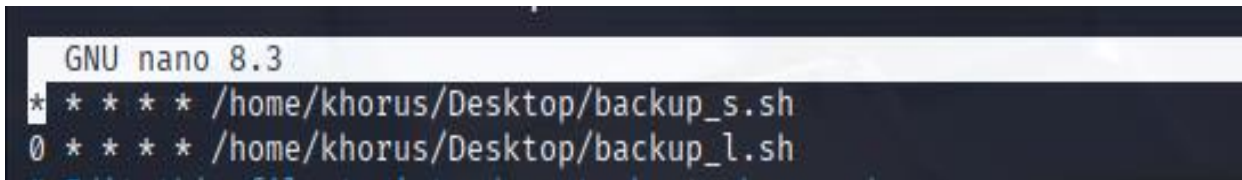


Рисунок 3.10 – Зміни crontab

Після того як відтворення скриптів було внесено до crontab, кожну 1 хвилину почали робитися «hot» бекапи, та кожну годину «cold» бекапи (дивитися рисунок 3.11)

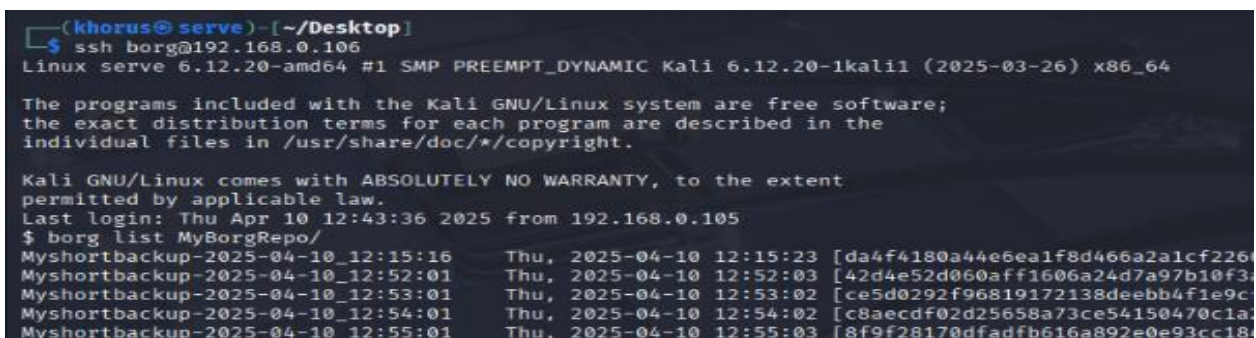


Рисунок 3.11 – Перевіряємо правильність роботи скриптів та внесення змін до crontab

Після цих маніпуляцій резервування даних за двома нашими сценаріями йде планово та стабільно, тепер можна перевірити чи все йде належним чином з бекапами, тому відкриємо якийсь бекап на клієнті, але спочатку потрібно цей бекап повернути на цю машину(дивитися рисунок 3.12)

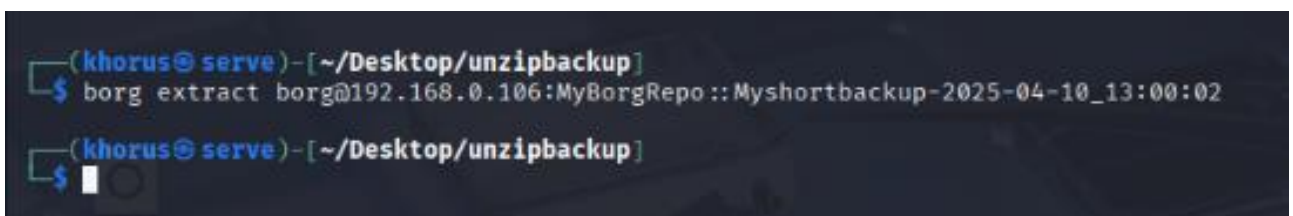


Рисунок 3.12 – Повернення файлу з репозиторію до клієнта

Далі перевіряємо що нам повернулося(дивитися рисунок 3.13)


```

(khorus@serve)-[~/Desktop]
$ cd unzipbackup/

(khorus@serve)-[~/Desktop/unzipbackup]
$ ls
home

(khorus@serve)-[~/Desktop/unzipbackup]
$ cd home

(khorus@serve)-[~/Desktop/unzipbackup/home]
$ ls
khorus

(khorus@serve)-[~/Desktop/unzipbackup/home]
$ cd khorus

(khorus@serve)-[~/Desktop/unzipbackup/home/khorus]
$ ls
Desktop

(khorus@serve)-[~/Desktop/unzipbackup/home/khorus]
$ cd Desktop

(khorus@serve)-[~/Desktop/unzipbackup/home/khorus/Desktop]
$ ls
database.db

(khorus@serve)-[~/Desktop/unzipbackup/home/khorus/Desktop]

```

Рисунок 3.13 – Перевіряємо чи повернулася наша БД

Після вище зробленого залишилася остання частина системи бекапу, але вона є досить важливою, це видалення старих бекапів. Для цього було зроблено два простих скрипти, за допомогою яких старі бекапи будуть в автоматизованому режимі видалятися, для «гарячого» резервування ми зберігаємо останніх 10 бекапів, для «холодного» резервування остані 3 бекапа

Скрипт для видалення застарілих «гарячих» бекапів (дивитися рисунок 3.14)

```

$ backup_deleteS.sh
1  #!/bin/bash
2  borg prune borg@192.168.0.106:MyBorgRepo --keep-last 10

```

Рисунок 3.14 – Скрипт для видалення хвилинних бекапів

Скрипт для видалення застарілих «холодних» бекапів (дивитися рисунок 3.15)

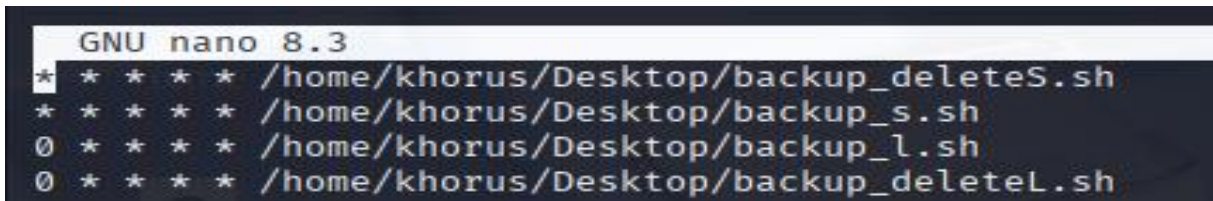
```

$ backup_deleteL.sh
1  #!/bin/bash
2  borg prune borg@192.168.0.106:MyBorgRepoL --keep-last 3

```

Рисунок 3.15 - Скрипт для видалення годинних бекапів

Далі вносимо зміни до crontab(дивитися рисунок 3.16)

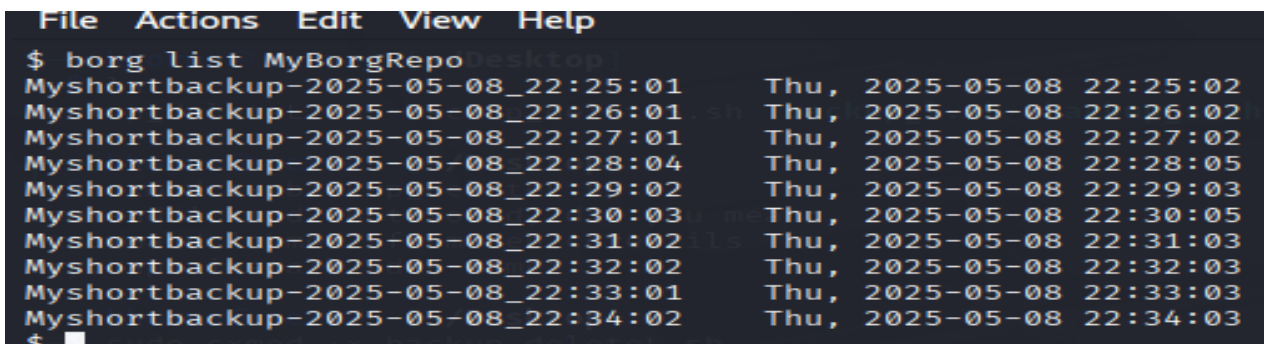


```
GNU nano 8.3
* * * * * /home/khorus/Desktop/backup_deleteS.sh
* * * * * /home/khorus/Desktop/backup_s.sh
0 * * * * /home/khorus/Desktop/backup_l.sh
0 * * * * /home/khorus/Desktop/backup_deleteL.sh
```

Рисунок 3.16 – змінений crontab

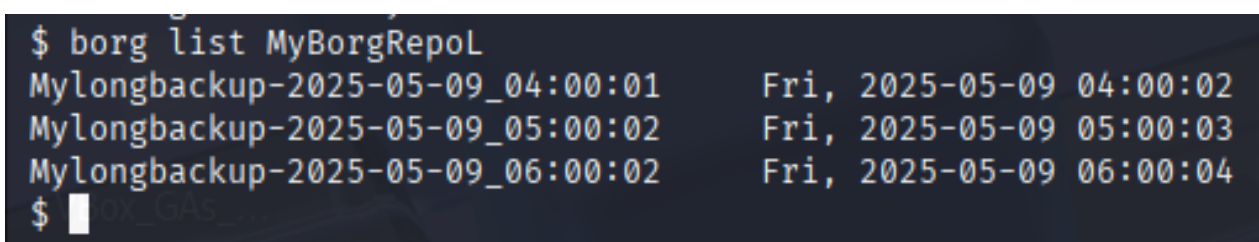
Після всіх вище зроблених дій, всю систему було протестовано досить тривалий час, все йшло за планом, без непередбачуваних ситуацій.

Перевіряємо результати(дивитися рисунки 3.17-3.18)



```
File Actions Edit View Help
$ borg list MyBorgRepo
Myshortbackup-2025-05-08_22:25:01 Thu, 2025-05-08 22:25:02
Myshortbackup-2025-05-08_22:26:01 Thu, 2025-05-08 22:26:02
Myshortbackup-2025-05-08_22:27:01 Thu, 2025-05-08 22:27:02
Myshortbackup-2025-05-08_22:28:04 Thu, 2025-05-08 22:28:05
Myshortbackup-2025-05-08_22:29:02 Thu, 2025-05-08 22:29:03
Myshortbackup-2025-05-08_22:30:03 Thu, 2025-05-08 22:30:05
Myshortbackup-2025-05-08_22:31:02 Thu, 2025-05-08 22:31:03
Myshortbackup-2025-05-08_22:32:02 Thu, 2025-05-08 22:32:03
Myshortbackup-2025-05-08_22:33:01 Thu, 2025-05-08 22:33:03
Myshortbackup-2025-05-08_22:34:02 Thu, 2025-05-08 22:34:03
$
```

Рисунок 3.17 – лист хвилинних бекапів після всіх внесених змін



```
$ borg list MyBorgRepoL
Mylongbackup-2025-05-09_04:00:01 Fri, 2025-05-09 04:00:02
Mylongbackup-2025-05-09_05:00:02 Fri, 2025-05-09 05:00:03
Mylongbackup-2025-05-09_06:00:02 Fri, 2025-05-09 06:00:04
$
```

Рисунок 3.18 – лист годинних бекапів після всіх внесених змін

Таким чином система працює штатно, резервування відбувається за розробленим нами сценарієм.

3.2 Тестування системи

Після завершення процесу розгортання системи резервного копіювання з використанням Borg було проведено серію тестувань для перевірки її працездатності, надійності та відповідності поставленим вимогам. Метою тестування є переконатися, що резервні копії створюються, зберігаються, відновлюються коректно, а також що система здатна працювати у різних сценаріях - як у звичайному режимі, так і при виникненні збоїв.

Створення резервної копії: було здійснено первинне резервне копіювання заданого набору даних. Borg показав високу швидкість обробки завдяки використанню дедуплікації, що дозволило зменшити об'єм збережених даних.

Інкрементне копіювання: було перевірено можливість створення інкрементних резервних копій, які включають лише змінені або нові файли. Це значно скоротило час і обсяг переданих даних, підтвердивши ефективність механізму Borg.

Відновлення даних: система успішно відновила дані з резервної копії у вказану директорію. Була перевірена цілісність файлів після відновлення за допомогою контрольних хеш-сум.

Імітація аварійної ситуації: було змодельовано ситуацію втрати основних даних. Borg продемонстрував здатність повного відновлення системи з резервної копії без втрат, що підтверджує її надійність у критичних ситуаціях.

Перевірка шифрування: у процесі тестування було встановлено, що збережені резервні копії шифруються відповідно до заданих налаштувань, що забезпечує конфіденційність даних.

Продуктивність і ресурси: під час тестування було зафіксовано рівень використання процесора, оперативної пам'яті та дискового простору. Система виявилась оптимізованою для використання в умовах обмежених ресурсів.

Результати тестування показали, що Borg є надійним і ефективним рішенням для резервного копіювання. Він забезпечує як швидке збереження, так і

повноцінне відновлення даних, відповідає вимогам безпеки та легко інтегрується в типову IT-інфраструктуру.

3.3 Практичні рекомендації щодо впровадження системи

Планування розкладу резервного копіювання. Рекомендується запускати резервне копіювання в нічний час або в години найменшого навантаження на систему, щоб уникнути впливу на продуктивність основних сервісів.

Застосовуйте комбінацію щоденних інкрементних копій та щотижневих повних резервних копій для оптимального балансу між швидкістю виконання і надійністю відновлення.

Використання SSH-ключів для безпечного з'єднання. Для аутентифікації між клієнтом і сервером використовуйте лише SSH-ключі замість паролів. Це підвищує рівень безпеки та дозволяє автоматизувати процес.

Автоматизація за допомогою Cron. Створіть окремі сценарії (bash-скрипти) для запуску резервного копіювання, логування результатів, моніторингу помилок і очищення старих архівів. Використовуйте crontab для щоденного виконання скриптів із детальним логуванням у системний журнал або окремий лог-файл.

Моніторинг і перевірка цілісності резервних копій. Після кожного резервного копіювання або з регулярністю раз на тиждень використовуйте команду `borg check` для перевірки архівів. Налаштуйте повідомлення на електронну пошту або месенджери (через інструменти як `msmtp`, `mailx`, Telegram Bot API) у випадку помилок.

Захист ключів шифрування. Зберігайте ключі шифрування у безпечному місці (наприклад, у менеджері паролів або зашифрованому томі). У разі втрати ключа відновлення буде неможливим.

Організація зберігання архівів. Використовуйте окремий сервер або хмарне середовище для зберігання резервних копій. Це забезпечує стійкість до збоїв локального обладнання. Застосовуйте ротацію резервних копій із збереженням лише обмеженої кількості останніх версій (`borg prune`).

Безперервне тестування відновлення. Мінімум раз на місяць проводьте тестове відновлення вибраних даних у тестове середовище, щоб упевнитися в працездатності бекапів.

Інтеграція з системами CI/CD (для корпоративного рівня). Якщо резервне копіювання впроваджується в IT-інфраструктурі підприємства, доцільно інтегрувати Borg з системами конфігураційного менеджменту (Ansible, Puppet) та CI/CD.

ВИСНОВКИ

У рамках впровадження системи резервного копіювання з використанням BorgBackup було створено безпечну, ефективну та автоматизовану систему зберігання даних. Використання Borg дозволило забезпечити надійне інкрементне резервне копіювання з високим рівнем стиснення та шифрування даних, що знижує ризики втрати важливої інформації.

Основними етапами реалізації були:

1. Вибір інструменту BorgBackup, що відповідає вимогам безпеки та ефективності.
2. Налаштування середовища для виконання резервних копій, включаючи встановлення необхідного програмного забезпечення та створення окремого користувача для бекапів.
3. Організація регулярного створення резервних копій за допомогою автоматизації через скрипти та налаштування crontab.
4. Тестування відновлення даних, що гарантує швидкий доступ до файлів у разі необхідності.
5. Впровадження заходів безпеки для захисту резервних копій, зокрема зберігання бекапів на іншому сервері, використання шифрування та контроль доступу.

Результатом стало створення надійної системи резервного копіювання, яка забезпечує регулярне збереження даних, їх безпеку та швидке відновлення в разі необхідності. Така система є важливою складовою стратегії збереження інформації на підприємствах та у організаціях, де дані мають високу цінність та потребують належного захисту від втрат.

ПЕРЕЛІК ПОСИЛАНЬ

1. Бойко В.Д., Хорус О.С. Організація та розгортання системи резервного копіювання з використання Borg // Кібербезпека в Сучасному світі: актуальні виклики: матеріали V Міжнародної науково-практичної конференції (м. Одеса 22 листопада 2024р.). 2024. Р. 147-150
2. ISO/IEC 27001:2013 - Information Security Management Systems (ISMS) [Електронний ресурс] - Стандарт, що містить вимоги до управління інформаційною безпекою, включаючи резервне копіювання. <https://www.iso.org/isoiec-27001-information-security.html> (Дата звернення: 12.04.2025)
3. Borg Documentation [Електронний ресурс] <https://borgbackup.readthedocs.io/en/stable/> (Дата звернення: 12.04.2025)
4. Бойко В.Д., Василенко М.Д., Слатвінська В.М. Версіонування файлової системи для боротьби з програмами-вимагачами // Інформатика, управління та штучний інтелект Тези восьмої міжнародної науково-технічної конференції (16 – 19 листопада 2021 року). Харків: НТУ "ХПІ", 2021. Р. 9.
5. NIST Special Publication 800-34 Rev.1 [Електронний ресурс] - Contingency Planning Guide for Federal Information Systems <https://csrc.nist.gov/publications/detail/sp/800-34/rev-1/final> (Дата звернення: 13.04.2025)
6. Wiki Рівні резервного копіювання [Електронний ресурс] <https://surl.lu/dzywju> (Дата звернення: 14.04.2025)
7. Acronis. (2023). Full, Incremental and Differential Backup Explained. [Електронний ресурс]. - Режим доступу: <https://www.acronis.com/en-us/blog/posts/incremental-differential-backups/> (Дата звернення: 14.04.2025)
8. Amazon Web Services (AWS) Backup Architecture Best Practices [Електронний ресурс] <https://aws.amazon.com/backup/> (Дата звернення: 18.04.2025)
9. GitHub репозиторій проєкту [Електронний ресурс] <https://github.com/borgbackup/borg> (Дата звернення: 20.04.2025)

10. Bakula backup documentation [Электронный ресурс]
https://www.bacula.org/15.0.x-manuals/en/developers/Bacula_Developer_Notes.html (Дата
звернення: 22.04.2025)
11. Rsync documentation [Электронный ресурс]
<https://rsync.samba.org/documentation.html> (Дата звернення: 22.04.2025)
12. Duplicity documentation [Электронный ресурс]
<https://duplicity.nongnu.org/docs.html> (Дата звернення: 22.04.2025)